D3.1 - SENSOR NETWORK ARCHITECTURE AND SPECIFICATIONS

DOCUMENT ID - TYPE:	D3.1 (REPORT)	PROJECT TITLE:	Smart Age-friendly Living and Working Environment
DELIVERABLE LEADER:	Roessingh Research and Development (RRD)	GRANT AGREEMENT Nº:	826343 (H2020-SC1-DTH-2018-1)
DUE DATE:	31/10/2019	CONTRACT START DATE:	1 January 2019
DELIVERY DATE:	08/11/2019	CONTRACT DURATION:	36 Months
DISSEMINATION LEVEL:	Public (PU)	PROJECT COORDINATOR:	BYTE S.A.
STATUS - VERSION:	Final – v1.0		

A 3-year EC Funded project
Start date: 1st January 2019
Funding: This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826343

(Page intentionally blanked)

Funded by Horizon 2020 Framework
Programme of the European Union under
Grant Agreement No. 826343

AUTHORS - CONTRIBUTORS

Name	Organization
Harm op den Akker	Roessingh Research and Development (RRD)
Miriam Cabrita	Roessingh Research and Development (RRD)
Christiane Grünloh	Roessingh Research and Development (RRD)
Dimitrios Amaxilatis	SparkWorks ITC Ltd. (SPARKS)
Nikolaos Tsironis	SparkWorks ITC Ltd. (SPARKS)
João Quintas	Instituto Pedro Nunes (IPN)
Hugo Marcos	Instituto Pedro Nunes (IPN)
Rita Kovordanyi	Cognition and Interaction Lab (COIN), Linköping University

PEER REVIEWERS

Name	Organization
Dimitrios Amaxilatis	SPARKS
João Quintas	IPN
Hugo Marcos	IPN
Andre Pardal	IPN
Charalampos Vassiliou	BYTE

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

REVISION HISTORY

Version	Date	Author/Organisation	Modifications
0.1	02.07.2019	RRD	Proposed Table of Contents
0.2	01.10.2019	RRD	Revised Table of Contents
0.3	08.10.2019	RRD	Contributions to sections 3.2, 3.3, 3.4 and 4
0.3	18.10.2019	SPARKS	Contributions to sections 3.4, 3.6, 3.7, 4 and 5
0.4	18.10.2019	IPN	Contributions to sections 3.1, 3.8 and 3.10
0.5	19.10.2019	RRD	Merged partner contributions, added sections on Smart Scale, Experience Sampling, Sleep and Fitness Tracking
0.5	23.10.2019	COIN	Contribution to section 3.9
0.6	28.10.2019	RRD, IPN	Consolidated Version, Executive Summary, Sensing Architecture, Manual for Intelligent Mouse installation.
0.7	01.11.2019	SPARKS	Review comment provided
0.8	05.11.2019	IPN	Review comment provided
1.0	08.11.2019	BYTE	Review comments embedded, final revisions and submission to the EC.

GLOSSARY

ABBREVIATION	DESCRIPTION
AAL	Ambient Assisted Living
API	Application Programmer's Interface
BLE	Bluetooth Low Energy
BPM	Beats Per Minute
D	Deliverable
ECG	Electrocardiogram
EU	European Union
FP	FRAMEWORK PROGRAMME
GPII	Global Public Inclusive Infrastructure
IoT	Internet of Things
IPN	Instituto Pedro Nunes
MSD	Musculoskeletal Disorder
R2D2	Roessingh Research and Development Database
RRD	Roessingh Research and Development
RtF-I	Raising the Floor International
SOAP	Service Oriented Architecture Platform
SPARKS	SparkWorks
USB	Universal Serial Bus
WP	Work Package

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

1. Executive Summary

The aim of WP3 is the implementation of the platform, interfaces and procedures to collect data from office workers that is related to their health, well-being and functional abilities. This deliverable defines the high-level architecture of the SmartWork Unobtrusive Sensing Framework and is structured as follows: the introduction of the deliverable and its relation to other deliverables is discussed in Section 2, the assessment of sensors and sensor devices is presented in Section 3, the setup of the in-lab data collection showcased in Section 4, and the high-level sensing framework architecture is summarized in Section 5.

Specifically, Section 2 provides the positioning of this document in the context of the project as a whole. Section 3 gives an overview of the sensors and devices used for collecting the data and presents an assessment in relation to the suitability of the sensor in the various context of use and the potential energy efficiency. The analysis comprises devices such as intelligent mouse, fitness tracker, sleep tracker, smartphone as a sensor, smart body scale, Sparks ECG Vest monitor, Sparks Environmental Sensor, posture tracker, computer activity tracker, and workplace indoor location monitor. For each component, a simplified high-level architecture overview is provided to outline the connection between the specific component and the overall SmartWork Sensing Architecture described in Section 5.

In Section 4, we describe the setup of an in-lab data collection study. Realistic data is being collected early in the project to bootstrap the machine learning processes in WP4 and to get valuable experience with the sensors from a technology and usability point of view. To prevent putting an unnecessary burden on potential end-users, this data collection is being performed internally within the SmartWork consortium.

The outcome of the previous two sections informed the design of the SmartWork Unobtrusive Sensing Framework, which is described in detail in Section 5.

Table of Contents

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

Index of Figures

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

Index of Tables

2. Introduction

The primary objective of this deliverable, as a first output of Work Package 3, is to define the high-level architecture of the SmartWork sensing framework. In order to provide personalized support, the various SmartWork services require meaningful information about the end user's context and behaviour. Although low-level processing of sensor data is part of Work Package 3, additional data processing in order to provide semantically rich information to the services is handled in the context of Work Package 4.

This deliverable consists of three core parts. The first part is providing an analysis of the various sensors needed in the SmartWork project (Section 3). In the second part (Section 4), we describe the setup of an initial Data Collection Cohort, in which sensor data samples are collected in a process internal to the SmartWork consortium. These two sections provide input to third and final part, the Sensing Architecture (Section 5) in which the high-level sensing framework architecture is defined.

2.1. T3.1 positioning and interdependencies

The relationship of D3.1 to other deliverables of SmartWork is described in Table 1 below.

Deliverable	Relationship
D2.1	*Deliverable D2.1 "State of the Art (SOA) review and benchmarking of best practices", released in* M3 of the project, reviewed the existing commercially available solutions and state-of-the art research methodology and approaches on the multi-dimensional aspects relevant for the implementation of the SmartWork system and services.
D2.2	Deliverable D2.2 "Co-creation methodology, requirements, scenarios and use cases" released in M6 retrieved the wishes and needs of end-users regarding the SmartWork system.
D2.3	*D2.3 Data Collection Protocol* [M6] established a preliminary protocol for the data collection phases within the life cycle of the project.
D3.2	This document provides the general architecture overview that serves as input to D3.2: First version of the Unobtrusive Sensor Network.
D3.3-D3.7	This document serves as a reference point, portraying the overall Sensor Network Architecture, for the rest of the WP3 deliverables.

TABLE 1: RELATIONSHIP BETWEEN OTHER PROJECT DELIVERABLES AND THE CURRENT DOCUMENT D3.1.

The results of the work described in the current deliverable D3.1 will guide the implementation of the following tasks and WPs (see Table 2 below).

Tasks/WPs	Relationship
T3.2	*T3.2 Unobtrusive sensor network implementation* [M6-M22] concerns the actual implementation of the results of the current deliverable.
WP7	*WP7: System Integration and SmartWork Services Implementation* [M16-M32] integrates technologies and implements the SmartWork services and interventions.

TABLE 2: THE CURRENT DELIVERABLE D3.1 PROVIDES INPUT TO THESE PROJECT TASKS AND WORK PACKAGES.

3. Assessment of Sensors

For each of the sensors specified in the sub-sections below, we perform an analysis of (I) the suitability of the sensor in the various contexts of use within the SmartWork services, and (II) an assessment of potential energy efficiency strengths or weaknesses.

In this section, we discuss from the point of view of *devices*, rather than from the point of view of the *information* delivered, although there is definite overlap in devices being able to deliver multiple types of information.

In Table 3 below we map the devices (rows) linking to the subsections below with the information that these devices provide (in the columns).

	Intelligent Mouse (0)	Fitness Tracker (3.2)	Dedicated Sleep Tracker (3.3)	Smartphone as a Sensor (3.4)	Smart Scale (3.5)	SPARKS ECG (3.6)	SPARKS Environmental Sensor (3.7)	Posture Tracker (3.8)	Computer Activity Tracker (3.9)	Indoor Location Monitor (3.10)
User Heart Rate	✓	✓	✓	✓		✓				
User Heart Rate Variability	(✓)					(✓)				
Peripheral Oxygen Saturation	✓									
Blood Pressure	(✓)			✓						
Galvanic Skin Response	✓									
Skin Temperature	✓									
Stress / Relaxation	✓									
Trembling	✓									
Physical Activity (Calories Burned)		✓								
Physical Activity (Steps)		✓								
Physical Activity (Distance Covered)		✓								

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

Physical Activity (Floors Climbed)		√								
Physical Activity (Intensity)		√								
Sleep Quality		√	√							
Snoring			√							
Self-Reported Physiology				√						
Urban Geolocation				√						
Ambient Noise				√						
Self-Reported Experience				√						
Self-Reported Nutrition Intake				√						
Weight				(√)	√					
RR-Interval						√				
Full ECG Recording						√				
Arrythmia Events						√				
Environmental Temperature							√			
Environmental Humidity							√			
Environmental VOC Levels							√			
Estimated Pose								√		
Computer Activity Use									√	
Estimated Indoor Location										√

TABLE 3: MAPPING OF DEVICE TO THE DIFFERENT TYPES OF INPUTS PROVIDED TO THE SMARTWORK SYSTEM.

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

3.1. Intelligent Mouse

Within the framework of the SmartWork project, an intelligent mouse capable of measuring several digital biomarkers will be integrated and further developed (i.e. SmartWork adopts the previous results achieved in CogniWin – AAL project, as stated in the proposal stage).

The main goal of the Intelligent Mouse is to assess the physiological state of the user in an unobtrusive way (i.e. using electro-physiological sensors), seamlessly integrated with the work environment.

The architecture to be developed by IPN will be a bundle of hardware and software tools, represented in the following diagram:

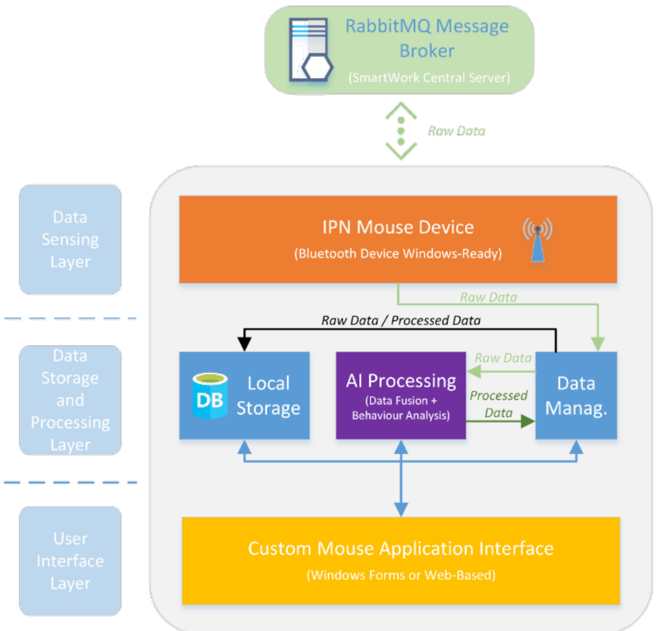


FIGURE 1: ARCHITECTURE OF THE INTELLIGENT MOUSE PART OF THE SMARTWORK SYSTEM, DEVELOPED BY IPN.

The IPN wireless mouse will be mainly used by the office workers during their everyday interaction with a desktop or laptop device. Since the device offers Bluetooth connectivity, the pointing functions will be available in all major operating systems or devices that support Bluetooth connections and have the respective driver. Regarding the sensorial part of the IPN mouse, the full deployment is assured by a daemon application (currently only developed with compatibility for Microsoft Windows), that will be responsible for the low-level interactions with the integrated mouse sensors and data gathering procedures in order to analyse relevant parameters about user's health.

The data collected will be locally analysed, stored and combined with some external data provided by SmartWork web-services in order to link the retrieved information with the respective user profile. That information is then sent via a message broker to SmartWork, where the data will be available for further consumption by other SmartWork services that need user relevant information. For this reason, an active internet connection will be needed, but that is something that is granted in a typical office workspace.

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

3.1.1. Suitability in SmartWork Contexts of Use

In the context of the SmartWork project, the intelligent mouse will be capable of sensing and inferring about the following parameters:

Sensed Parameters:

- Heart-Rate
- Heart-Rate Variability (experimental)
- Peripheral Oxygen Saturation
- Blood Pressure (experimental)
- Galvanic Skin Response (GSR)
- Skin Temperature
- Trembling

Mouse Inherent Parameters:

- Mouse XY Position
- Mouse Click
- Clicked Key

Computed Parameter:

- Stress / Relaxation

Communication Technologies:

- Bluetooth

Several parameters provided by the mouse such as Galvanic Skin Response, Heart rate, Temperature, and Trembling can be measured and used to infer psycho-motor states of the user (e.g. arousal associated with stress, anxiety, fatigue, lack of confidence, among others).

These digital bio-markers were selected because they are correlated to physiological or behavioural changes during periods of altered state.

- The **Galvanic Skin Response** reacts to changes on skin resistance due to sweat released during a stress situation.
- The **heart rate monitor** can also be used as a good indicator as it tends to increase during an altered state.
- **Temperature** also tend to increase naturally during periods of stress.
- **Mouse trembling** can indicate "insecurity" and can also contribute to the estimation of altered state; this can be performed using an Inertial Measurement Unit (IMU) on the mouse or with the application running on the computer monitoring the mouse positioning.

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

Both pointing and sensing capabilities of the mouse will be proprietary developed by IPN, combining the sensors and required hardware to unobtrusive sense the user's physiological state with the lens, optical sensor and battery required for the pointing functions of the mouse in a custom Printed Circuit Board (PCB) with only one processing unit. The microcontroller will be responsible for reading and pre-processing the raw data coming from the sensors, transmitting it over Bluetooth to the computer. Besides that, it will also implement the Bluetooth Stack, needed to ensure a proper and standard BLE connection between the device and the user's computer.

Regarding the software architecture, it should be easily integrated and deployed with the remaining modules that will be developed in the scope of the SmartWork project by the remaining partners. For that reason, the programming framework and tools for the design of the software will be decided in agreement with the remaining partners.

Additionally, it is intended that the intelligent mouse be a plug-n-play Bluetooth HID device under Microsoft Windows (and possibly will be supported by other OSs, but a thorough investigation of such possibility is out of scope of the SmartWork project).

The main focus of the upcoming software development is to provide data for the following indicators based on raw-data retrieved:

- Feeling Warm/Cold (Temperature sensor)
- Slow/Fast Heart-Rate (Heart-Rate sensor)
- Dry/Wet skin (Skin response sensor)
- Sudden/smooth Mouse motion (Mouse motion stream and IMU)

The previous information will also be used to feed a behaviour analysis method that will be responsible for estimating the user's psycho-motor state, based on the raw data collected and some *a priori* user related aspects. The psycho-motor states to be inferred are:

- Anxiety/Stress
- Frustration
- Neutral /Normal
- Fatigue/Sleepiness

By definition, the psycho-motor state estimation will output a single value, representing the probability or likelihood of the user to be at that state. This means that the mouse software will not firmly classify any of the psycho-motor states, providing only a partial indication for each one of the possible scenarios. The information will be made available for the remaining modules so it can be fused with other relevant information, the eye-tracker, for example, in order to provide a more robust inference.

Since the data about the psycho-motor state of the user are inferred and estimated, the system should be put to the test in end-user trials with long periods of runtime and the output should be compared with specialists' opinion.

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

3.1.2. Assessment of Energy Efficiency

The mouse to be developed will be a battery powered Bluetooth device, and some special constraints were taken into consideration in order to minimize the energy consumption and extend the battery lifespan. This way, it is ensured that the user does not have the burden of having to charge another portable device. The main topics that will make the mouse an energy efficient device include:

- **Fully customized architecture:** The mouse will be entirely developed at IPN both pointing and sensing modules. Some of the internal components, namely the microcontroller, will be shared among the pointing and sensing capabilities of the device, ensuring a proper reuse of the components and saving in terms of energy, once there are no duplicated components performing different functions.
- **Bluetooth Low-Energy communication:** the data link between the wireless device and the computer will be established using the BLE protocol. Besides this protocol having a slightly lower data-rate, when compared with classic Bluetooth protocol, it surpasses the classic approach in terms of energy efficiency, which is the most valuable characteristic in the scope of this project.
- **Low-Power optical sensor:** For the pointing function of the mouse, the optical sensor chosen has a very low current consumption when in operation, contributing to a higher efficiency of the device as a whole.
- **Efficient Microprocessor:** Despite the vast list of microcontrollers available, the chosen unit also has a low-current consumption when active. In addition, several different low-power modes can be configured to fulfil all the computational needs of the device and minimize the amount of current drawn when in sleep mode. This, in conjunction with an integrated architecture packing different peripherals in a single package helps to keep the energy consumption at a low level.

3.1.3. Component Architecture

In order to provide the complete picture of the SmartWork Sensing Architecture in Section 5, we provide for each of the components their simplified high-level architecture overview. Below, in Figure 2, the high-level component architecture is given for the Intelligent Mouse.

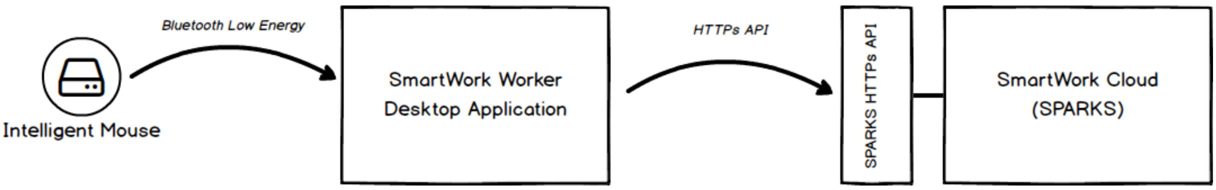


FIGURE 2: HIGH-LEVEL ARCHITECTURE OF THE INTELLIGENT MOUSE COMPONENT.

Funded by Horizon 2020 Framework
Programme of the European Union under
Grant Agreement No. 826343

3.2. Fitness Tracker

We present below the process and specifications on the final selection regarding the dedicated fitness tracker to be used in the SmartWork project.

In order to guide the process of choosing the optimal fitness tracking device for the project's purposes, Table 4 below provides the pros and cons of each one of the devices as they were listed in *D2.1 "State of the Art (SOA) review and benchmarking of best practices"* within the Fitness Trackers category.

Considering that the fitness tracker must be worn while at work, on-the-go, and at home, its usability and *wearability* are two of the most important criteria when selecting a device. Consequently, research-oriented devices (Actigraph and MOX 2) were dropped out as they are less comfortable to wear, sometimes even bulky, and have shorter battery life than consumer-oriented devices.

Next, considering the inexistence of an open API, or only limited to certain operating systems in the market, the Xiaomi and the Apple Watch options were eliminated.

Finally, given the fact that the Fitbit devices are widely used in research (as shown by large research platforms, such as Fitabase that supports researchers collecting and analysing data from Fitbit devices[1]).

	Actigraph WGT3X-BT	MOX2	Fitbit	Withings	Xiaomi MiBand 4	Apple Watch
Cost	-	+/-	+/-	+/-	+	-
Output in steps, distance, minutes in physical activity intensity	+	+	+	+	+	+
Heart rate measurement	-	-	+/-	+/-	+	+
Communication protocol / energy efficiency	+	+	+	+	+	+
Battery	-	-	+/-	+	+	-
Ease of use / Aesthetics	-	-	+	+	+	+
Transparency (raw data available)	+	+	-	-	-	-
Waterproof	+	-	+/-	+/-	+	+

[1] https://www.fitabase.com/

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

Open API	-	+	+	+	-	+/-
Use of the sensor in (clinical) research	+	+/-	+/-	-	-	+

TABLE 4: COMPARISON OF THE FITNESS TRACKERS ELICITED IN D2.1 STATE OF THE ART.

Given the variety of Fitbit devices available on the market, we have performed a second-stage analysis comparing the different devices based on costs, functionalities and wearability (see Table 5 below). The models Charge3 and the Fitbit Versa 2 were the models that scored best. Older adults participating in previous research studies have expressed difficulties in putting on and off the wrist band of the Inspire line, due to the fact that it is too small for people with limitations in the fine motor skills. As SmartWork should be as inclusive as possible, we have decided for the Fitbit Charge 3 as the ultimate fitness tracker to be used in the project.

	Inspire	Inspire HR	Charge 3	Versa 2
Cost[2]	€69.95	€99.95	€149.95	€ 199.95
Heart rate tracking	-	+	+	+
Waterproof	-	+	+	+
Sleep stages tracking	-	+	+	+
Wearability	-	-	+	+

TABLE 5: COMPARISON OF SEVERAL FITBIT MODELS BASED ON COSTS, FUNCTIONALITIES AND WEARABILITY.

It is important to highlight that, although the **Fitbit Charge 3** was chosen as the preferred device within the SmartWork trials, the SmartWork system should be built aiming for scalability, and therefore, able to use other devices, whenever possible.

3.2.1. Suitability in SmartWork Contexts of Use

The fitness tracker device to be used in the SmartWork project has as a primary function to provide input to the HealthyMe service.

Sensed Parameters:

The following information is provided with minute-by-minute accuracy:

- Physical Activity (Calories Burned)

[2] Prices as taken from the official Fitbit website as of the 4th of October, 2019.

- Physical Activity (Steps Taken)
- Physical Activity (Distance Covered)
- Physical Activity (Floors Climbed / Elevation)
- Physical Activity (Intensity Classification – Sedentary, Lightly Active, Fairly Active, Very Active)
- Heart Rate

Communication Technologies:

- Bluetooth low energy

Communication with SmartWork components:

- There is already an existing connection between Fitbit and the Roessingh Research and Development Database (R2D2). The following section define the details on creating and managing the Fitbit-to-R2D2 connections (see Section 3.2.2).

3.2.2. API Details

3.2.2.1. Verifying Fitbit Link Between User and R2D2

/sensor/fitbit/project/{project}/verifyLink	GET
Description	
Checks whether a user is linked to a Fitbit sensor in the specified project. The project must support Fitbit sensors. If the link exists, the result will contain details about the link. Otherwise the link will be null.	
Authorisation	
Users need to have access to the specified project. Admins can access all projects. All users can access their own links within a project that they can access. Patients can only access their own links. Professionals can access the links of users to whom they were granted access. Admins can access all links.	
URL parameters	

{project}	Project code. The project must support Fitbit sensors.
user	(optional) Email address of the user to which the Fitbit should be linked. Omit or leave empty to check for yourself.

Content

–

Response (application/json)

JSON object with the following property:

link	The link (see below) or null.

A link is a JSON object with the following properties:

linkId	Link ID.
dataSynced	true if the first data sync from Fitbit to R2D2 has been completed, false otherwise

Errors

–

Example

Request

```
GET https://servlets.rrdweb.nl/r2d2/v5.1.1/sensor/fitbit/project/default/verifyLink
    ?user=user@rrd.nl
X-Auth-Token: …
```

Response

```
{
    "link":{
        "linkId":"dd0c77e218fa4b4596267cdb32afdd2d",
        "dataSynced":true
```

```
    }
}
```

3.2.2.2. Linking Fibit Account to R2D2 Account

/sensor/fitbit/project/{project}/link	POST
Description	
Creates a link between a user and a Fitbit sensor in the specified project. If a link already exists, it will be updated, but only for the specified project. The project must support Fitbit sensors.	
Authorisation	
Users need to have access to the specified project. Admins can access all projects. All users can access their own links within a project that they can access. Patients can only access their own links. Professionals can access the links of users to whom they were granted access. Admins can access all links.	
URL parameters	
{project}	Project code. The project must support Fitbit sensors.
user	(optional) Email address of the user to which the Fitbit should be linked. Omit or leave empty to link to yourself.
Content (application/x-www-form-urlencoded)	
fitbitAuthCode	The Fitbit authorization code. This code should be obtained using the OAuth 2.0 Authorization Code Grant flow as documented at https://dev.fitbit.com/docs/oauth2/.
redirectUri	(optional) The redirect URI that was used to get the authorization code. This should be one of:

	rrdnlapp://app.rrdweb.nl/fitbit_callback (default) http://app.rrdweb.nl/web_fitbit_callback
Response (application/json)	
JSON object with details about the link. It has the following properties:	
linkId	Link ID.
dataSynced	This is always false. It indicates whether the first data sync from Fitbit to R2D2 has been completed.
Errors	
–	
Example	

Request

POST https://servlets.rrdweb.nl/r2d2/v5.1.1/sensor/fitbit/project/default/link

X-Auth-Token: …

Content-Type: application/x-www-form-urlencoded

user=user@rrd.nl&fitbitAuthCode=…

Response

```
{
   "linkId":"dd0c77e218fa4b4596267cdb32afdd2d",
   "dataSynced":false
}
```

3.2.2.3. Remove a Fitbit to R2D2 Account Link (Project Specific)

/sensor/fitbit/project/{project}/link	DELETE

Description	
Deletes any link between the specified user and a Fitbit sensor in the specified project. Any links in other projects will remain.	
Authorisation	
Users need to have access to the specified project. Admins can access all projects. All users can access their own links within a project that they can access. Patients can only access their own links. Professionals can access the links of users to whom they were granted access. Admins can access all links.	
URL parameters	
{project}	Project code. The project must support Fitbit sensors.
user	(optional) Email address of the user for whom the Fitbit link should be deleted. Omit or leave empty to delete your own Fitbit link.
Content	
–	
Response (application/json)	
–	
Errors	
–	
Example	
Request DELETE https://servlets.rrdweb.nl/r2d2/v5.1.1/sensor/fitbit/project/default/link	

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

?user=user@rrd.nl

X-Auth-Token: …

Response

–

3.2.2.4. *Remove a Fitbit to R2D2 Account Link (All Projects)*

/sensor/fitbit/link DELETE
Description
Deletes all links between the specified user and any Fitbit sensor.
Authorisation
(details ommitted)
…

3.2.2.5. *Wait for a Fitbit Data Synchronisation*

/sensor/fitbit/project/{project}/waitDataSync v5.0.3 GET
Description
Tries to wait until the first data sync is complete for a new Fitbit link. It waits at most 30 seconds and returns if the data sync is complete or it can't be completed. The result indicates possible reasons.
Authorisation
Users need to have access to the specified project. Admins can access all projects.

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

All users can access their own links within a project that they can access.

Patients can only access their own links.

Professionals can access the links of users to whom they were granted access.

Admins can access all links.

URL parameters

{project}	Project code. The project must support Fitbit sensors.
user	(optional) Email address of the user to which the Fitbit is linked. Omit or leave empty to check for yourself.

Content

–

Response (application/json)

JSON string with one of the following result codes:

- COMPLETE: The first data sync is complete.
- IN_PROGRESS: The first data sync was still in progress after 30 seconds.
- NO_TRACKER: The link no longer exists because no tracker was attached to the Fitbit account.
- INVALID_TOKEN: The link no longer exists because the access token was invalid and could not be refreshed.
- NO_PERMISSION: The link no longer exists because no sufficient permissions were granted by the Fitbit user.
- NO_LINK: The link no longer exists for an unknown reason.
- ERROR: The Fitbit data could not be downloaded at this time because of an error.

Errors

–

Example

Request

GET https://servlets.rrdweb.nl/r2d2/v5.1.1/sensor/fitbit/project/default/waitDataSync

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

?user=user@rrd.nl

X-Auth-Token: …

Response

"COMPLETE"

3.2.2.6. *Request synchronisation of Fitbit data to R2D2 Account*

/sensor/fitbit/project/{project}/requestDataSync	GET
Description	
Requests to run a data sync for a Fitbit link now. This query waits until the sync is completed, the link with Fitbit is removed, or an error occurs.	
Authorisation	
Users need to have access to the specified project. Admins can access all projects. All users can access their own links within a project that they can access. Patients can only access their own links. Professionals can access the links of users to whom they were granted access. Admins can access all links.	
URL parameters	
{project}	Project code. The project must support Fitbit sensors.
user	(optional) Email address of the user to which the Fitbit is linked. Omit or leave empty to run a sync for yourself.
Content	
–	

Response (application/json)
JSON string with one of the following result codes: • COMPLETE: The data sync is complete. • NO_TRACKER: The link no longer exists because no tracker was attached to the Fitbit account. • INVALID_TOKEN: The link no longer exists because the access token was invalid and could not be refreshed. • NO_PERMISSION: The link no longer exists because no sufficient permissions were granted by the Fitbit user. • NO_LINK: The link does not exist or no longer exists for an unknown reason. • ERROR: The Fitbit data could not be downloaded at this time because of an error.
Errors
–
Example
Request GET https://servlets.rrdweb.nl/r2d2/v5.1.1/sensor/fitbit/project/default/requestDataSync ?user=user@rrd.nl X-Auth-Token: … **Response** "COMPLETE"

3.2.3. Assessment of Energy Efficiency

From the project's overall State-of-the-Art analysis document (D2.1), the following two requirements were defined related to the energy efficiency of the Fitness Tracker device:

- **Battery longer than 5 days**: to reduce the burden on the user of having to charge "yet another device";
- **Bluetooth Low Energy**: this feature enables continuous communication with the smartphone, and consequently, with the other SmartWork services. This allows real-time feedback everywhere and at any time. In line with the energy efficiency requirements of SmartWork, Bluetooth low energy is the preferred communication protocol.

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

For the chosen Fitness Tracker device, these requirements are met. Since these are commercial devices, a more in-depth analysis of energy efficiency is not possible.

3.2.4. Component Architecture

In order to provide the complete picture of the SmartWork Sensing Architecture in Section 5, we provide for each of the components their simplified high-level architecture overview. Below in Figure 3, the high-level component architecture is given for the Fitness Tracker device.

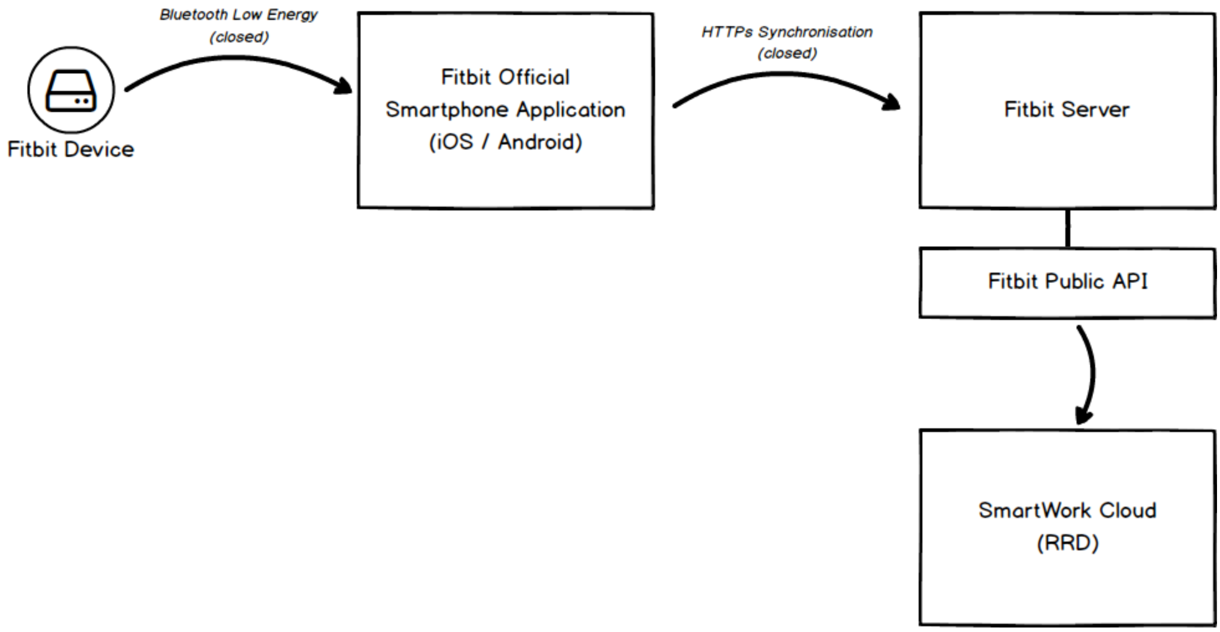


FIGURE 3: HIGH-LEVEL COMPONENT ARCHITECTURE FOR THE FITNESS TRACKER SENSOR.

Funded by Horizon 2020 Framework
Programme of the European Union under
Grant Agreement No. 826343

3.3. Dedicated Sleep Tracker

3.3.1. Suitability in SmartWork Contexts of Use

Within *D2.1 "State of the Art (SOA) review and benchmarking of best practices"* we have explored several modalities to monitor the user's sleep. Medical oriented devices such as EEG and polysomnography are out of the scope due do the complexity of use, outside the clinical setting. However, as technology evolves, the SmartWork team will keep an eye out to see if there is any consumer-oriented device able to perform this type of analysis, and possibly, reconsider our choice of device. Smartphone-based applications were also eliminated from the final discussion as there is little evidence on their accuracy and they have been barely used in research trials. The final selection was therefore between the Beddit (from Apple)[3], and Sleep (from Withings)[4], two devices placed on the mattress and that monitor sleep. The Beddit device was excluded as an option, as the API is only compatible with iOS, and therefore, not inclusive for users with Android devices. As an additional advantage, the Withings device is already integrated with R2D2, reducing efforts. Having said that, the **Withings Sleep** device is the preferred to measure sleep, in the case we opt for a dedicated sleep tracking device.

However, it is worth pointing out that the Fitbit Charge 3, the fitness tracker chosen in Section 3.2, also tracks sleep. Therefore, one option to lower the costs of a SmartWork system in the future was to use the Fitbit device both for fitness and sleep tracker. In terms of output, Withings Sleep is the one of the two devices which provides a snoring detection and analysis over time. The biggest difference between the two devices is that the Fitbit Charge 3 must be worn on the wrist and the Withings Sleep device must be placed under the mattress. Consequently, the data collected by the Withings Sleep might be disturbed in cases when the user shares a bed with another person. Furthermore, by being a wearable device, the Fitbit Charge 3 is able to detect sleep even if the user is not sleeping in the bed; this is the case of naps on the couch, for example. Within the context of the SmartWork project, we are planning to perform a small-scale study comparing data collected with the Withings Sleep device, Fitbit, and self-reported questionnaires.

Sensed Parameters:

- Withings:
    - sleep duration
    - sleep onset
    - time to wake
    - sleep cycles (deep, light and REM phases)
    - continuous and average heart-rate

[3] https://www.beddit.com/
[4] https://www.withings.com/nl/en/sleep

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

- snoring duration
- sleep quality score

- Fitbit:
  - dateOfSleep
  - duration (in ms)
  - efficiency
  - isMainSleep (boolean)
  - levels (incl. count, minutes, thirtyDayAvgMinutes) for deep, light, rem and wake.
  - minutesAfterWakeup
  - minutesAsleep
  - minutesAwake
  - minutesToFallAsleep
  - startTime
  - timeInBed (in min)
  - type (stages)
  - totalMinutesAsleep
  - totalSleepRecords
  - totalTimeInBed

Communication Technologies:

- Withings:
  - Bluetooth 4.0 for installation
  - Wi-Fi 2.4 GHz b/g/n; WEP/WPA/WPA2
- Fitbit Charge 3:
  - Bluetooth low energy

Communication with SmartWork components:

The communication between the Fitbit server and R2D2 is explained in the component architecture section for the Fitness Tracker (see Section 3.2.4). The integration of the Withings Sleep with R2D2 will be developed following the same architecture schema (i.e. an integration to a public API that grants access to data stored on the proprietary device vendor's server).

3.3.2. Assessment of Energy Efficiency

For the Withings Sleep sensor, energy efficiency is not a relevant issue, since the tracker is connected to a main power supply next to the user's bed.

For the alternative measuring of sleep through the wearable Fitbit device, see the previous section on energy related to the Fitness Tracker (Section 3.2.3).

3.3.3. Component Architecture

In order to provide the complete picture of the SmartWork Sensing Architecture in Section 5, we provide for each of the components their simplified high-level architecture overview. Below in Figure 4, the high-level component architecture is given for the Dedicated Sleep Tracker (Withings Sleep) device. The architecture is conceptually the same as for the Fitbit Fitness Tracker device as described in Section 3.2.4.

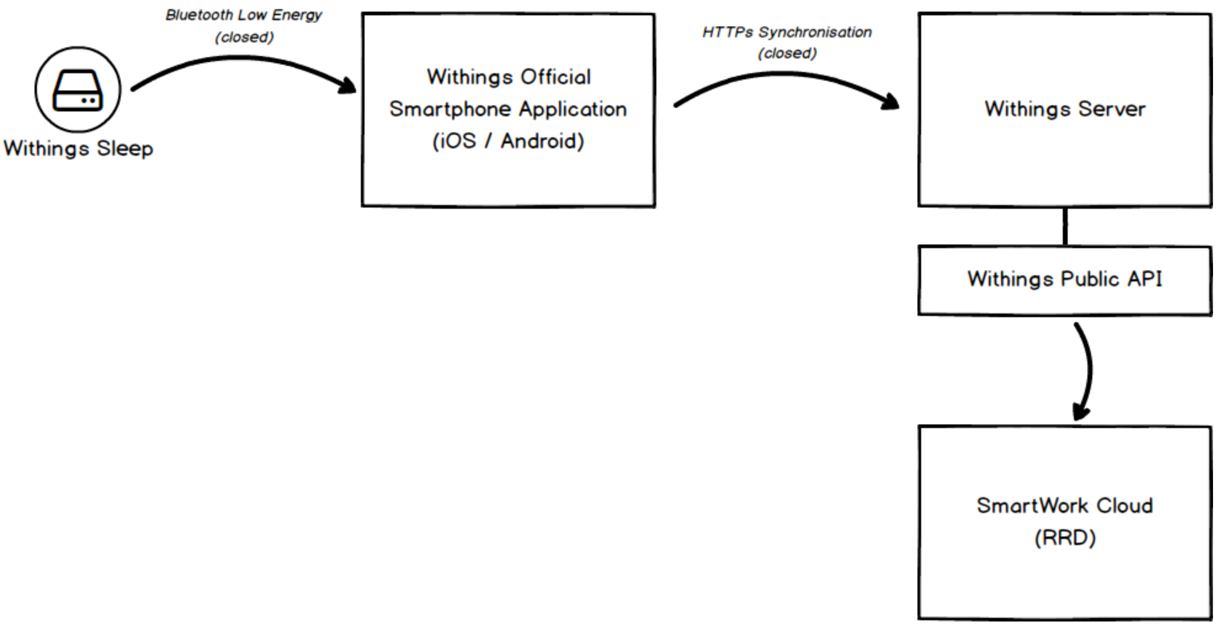


FIGURE 4: HIGH-LEVEL COMPONENT OVERVIEW FOR THE DEDICATED SLEEP TRACKER.

3.4. Smartphone as a Sensor

3.4.1. Suitability in SmartWork Contexts of Use

Smartphones are an integral part of our everyday lives. We tend to carry them around at home, at work and on the go. This is also the case with older workers that, although not so familiar with new technologies, have in most cases adopted the use of smartphones for their simplicity of interaction compared to operating a desktop or laptop computer. As a result, SmartWork will provide simple interfaces for interacting with its users through their smartphones for data collection and background sensing as well as providing input of self-reported measurements. Modern smartphones are also equipped with multiple sensors, including Location sensors, a microphone and communication interfaces (e.g. Wi-Fi, Bluetooth). These two facts can allow us to effectively monitor both the worker's location and their environments as well as transform the smartphone into a gateway device that can communicate with and integrate external devices to the SmartWork system.

3.4.1.1. Self-Reported Physiological Measurements

Based on the input received from the initial discussions with the SmartWork users we have identified a strong request for monitoring a set of chronic conditions older workers usually suffer from and can be related to the habits of an office worker. Such conditions are Diabetes, Asthma, Hypertension and other heart related conditions. Additionally, users appeared concerned about their dietary habits and wanted a method for monitoring their calorie intake as well as their eating and drinking habits while at work and/or at home. Based on this information we chose to include in the *healthyMe* application and the well-being profiles of the users a method for keeping track of such conditions without changing the habits and everyday schedules of the users that will participate in the trials. Users will keep monitoring their conditions based on their doctor's suggestions as they would without SmartWork and will be able to log their body measurements and get reminders on when they need for example to measure their blood pressure, using their own blood pressure monitor and input the values back to the application. In more detail, we monitor the following conditions and collect the data presented.

Sensed Parameters:

- Blood Sugar Level (mg/dL or mg/L)
- Forced exhalation volume in one second - FEV1 (L)
- Forced vital capacity – FVC (L)
- Systolic Pressure (mmHg)
- Diastolic Pressure (mmHg)
- Heartrate (bpm)

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

3.4.1.2. Urban Geolocation Monitoring

In order to provide users of the SmartWork platform with better suggestions and interventions for their daily routines we plan to use the user's location to understand whether the user is at work, at home or on the move. The location data that will be collected are limited to context information about where the user is (e.g. at work or at home) and not the actual GPS coordinates of the user. Each user will be able to set their home and work locations on their phones. This information will be stored only on their phones so SmartWork will have no access to it. Then using their phone's geolocation services, we will be able to understand if they are close to any of the locations defined and then report their contextual location to the SmartWork services.

Sensed Parameters:

- Urban Geolocation (user's approximate position – Work, Home, Unknown).

3.4.1.3. Ambient Noise Monitoring

Noise pollution is a common problem in office workplaces, something that was also pointed by many workers that replied to our questionnaires. To measure the noise levels at the worker's workplace we are going to use the data from their phones, as they are always usually placed near their workplace. This will be implemented using the phone's microphone for a small duration (less than 0.5 seconds) every 1 minute or less. The application will measure only the maximum amplitude of the sound waves from the phone and cannot record or store the actual sound recorded with the microphone. Users are also able at all times to disable the noise monitoring from the application on their phone using the settings of the application.

Sensed Parameters:

- Ambient Noise Level (decibels)

3.4.1.4. *Experience Sampling Method*

The smartphone can also be used to collect subjective experience of the user. The Experience Sampling Method (ESM), also known as Ecological Momentary Assessment, concerns the assessment of feelings or activities at the current moment that they happen, by directly asking the user to report on his/her experiences. By reporting on the experience at the current moment, ESM reduces the recall bias, a strong advantage when compared to other methods such as diary logging. Moreover, ESM can be used to validate the models generated from physiological data. For example, if stress is detected with physiological measurements, the user can be asked whether he is indeed experiencing stress.

Despite being often set up as short questions designed to avoid disturbing the user, when a large amount of questions is asked in a short period of time or if the questions are too repetitive, the user can become annoyed leading to unreliable data.

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

The Activity Coach application designed by RRD has an ESM module that allows the researcher to tailor the questions and timing to the exact study. Figure 5 provides an example developed in the context of wellbeing of the older worker, within the AAL Funded project PEARL, in which ESM was applied to investigate the physical activity, fatigue and satisfaction associated to several work activities (e.g. emailing) and contexts (e.g. alone vs. with colleagues).

Sensed Parameters:

- Self-Reported Experience

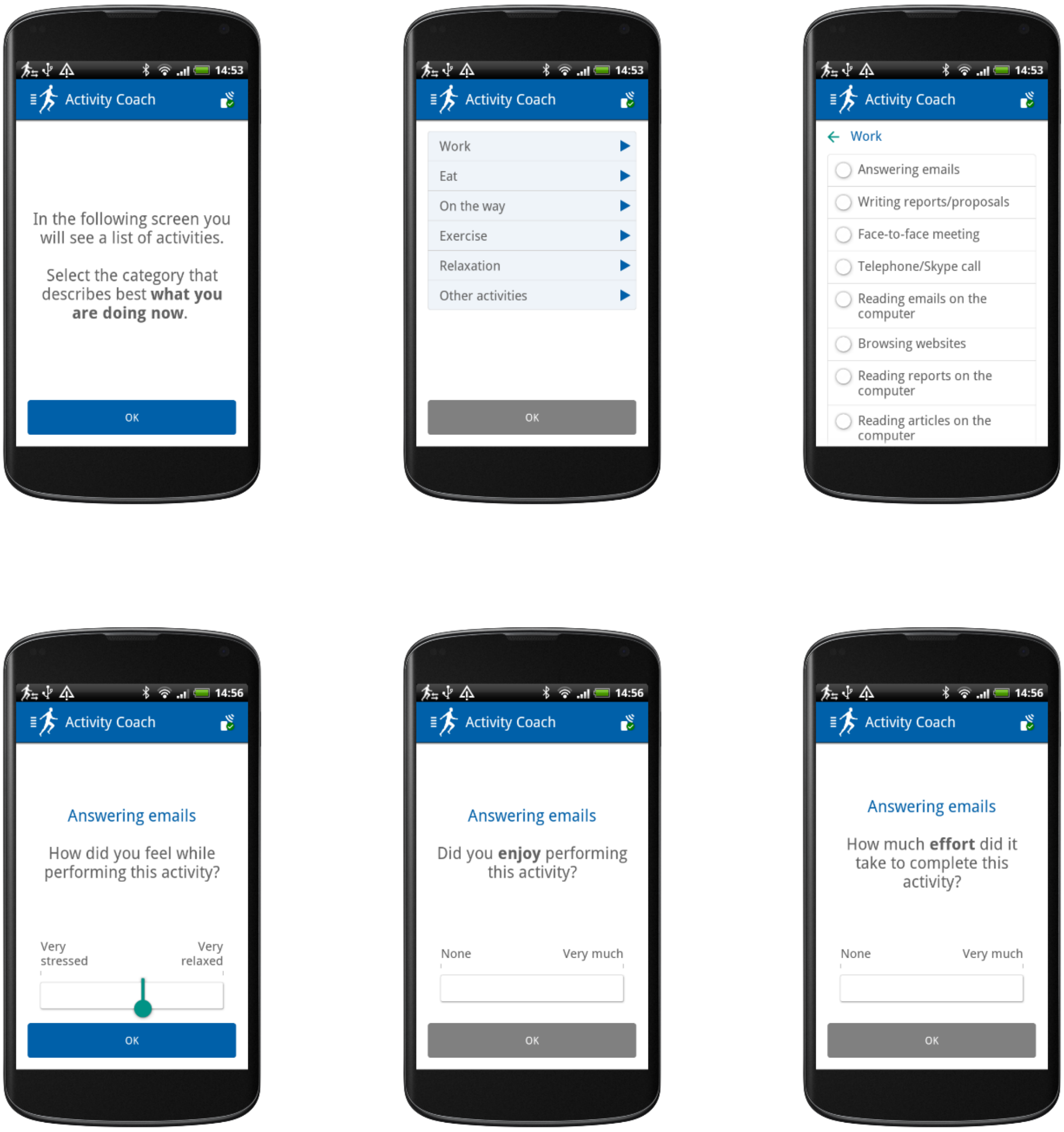


Figure 5: Example of Experience Sampling Method module within the Activity Coach application applied in the context of office workers (AAL Funded project PEARL).

3.4.1.5. Daily Nutritional Intake Log

Healthy nutrition is an important aspect of a general healthy lifestyle and thus becomes a relevant element for the SmartWork *healthyMe* service. Reliable diet registration is needed as a first step towards influencing dietary behaviour in a positive way. However, diet registration is very time consuming for the end-user and compliance towards Diet Logs is low. Regularly, essential information about ingredients, portion sizes and preparation methods to get accurate nutritional information is lacking. For the healthyMe service, we aim to develop an interactive and personalized diet registration method that can be used on the smartphone. With this tool we can easily gather a lot of accurate information and we can monitor dietary changes.

Sensed Parameters:

- Self-Reported Nutrition Intake

3.4.2. Assessment of Energy Efficiency

Energy efficiency is crucial for all smartphone applications, due to their importance in our lives and the limited capacity of their batteries. The SmartWork smartphone application is designed to run at all times, mainly in the background, collecting data, transferring data to the cloud and receiving push notifications. To avoid issues with excessive power consumption we base our design on well-established software libraries and event-based interactions instead of continuous polling on the smartphone's sensors.

Starting with the Geolocation, we use the Android Geofencing API instead of the GPS sensor of the phone directly as we are not interested in the actual location coordinates. This offers us the same results without any significant change in the power drain of the phone. Similarly, noise monitoring is done using Android AsyncTasks, a feature that allows to schedule sampling on predefined intervals executed by the operating system. Finally, cloud to device notifications will be implemented using Firebase Messaging, a method for receiving push notifications instead of continuous polling on the SmartWork services. Push notifications are much more efficient than any other self-developed solution.

A more detailed power consumption analysis of the developed application will be provided in future deliverables, as the development of this application progresses.

3.4.3. Component Architecture

In order to provide the complete picture of the SmartWork Sensing Architecture in Section 5, we provide for each of the components their simplified high-level architecture overview. Below in Figure 6, the high-level component architecture is given for the Smartphone as a Sensor.

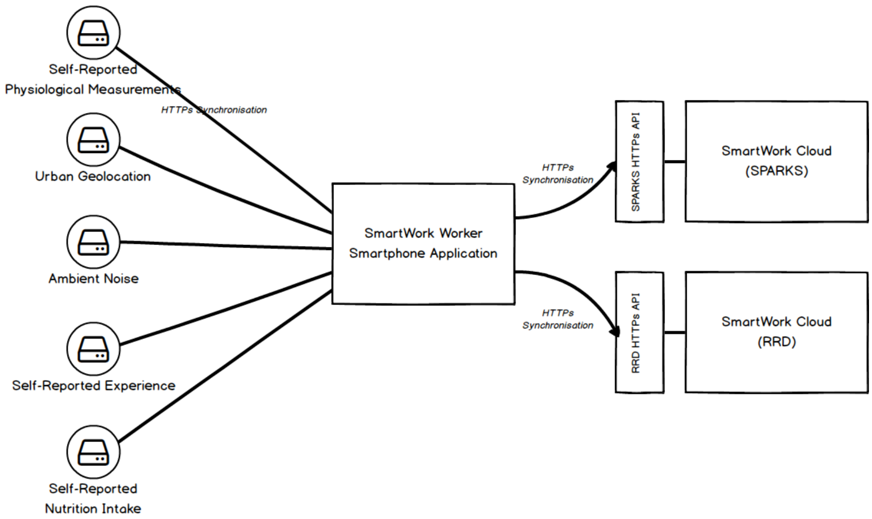


FIGURE 6: HIGH-LEVEL COMPONENT ARCHITECTURE FOR THE SMARTPHONE-AS-A-SENSOR.

Funded by Horizon 2020 Framework
Programme of the European Union under
Grant Agreement No. 826343

3.5. Smart Scale

3.5.1. Suitability in SmartWork Contexts of Use

Anthropometric parameters, such as weight and BMI, reflect both health and nutritional status. Monitoring weight is one of the most prevalent actions to monitor one's health, as everyone has measured their own body weight at least once in a lifetime. Smart scales measure body weight, often together with other parameters, and automatically synchronize and store the data collected in order to show trends over time to the user. Following the overview started in *D2.1 "State of the Art (SOA) review and benchmarking of best practices"*, in this sub-chapter we compare four body scales, two from Withings and two from Fitbit (see Table 6 below). These brands were selected as the use of other devices from these companies is expected for monitoring other health domains (see section 3.2 and 3.3). Both Withings and Fitbit have a simpler and a sophisticated version of a smart scale.

The Withings Body Cardio and the Fitbit Aria 2 scales use bioelectrical impedance analysis to determine body composition (body fat % and lean mass %), making these scales unsuitable for users with pacemakers or other internal medical devices. Considering the high prevalence of cardiovascular diseases and consequent high rate of implantable devices among the older population, these two scales were excluded.

The Withings Body scale and the Fitbit Aria Air have very similar specifications; however, at the time of this deliverable (October 2019) the Fitbit Aria Air is only available for pre-ordering, making the **Withings Body** scale the preferred option.

	Withings Body	Withings Body Cardio	Fitbit Aria Air	Fitbit Aria 2
Cost	60€	150€	60€	130€
Output: weight, BMI,	+	+	+	+
Output: body fat % and lean mass %	-	+	-	+
Output: Heart rate, pulse wave velocity	-	+	-	-
Multiple users (up to 8 users)	+	+	+	+
Battery	4 AAA batteries	Integrated, rechargeable	3 AAA batteries	3 AA batteries
Measurement range	5 – 180 kg	5 – 180 kg	5 – 180 kg	5 – 180 kg
Open API	+	+	+	+
Uses bioelectrical impedance analysis	-	+	-	+

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

TABLE 6: COMPARISON OF FOUR SMART SCALES: WITHINGS BODY, WITHINGS BODY CARDIO, FITBIT ARIA AIR AND FITBIT ARIA 2.

Sensed Parameters:

- Weight; BMI

Communication Technologies:

- Wi-Fi

Communication with SmartWork services/components:

- The Withings Body smart scale is integrated with R2D2.

Despite having selected a preferred smart scale device, the consortium should consider if its use is indeed justifiable. It is reasonable to assume that every household has a body scale. Therefore, the question is whether an automatic synchronization of the body weight outweighs the costs of the device, and burden to the user with installation of an additional device. An option to consider is to rely on the fact that every household has a body scale, and ask the user to manually input the weight on the dedicated SmartWork application, within the self-reported physiological parameters (see Section 3.4.1.1).

3.5.2. Assessment of Energy Efficiency

Smart scales are used on average less than once per day. From experience, the energy drain on the battery powered devices is negligible.

3.5.3. Component Architecture

In order to provide the complete picture of the SmartWork Sensing Architecture in Section 5, we provide for each of the components their simplified high-level architecture overview. Below in Figure 7, the high-level component architecture is given for the Withings Body sensor.

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

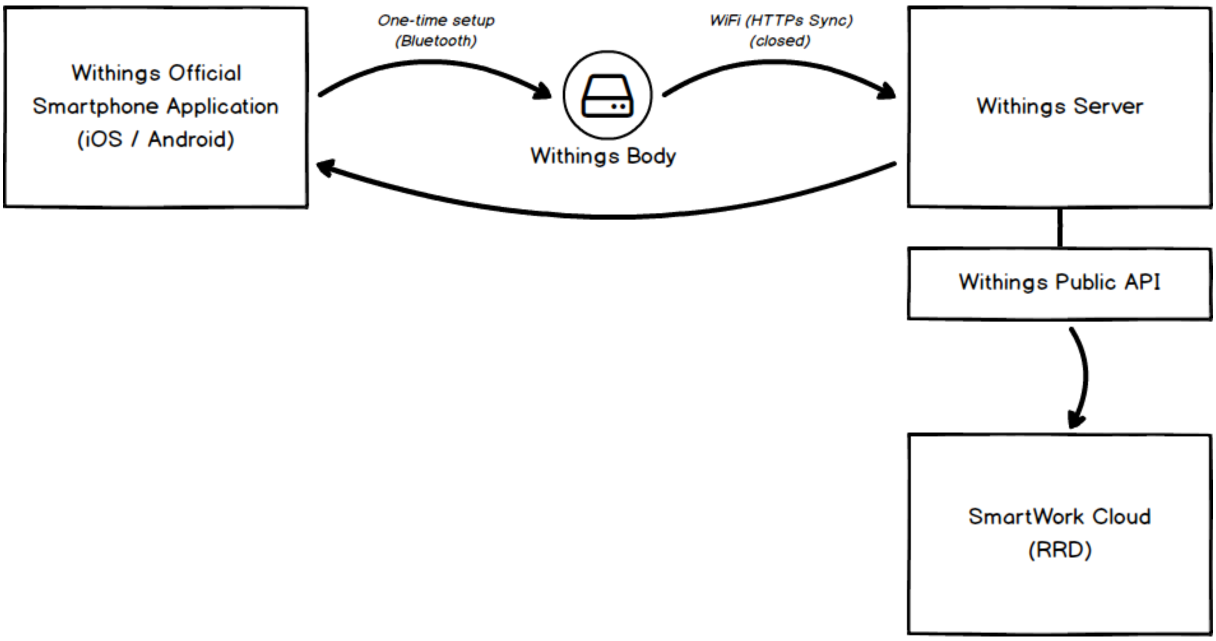


FIGURE 7: HIGH-LEVEL COMPONENT ARCHITECTURE FOR THE WITHINGS BODY SENSOR.

Funded by Horizon 2020 Framework
Programme of the European Union under
Grant Agreement No. 826343

3.6. SPARKS ECG

The SPARKS ECG Vest monitors the wearer's heart condition generating information about the beats per minute, PQRS and RR intervals. It is also capable of sensing notifications about the categorization of the beats observed when used in conjunction with an analysis algorithm on the wearer's smartphone. The vest needs to be worn under the user's clothes as it needs direct contact with the skin for the electrodes to work properly. The vest collects the electrode data locally and transmits them to the user's smartphone (through the SmartWork *healthyMe* application) in real-time. If needed the communication can be set in non-real time, so that the data are transmitted in batch once the vest is taken off. The duration of the vest's recoding can vary depending on what needs to be detected. Small sessions can be performed during physical work tasks or when a possible troubling condition is sensed. Longer sessions (in the so called, holter mode) can be used to detect more data if needed. In that case real-time data collection is not viable due to the high battery usage and the long duration of the recording, and post-recording synchronization is needed. Such sessions will not be performed for SmartWork as it exceeds the scope of the project and requires that the data are inspected by a medical doctor. Such further processing of the data from medical personnel is needed in order to extract any valid and safe medical conclusions in all cases.

3.6.1. Suitability in SmartWork Contexts of Use

The Sparks ECG device has been designed using the Intel Curie processor and architecture[5] following all CE and RoHS specifications. It offers us two main features that make it ideal for use in the SmartWork project:

1. It is an easy to wear and use device, as it is worn as a vest without the need for complex patches that need to be attached to the body;
2. It is a low power device, designed to operate with limited resources and communicates with a user's smartphone.

Sensed Parameters:

- Heartrate (BPM)
- RR-interval (in milliseconds)
- Full ECG (EDF trace file)
- Arrhythmia Events (event-list)

Communication Technologies:

- Bluetooth Low Energy

[5] https://software.intel.com/en-us/node/701261

3.6.2. Assessment of Energy Efficiency

The SPARKS ECG Vest was developed to be as power efficient as possible. It was tested in two scenarios regarding its power efficiency in data transmission and offline operation. Regarding the data transmission, the power consumption of the wearable device was measured while transmitting data to the Android device so that we can understand the battery that would suffice for multiple hours of operation while connected to a smartphone and transmitting data. In more detail we identified 4 different states for the operation of the BLE radio of the device:

- **Advertising:** The BLE radio of the ECG device is on and advertising its presence in all the other BLE devices in the vicinity. In this case the electrical current drawn by the device is *39.92 mA*.
- **Connected:** The BLE radio of the ECG device is on and maintaining a connection to a single smartphone. In this case the electrical current drawn by the device is *39.77 mA*.
- **Streaming Data (with Subscriber):** The BLE radio of the ECG device is on and streaming data to a single smartphone. In this case the electrical current drawn by the device is *40.51 mA*.
- **Streaming Data (no Subscriber):** The BLE radio of the ECG device is on and streaming data without a connected smartphone. In this case the electrical current drawn by the device is *39.91 mA*.

Regarding the operation of the wearable device itself, we identified the following three states:

- **Normal operation:** The wearable device is operating on a loop, processing and reading data from the sensors. In this case both cores are busy, and the power consumption is ta *50.2 mA*. The BLE radio is disabled in this case.
- **Deep sleep ARC, x86 busy:** The ARC processor of the wearable device is in deep sleep while the x86 processor is in a busy waiting loop. The power consumption is at *42.6 mA*. The BLE radio is disabled in this case.
- **Deep sleep ARC and x86:** The ARC and the x86 processor of the wearable device is in deep sleep. The power consumption is at *12.74 mA*. The BLE radio is disabled in this case as well. This is the lowest possible power consumption for the board that can be maintained in standby mode.

3.6.3. Component Architecture

In order to provide the complete picture of the SmartWork Sensing Architecture in Section 5, we provide for each of the components their simplified high-level architecture overview. Below in Figure 8, the high-level component architecture is given for the SPARKS ECG sensor.

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

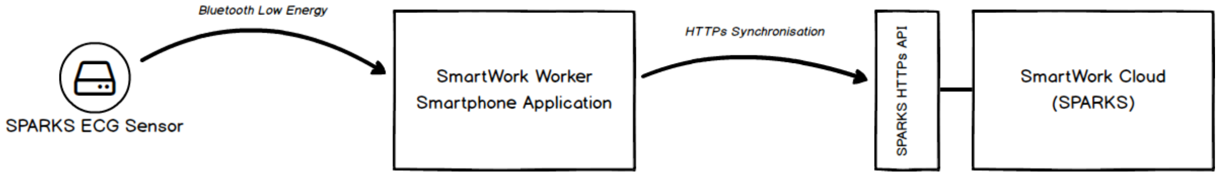


FIGURE 8: HIGH-LEVEL COMPONENT ARCHITECTURE FOR THE SPARKS ECG SENSOR.

3.7. SPARKS Environmental Sensor

An additional request from the SmartWork users interviewed was the need to monitor the environmental conditions of their workplaces. The options in this field are unlimited, due to the recent popularity of IoT and Smart Home gadgets. Although many devices exist, most of them only work with their manufacturer's application and rarely offer an open API to access their data in real time. Additionally, SparkWorks has already implemented similar solutions for use in other EU funded and internal projects and has experience with open and non-open-source solutions for real-time data collection in home, office or public building environments. As a result, we decided to build a simple device that can collect the required measurements, is simple to use and integrate with the rest of the SmartWork infrastructure, and can be easily extended if any requirements change in the course of the project (e.g., extend with another sensor). The SPARKS environmental sensor needs to be placed in the working area of the user. The SmartWork HealthyMe application simply connects to the device using Bluetooth Low Energy (BLE) and subscribes to receive notifications for the monitored parameters. The device collects data for temperature, relative humidity and VOC gases. The value of the VOC gases is an indication for changes in the level VOC concentration and not an absolute value, as this would require expensive in-lab calibration and is outside the scope of the project. The user's smartphone needs to be present in proximity to the device for data collection to occur. Every measurement received is uploaded to the SmartWork Cloud once received through the smartphone of the worker. Also, in cases where a single office is shared by multiple users, the same sensor device can be shared by all workers.

The device designed is based on the Espressif ESP32 processor[6] and is powered by any USB 5V wall charger. The sensor module used to collect the data is the Bosh BME680 sensor[7]. Both the processor and the sensor module are CE[8] and RoHS certified[9].

3.7.1. Suitability in SmartWork Contexts of Use

The Sparks Environmental Sensor module is a deesign that was adapted to fit exactly the needs of SmartWork project. As a result, it can be adapted and configured based on the needs of the project both in terms of sensors and operation. It is a device that can fill in gaps of information regarding the worker's environment that cannot be covered by any other single off-the-shelf device reducing the total number of devices needed. It is also a miniature device that requires a very small space in the worker's office and Is barely noticeable, fitting directly in the unobtrusive nature of the SmartWork project. The data of the device are also collected via Bluetooth Low Energy, a low

---

[6] https://www.espressif.com/en/products/hardware/esp32/overview
[7] https://www.bosch-sensortec.com/bst/products/all_products/bme680
[8] https://www.espressif.com/en/certificates?field_product_value%5B%5D=ESP32
[9] https://cdn.sparkfun.com/assets/a/3/5/0/4/BME680-Layout_Considerations.pdf

power communication technology that allows us to integrate it directly to the HealthyMe smartphone application that will handle all the data collection, authentication, and management issues, similarly to all other Internet of Things (IoT) devices that will be integrated to the project. Also, as SPARKS will develop and distribute the devices for the trials, the devices can be upgraded or retrofitted with more sensors in the duration of the project shall the need for this be detected by other partners.

Sensed Parameters:

- Temperature (centigrade)
- Relative Humidity (%RH)
- VOC levels (sensor Ohm resistance)

Communication Technologies:

- Bluetooth Low Energy

3.7.2. Assessment of Energy Efficiency

The Environmental Sensor module will always be directly connected to power up through a USB power supply. This power supply can come either from the USB wall charger or any USB port on the worker's computer. Its power requirements are minimal and can be supplied by all modern USB chargers with no issues. It communicates its data to the smartphone using a BLE connection, a low power communication protocol that most IoT devices use when they do not require a direct connection to the Internet. The processor used in the device, ESP32, is also a low power microcontroller that has become a kind of industry standard for IoT devices. It offers multiple communication interfaces and sensor inputs that can be switched on and off on demand, thus further reducing its power consumption. Especially when the microcontroller's sleep modes are used, its power consumption can be further reduced from 200mA in active mode to as low as 2.5μA when in full hibernation. A more detailed power consumption analysis of the developed application will be provided in future deliverables.

3.7.3. Component Architecture

In order to provide the complete picture of the SmartWork Sensing Architecture in Section 5, we provide for each of the components their simplified high-level architecture overview. Below in Figure 9, the high-level component architecture is given for the SPARKS ECG sensor.

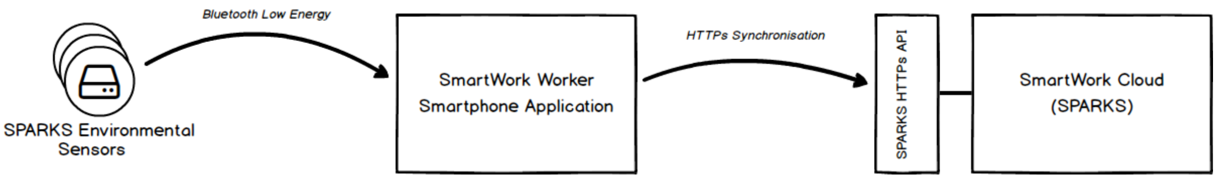


FIGURE 9: HIGH-LEVEL COMPONENT ARCHITECTURE FOR THE SPARKS ENVIRONMENTAL SENSOR(S).

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

3.8. Posture Trackers

The relevance of addressing posture in the workplace (and in living environments) has been supported by several works related with ergonomics and treatment of musculoskeletal disorders (MSDs).

3.8.1. Suitability in SmartWork Contexts of Use

Based on work performed in the Therapeutic Exercise at Workspace (THEWS) Erasmus+ project[1], IPN has learned valuable lessons from physical therapists, physical exercise experts and spine surgeons. Musculoskeletal disorders such as neck/back pain or upper limb disorders remain the most common occupational disease in the European Union and workers in all sectors and occupations can be affected. They are also an increasing problem and one of the most important causes of long-term sickness absence. Beside the effects on workers themselves, MSDs may lead to high costs to enterprises and the society as a whole and is related with high socioeconomic consequences in terms of health expenses and loss of working days. While sitting, office workers tend to lean forward or to slouch in the chair. This partial immobilization can cause low back pain or neck pain because static posture increases stress on the back, neck, shoulders, arms and legs. In particular, sitting can add large amounts of pressure to the back muscles, spinal discs and ligaments. Therefore, strength and conditioning professionals, athletic trainers, and physical therapists often come across ongoing or recurrent complaints of neck pain. While the etiology of musculoskeletal pain symptoms is multi-factorial, there is a general consensus about the beneficial effects of therapeutic exercise.

Additionally, IPN is further investigating vision-based approaches for pose estimation that will be used in the CogniViTra AAL project, where such technology will be applied for gesture recognition in a cognitive and physical training system.

Therefore, we identified the possible synergy of these projects (THEWS, CogniViTra and SmartWork) that can benefit from mutual exchange of knowledge and technology.

In that sense, IPN identified an optional (i.e. experimental feature) for SmartWork that could adopt the pose estimation approach that is being developed in CogniViTra and to be adapted for the workplace environment.

However, it must be clear that the development of such technology is out of scope of SmartWork. Any integration that might occur during the project will be a direct result of the synergy between projects and the development of such integration will be owned by IPN.

Sensed Parameters:

- Estimated Pose (positions of joints and links of the human skeleton).

Communication Technologies:

- Serial bus communication;
- Integration with other software components through message bus (e.g. RabbitMQ).

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

3.8.2. Assessment of Energy Efficiency

Not relevant aspect for this component because the sensor used will be either a built-in web camera or a USB 3.0 connected visual, infrared and depth sensor.

3.8.3. Component Architecture

To be determined.

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

3.9. Computer Activity Tracker

The user's activity when working with a computer (laptop or desktop) will be tracked using not only physiological data, but also behavioural data. More specifically, the user's eye gaze will be tracked. The intent is to match observed eye gaze with the active window on the computer screen, in order to assess in which task the user is presently engaged in.

3.9.1. Suitability in SmartWork Contexts of Use

Eye gaze tracking and active window tracking will be used in the Cognitive State Estimation module (CSE) in SmartWork for training of machine learning models, as a complement to other physiological and activity tracking data. The models will be used for assessing when the user is focusing on a computer task vs. has been distracted, when the user is mentally fatigued, stressed, or mentally overloaded.

Not all of the above data types have a direct bearing on all cognitive state estimations. For example, eye data (together with mouse data) can be expected to be more central for estimating task focus, while physiological measures are more indicative of task-related stress. A key component in machine learning is to determine the strength of the correspondence between various indata types and the desired estimates.

3.9.1.1. Computer Activity Tracking through GPII "Windows Metrics"

The GPII module of the Morphic system developed by Raising the Floor International (RtF-I) provides the ability to capture certain ways in which the system is being used. The data is written to a log file and uploaded to a log server via an external process, using Filebeat.

The main purpose is to be able to record the impact that GPII has on how the user uses their computer, while making sure that no personally identifiable data is captured.

A large number of different system-use variables are used. Two examples below are taken from the Windows Metrics documentation available through GitHub[10]. In Example 1 below, an example is given how the version of the gpii-windows, gpii-universal, gpii-app, and windowsMetrics modules are logged on startup. In Example 2 below, an example is given for system information that is logged on startup.

```
{
    "module": "metrics",
    "event": "version",
```

[10] https://github.com/GPII/windows/blob/master/gpii/node_modules/windowsMetrics/README.md

```
    "level": "INFO",
    "data": {
        "windowsMetrics": "0.3.0",
        "gpii-app": "0.3.0-dev.20180315T174453Z.02de41c",
        "gpii-windows": "0.3.0-dev.20180315T174453Z.02de41c",
        "gpii-universal": "0.3.0-dev.20180905T001251Z.a9b24952"
    }
}
```


EXAMPLE 1: GPII WINDOWS METRICS EXAMPLE FOR VERSIONING INFORMATION.


```
{
  "module": "metrics",
  "event": "system-info",
  "level": "INFO",
  "data": {
    "cpu": "Intel(R) Core(TM) i7-4770 CPU @ 3.40GHz",
    "cores": 2,
    "memory": "2.0",
    "resolution": "1280x951",
    "scale": "1.25",
    "osRelease": "10.0.17134",
    "osEdition": "Windows 10 Enterprise Evaluation",
    "osBits": "64",
    "systemMfr": "innotek GmbH",
    "systemName": "VirtualBox"
  }
}
```


EXAMPLE 2: GPII WINDOWS METRICS EXAMPLE FOR SYSTEM INFORMATION.

Sensed Information (GPII Windows Metrics):

- GPII Versions
- System Information
- Live configuration
    - Resolution
    - SystemParametersInfo
    - Other
- Application Launch
- Application Close
- Window Activation
- Inactivity
- Typing (Including e.g. SHIFT / ALT Combinations)
- Typing rate (including corrections)
- Mouse usage (used or not)

Communication Technologies:

N/A

3.9.1.2. Active Window Tracking Script

An alternative to tracking of which window on the user's desktop or laptop is currently active will be based on a software developed by Kalle Hålldén[11]. The software was originally written to help the users track where (in which applications, and on which websites) they spent their time. Among the dependencies is Selenium, which is licensed under Apache 2.0. The active window tracking software is available on Github, and we have obtained the author's permission to use the software under the MIT license.

3.9.1.3. Eye Tracking

The software approximates eye gaze direction and assesses blinks from low-resolution webcam input. The software is built on Open Computer Vision (Open CV) and dlib library[12] for real-time detecting (recognizing and locating) 68 predefined landmarks on the human face – twelve of these landmarks

[11] Active window tracking: https://github.com/KalleHallden/AutoTimer
[12] dlib models for face landmark detection: https://github.com/davisking/dlib-models

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

mark the position of the two eyes in the image frame. Face landmark detection uses a model that has been trained on the 300-W faces dataset[13].

The software for eye gaze tracking that we will be adapting, can be considered a derivative (via dlib's pretrained model for face recognition) of the 300-W faces dataset. The license for this dataset excludes commercial use. However, the software for eye tracking has been released as open source, and is available on Github[14], under the MIT license.

Eye gaze is approximated by calculating the horizontal ratio between the location of the iris and the width of the eye. The resulting ratio is bound to [0 1], with 0 meaning extreme right, 0.5 straight ahead, and 1 meaning extreme left. This ratio is obtained along both the horizontal and vertical axes, where vertical gaze coordinates 0 denotes that the gaze is directed to the top, 0.5 straight ahead, and 1 to the bottom.

Currently, eye-centred gaze coordinates are translated into approximate screen-centred locations, by thresholding the above coordinates, using 0.35, and 0.65 as thresholds for right and left gaze direction.

Sources of error include the low image resolution of the webcam input, and the assumption that the pupil always lays at the centre of the iris from the camera's point of view, irrespective of gaze direction. In other words, the algorithm is strictly 2D, and does not use a 3D model of the eye globe.

Perclos (percent of the eyes covered by the eyelid) is calculated by taking the ratio of the width of the eye and its height. The user is assumed to be blinking if this ratio is greater than 3.8. Sources of error include cases when the users might be squinting, or have their head turned downwards or upwards.

3.9.2. Assessment of Energy Efficiency

Computer Activity Tracking is software that runs on the worker's desktop or laptop device. It is very unlikely that energy drain caused by such software is significant in any way.

3.9.3. Component Architecture

In order to provide the complete picture of the SmartWork Sensing Architecture in Section 5, we provide for each of the components their simplified high-level architecture overview. Below in Figure 10, the high-level component architecture is given for the separate Computer Activity Tracking components.

[13] The 300-W faces dataset: https://ibug.doc.ic.ac.uk/resources/facial-point-annotations/
[14] The eye gaze tracker: https://github.com/antoinelame/GazeTracking

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

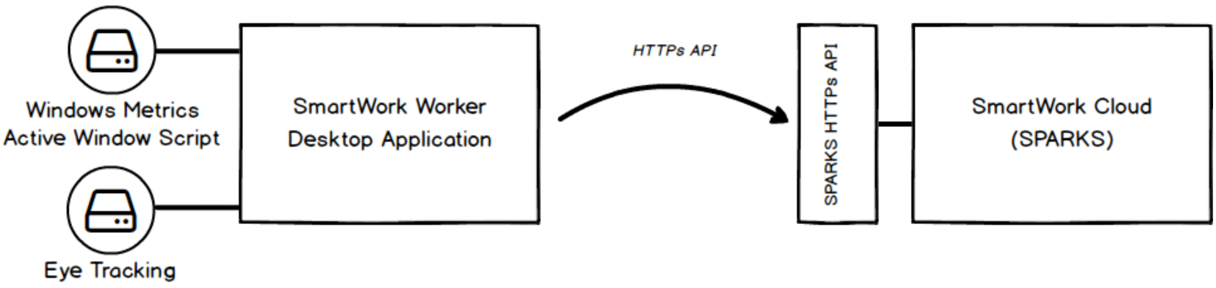


FIGURE 10: HIGH-LEVEL ARCHITECTURE OVERVIEW FOR THE COMPUTER ACTIVITY TRACKING COMPONENTS.

Funded by Horizon 2020 Framework
Programme of the European Union under
Grant Agreement No. 826343

3.10. Workplace Indoor Location Monitor

Within a living space equipped with different kind of sensors, the user's actions and behaviours can provide useful information about their needs. As a lot of these needs are directly related to the location of the user, knowing the users location in a system is the problem that this technology is covering.

3.10.1. Suitability in SmartWork Contexts of Use

Such technology can be used for several scenarios, from helping elderly people with positioning awareness problems and enabling some caregivers to overview the activities and problems of those users, to providing an easy way to automatic check-in and check-out of workplace and other living areas.

For the development of this solution, IPN has developed an indoor positioning system that uses Wi-Fi information, which is widely deployed in households and workplaces and devices are available at cheap prices.

This technology can be integrated and extended in the SmartWork ecosystem as the current solution was developed as an application for mobile phones that uses Android OS.

The overall localization system uses a client-server approach, where the localization algorithms logic is on the server-side. A general overview of the system's architecture can be seen in Figure 11 (Quintas, et al., 2013).

FIGURE 11: DRAFT system architecture overview of the indoor location monitoring system proposed by IPN.

Sensed Parameters:

- Estimation of Indoor Location

Communication Technologies:

- Wi-Fi
- Integration via SOAP web-services API

3.10.2. Assessment of Energy Efficiency

To be determined.

3.10.3. Component Architecture

To be determined.

Funded by Horizon 2020 Framework
Programme of the European Union under
Grant Agreement No. 826343

4. In-Lab Data Collection Cohort

In order to bootstrap the machine learning training processes in WP4, the project is starting early with collecting realistic datasets, while not burdening potential end-users with technically complicated processes. From the measuring devices listed in the previous section (3) we aim to gradually include these into prototype data collection applications that can be used by researchers and colleagues within the project.

Consortium members may volunteer to participate in this in-lab data collection cohort to provide any or all of the available data collection resources. At this point in time, three different data collection resources are available: the Activity Coach application (see Section 4.1), the SmartWork Data Collector (see Section 4.2) and the Smart Mouse (see Section 4.3). The use of these three data collectors are described in the subsections below.

4.1. Manual installation Activity Coach

Download and Install Activity Coach application

1. Download the Activity Coach application from the Google Playstore: https://play.google.com/store/apps/details?id=nl.rrd.activitycoach.alarm.clock.inc&hl=en
2. Install on your Android device.

Create new account

3. Select CREATE ACCOUNT option (image)
4. Read the license agreement, and if applicable, proceed by selecting AGREE.
5. Provide your email address.
6. Provide a password
7. Repeat the password

Join the SmartWork project

8. Send an email with your email to m.cabrita@rrd.nl asking to add you to the SmartWork project. Our developers will assign your account to the SmartWork project.

Link Fitbit and Activity Coach account:

9. Open the menu (upper left corner) and select “Step counting”.
10. Choose “Fitbit” and confirm by selecting “SET UP FITBIT”.
11. You will see information on how to link the Fitbit and Activity Coach accounts. Select “NEXT” after reading the information
12. If applicable, provide your consent to data collection by selecting “AGREE”.

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

13. You will be automatically directed to the Fitbit login page via your preferred web-browser. Provide your Fitbit log in details.
14. After logging in you will be redirected again to the Activity Coach application.
15. Click OK to conclude procedure. Your current number of steps should now be visible on the main Activity Coach page.

Link Withings and Activity Coach Account.

16. Open the menu (upper left corner) and select “Body weight”.
17. Choose “Fitbit” and confirm by selecting “SET UP YOUR SCALE”.
18. You will see information on how to link the Withings and Activity Coach accounts. Select “NEXT” after reading the information
19. If applicable, provide your consent to data collection by selecting “AGREE”.
20. You will be automatically directed to the Fitbit login page via your preferred web-browser. Provide your Withings log in details.
21. After logging in you will be redirected again to the Activity Coach application.
22. Click OK to conclude procedure. Your latest measurement should now be visible on the WEIGHT tab of the main page.

4.2. Manual installation SmartWork Data Collector

Create a new SmartWork account

1. Visit the SmartWork SSO Registration page
2. Fill in a username and password for the account
3. Select INLAB your trial site location
4. Select the EMPLOYEE role
5. Click the register button

Download and Install the SmartWork Data Collector application

6. Download the SmartWork Data Collector application from the Google Playstore
7. Install on your Android device.
8. Launch the application and login with your account from the previous step
9. Allow the application to access Location, Bluetooth and the phone’s microphone.

Link the Sparks Environmental device:

10. Open the application’s menu on the left of the screen
11. Select the “Environmental Sensor” option in the menu

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

12. Select the "Start Scan" button
13. Wait until the Sparks Environmental Sensor appears and click "Connect"

Link the Sparks ECG device:

14. Open the application's menu on the left of the screen
15. Select the "ECG Sensor" option in the menu
16. Select the "Start Scan" button
17. Wait until the Sparks ECG appears and click "Connect"

Add Home and Work Locations:

18. Open the application's menu on the left of the screen
19. Select the "ECG Sensor" option in the menu
20. A map view with your current location will appear in the application.
21. Navigate to your Home/Work location and click on the map.
22. A menu will appear with 3 options. Home, Work and Cancel. Select the title for the location.
23. Repeat this process with as many locations you want to monitor.

Background Monitoring:

24. Leave the application running in the background for the measurements to be collected and transferred to the project's cloud services.

4.3. Manual installation Smart Mouse

The following installation procedures described are related with the working application available up to the date of this deliverable. Since the mouse will be upgraded, a new version of the application will be developed, and the installation procedures may differ.

Download the needed files

1. Log in into SmartWork SharePoint/Teams
2. Navigate to "WP3 - Unobtrusive Sensing and Low-Level Processing"
3. Download the zip file called "SmartWork_IntelliMouse_tray.zip" to your computer

Prepare the application and start-up

4. Extract the previous zip file to any location in your computer that do not have any permissions restriction. The root of hard-drive is the recommended directory.
5. Open the directory where the files were extracted to. You should see a list of files similar to the one displayed in the next figure:

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

Logs	18/10/2019 16:05	Pasta de ficheiros	
configData	28/10/2019 10:48	Documento XML	1 KB
Microsoft.Diagnostics.Tracing.EventSour...	01/12/2015 14:31	Extensão da aplica...	167 KB
Microsoft.Diagnostics.Tracing.EventSource	01/12/2015 14:31	Documento XML	199 KB
mouseIcon.ico	07/10/2019 17:23	Icon File	24 KB
MySql.Data.dll	17/02/2015 18:47	Extensão da aplica...	447 KB
Newtonsoft.Json.dll	13/05/2019 10:19	Extensão da aplica...	487 KB
Newtonsoft.Json	13/05/2019 10:19	Documento XML	472 KB
RabbitMQ.Client.dll	09/09/2019 08:32	Extensão da aplica...	272 KB
RabbitMQ.Client	09/09/2019 08:32	Documento XML	344 KB
SmartWork_IntelliMouse	28/10/2019 10:48	Aplicação	90 KB
SmartWork_IntelliMouse.exe.config	03/10/2019 15:36	XML Configuratio...	1 KB
USBHIDDRIVER.dll	13/05/2019 10:19	Extensão da aplica...	28 KB
USBHIDDRIVER.pdb	13/05/2019 10:19	Program Debug D...	40 KB
USBHIDDRIVER	13/05/2019 10:19	Documento XML	11 KB

FIGURE 12 - EXPECTED FILE LIST WITHIN THE SPECIFIED DIRECTORY

6. Connect the IPN Mouse to the computer.
7. Ensure that the mouse was properly recognized by the Operating System and the drivers were installed
8. Launch the application by clicking in the file with blue logo called "SmartWork_IntelliMouse".
9. A console window should open, and the application should start without any errors. If so, you should see a window like:

```
[Application]
SmartWork Mouse Detected!

[RabbitMQ Test]
Configuration File Detected. Loading Settings..
Testing RMQ connection with the following parameters:

hostname= localhost
port= 5672
username= guest
password= guest
virtualhost= /

[Application]
Configuration File Detected. Loading Application Settings..
[Message Broker]
Sucessfull Connection with server
[Application]
Thread created to push mouse data to server
C:\Dados\GitLab\smartwork-tray_app\bin\Release\Logs
[Application]
Events created to manage mouse
[Application]
Thread created to process mouse incoming data
[Application]
Application has started!
```

FIGURE 13 - NORMAL APPLICATION START-UP

10. At this point, the application should be running and sending the acquired data to the configured message broker. For better productivity and comfort, the application can be minimized to the tray, running in background. To do that, go to the tray-bar and locate the application logo, similar to the next figure:

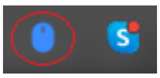


FIGURE 14 - APPLICATION LOGO, SIGNALLED AT RED

11. Right-click the icon and a list of options should appear. Press "Hide App" to minimize the application to tray. At this stage, the console window should disappear and the application will keep running in background.

Closing the application:

Option 1 (Silent Shutdown)

12. Locate again the application icon in the tray-bar and right-click it. A list of options will pop-up.
13. Press the option "Exit".
14. Wait a moment until the icon disappears. The application should now be closed.

Option 2 (Verbose Shutdown)

12 Locate again the application icon in the tray-bar and right-click it. A list of options will pop-up.

13 Press the option "Maximize App"

14 The console windows should now appear. If you get a window similar to the one in the next figure, double click in the window header to maximize it and everything should pop-up.

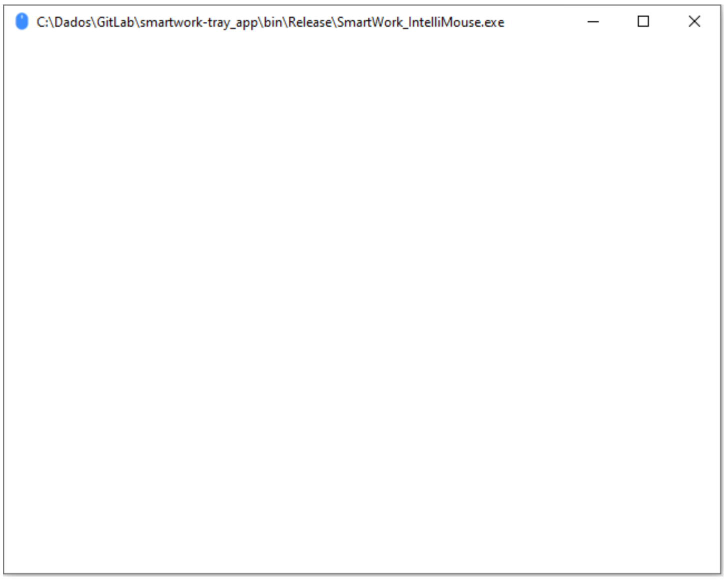
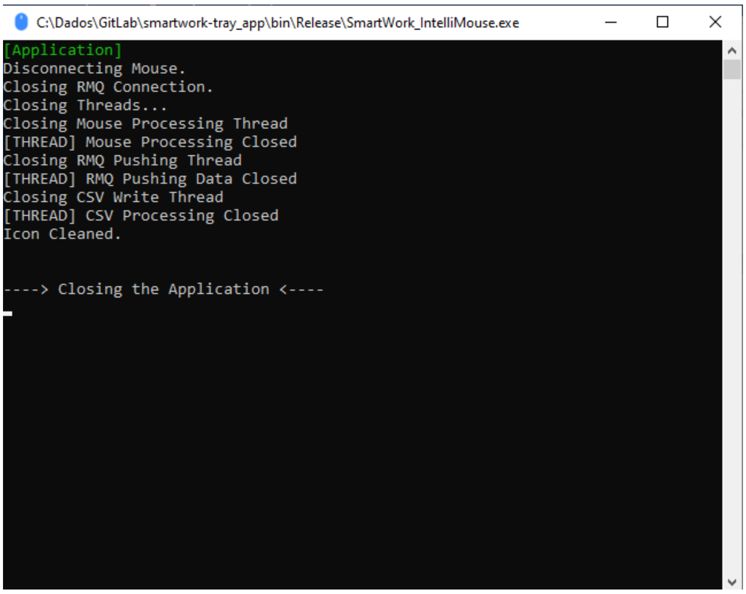


FIGURE 15 - A BLANK WINDOW MAY APPEAR IN SOME SITUATIONS WHEN THE "MAXIMIZE APP" IS REQUESTED

15 Press CTRL+C keys in keyboard to trigger the application close procedure. You should see a list of processed being terminated after the key press. The next figure illustrates that:

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343



16 After the procedures have completed, the window will close automatically. The application is now closed.

5. Sensing Architecture

In Section 3, an overview was given on all the various sensors that serve as input to the SmartWork ecosystem of applications and services. For each sensor, a high-level architectural overview was given on how the sensor data is transferred, either directly or through multiple "hops", to the SmartWork Cloud. In Figure 16 below, these separate sensor-specific architectures have been combined to display the overall Sensor Network Architecture.

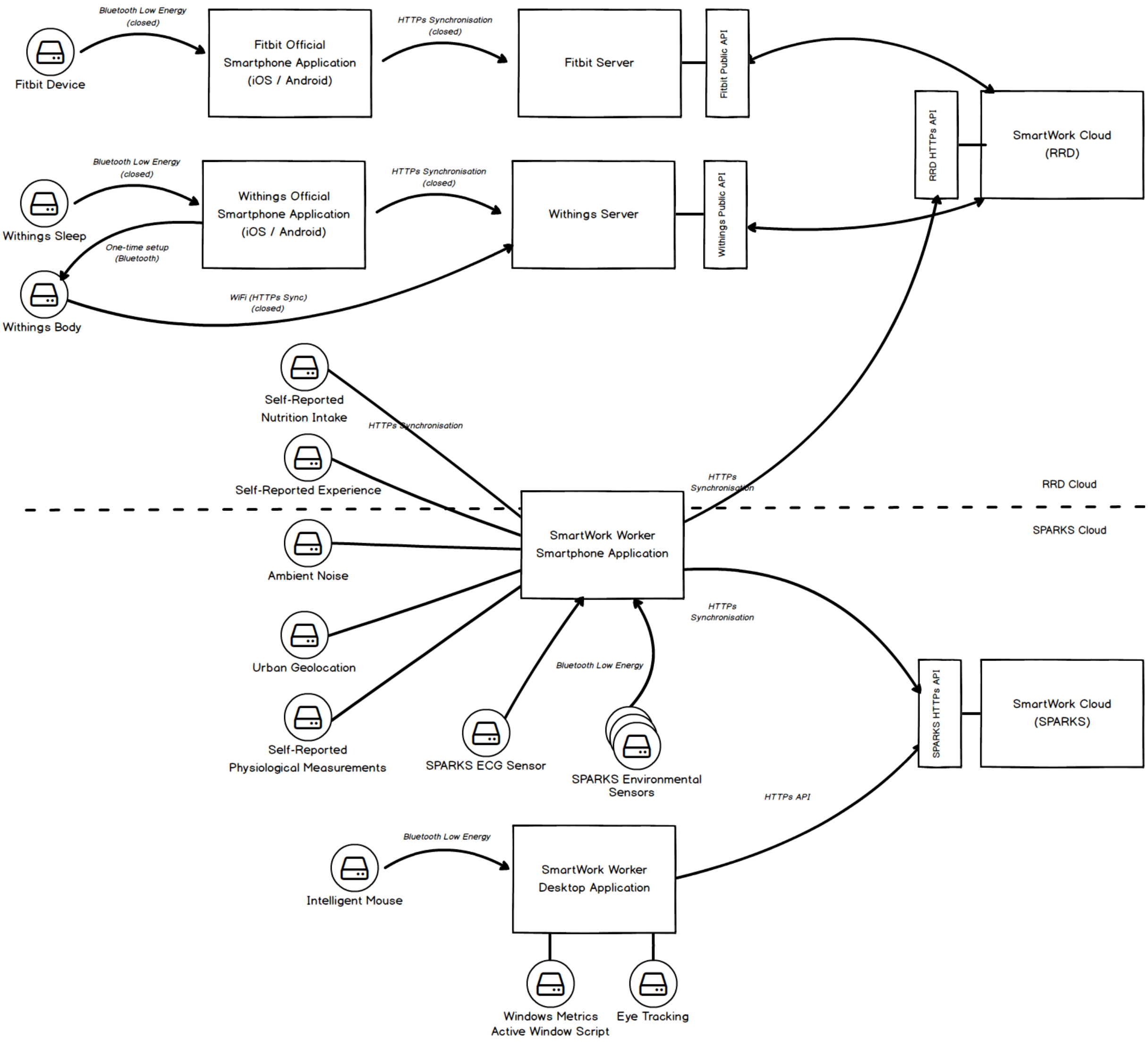


FIGURE 16: OVERALL HIGH-LEVEL SMARTWORK SENSOR NETWORK ARCHITECTURE.

All the various sensor devices are depicted as circles. For some sensors their data is transmitted wirelessly to a receiving application or server component (e.g. the Fitbit Device transmits wirelessly over Bluetooth to the Fitbit Official Smartphone Application). These connections are depicted as arrows, marked up by the type of transmission (e.g. WiFi, or Bluetooth) in Figure 16. For other sensors, the connection to a software component is not wireless, but through some type of embedding. For example, the SmartWork Worker Smartphone Application has *embedded* into the application 5 different "sensors" – the Self-Reported Nutrition Intake, Self-Reported Experience, Ambient Noise, Urban Geolocation and Self-Reported Physiological Measurements sensors. Such *embedded connections* are depicted by connecting lines in Figure 16.

Finally, as can be seen in Figure 16, there are two *destinations* for the data in the SmartWork Cloud – one part depicted as the SPARKS Cloud, and one part as the RRD Cloud. These two existing cloud infrastructures are used simultaneously as they each have their strengths and weaknesses regarding data storage and access.

Access from SmartWork applications to the data will be handled through one or both of these SmartWork Cloud components – but using a single authentication scheme, and using similar APIs.

5.1. RRD data collection

The RRD component of the SmartWork cloud is used primarily because of existing connections with the Fitbit Cloud API and a partially existing connection with the Withings Cloud API. Also, a smartphone application exists that supports experience sampling, reporting to this RRD Cloud. Features such as the Self-Reported Nutritional Intake as well as support for iOS are scheduled to be delivered within the context of the SmartWork Project.

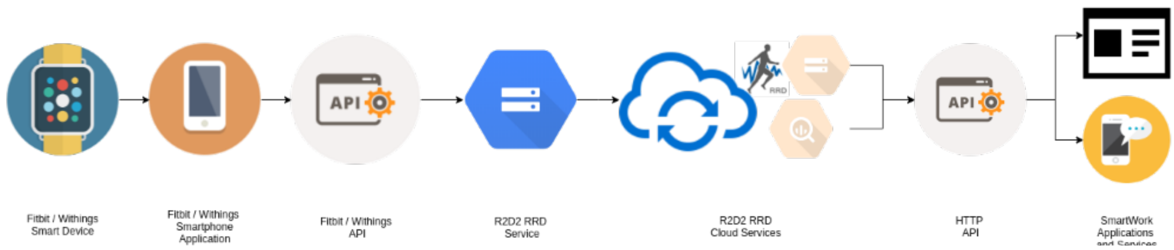


FIGURE 17: EXAMPLE OF DATA FLOW THROUGH THE RRD SMARTWORK CLOUD.

The API associated with the RRD Cloud component as depicted in Figure 16 above serves as a data sync for client applications (to send data to the cloud), but is also used to provide access to client applications for data retrieval.

5.2. SPARKS data collection

SPARKS offers all data producers of SmartWork with a data analytics, storage and sharing solution. The sensing devices and their accompanying software modules can send time-series data to SPARKS through multiple APIs based on their capabilities that are then processed and stored accordingly or also distributed to other SmartWork applications that require them for their operation.

The first connection that needs to be described is the data collection performed on the user's smartphone via the SmartWork Worker Smartphone Application. This flow contains both self-reported measurements and data collected by the on-board smartphone sensors. All data reported or collected are communicated to SPARKS using the Sparks HTTP API and the user's credentials. All

Funded by Horizon 2020 Framework
Programme of the European Union under
Grant Agreement No. 826343

data reported are stored under the specific user's identity and are then distributed to all other SmartWork applications that need them (e.g. AI and ML modules) – see also Figure 18 below.

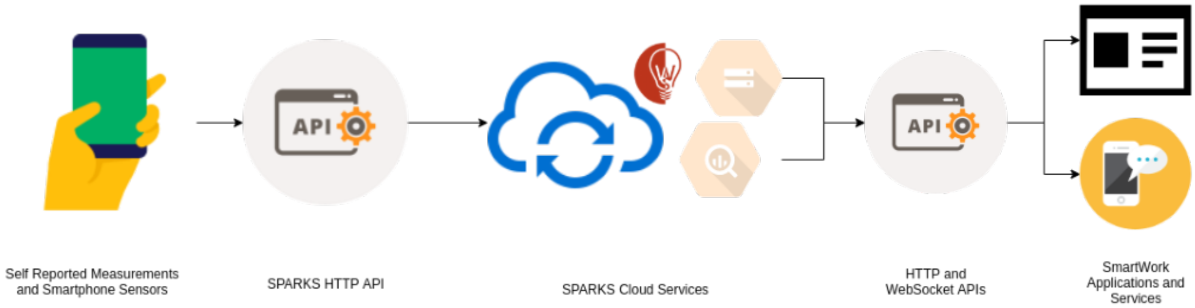


FIGURE 18: EXAMPLE OF DATA FLOW FROM SMARTPHONE APPLICATION TO SPARKS CLOUD.

A similar flow happens for the sensors and IoT devices that need the smartphone application to report their sensor data. An extra step is added in the beginning of the flow that allows for collecting the data from them to the phone. After the data are on the phone, the same flow as before is followed, through the Sparks HTTP API and using the user's credentials. All data reported are stored under the specific user's identity and are then distributed to all other SmartWork applications that need them (e.g. AI and ML modules) – see also Figure 19 below.

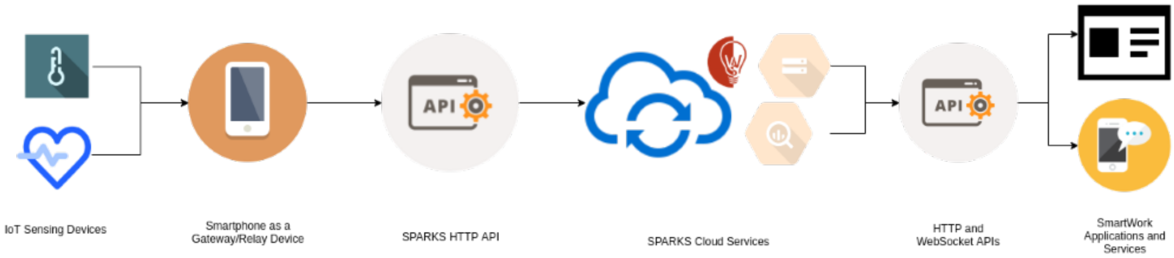


FIGURE 19: EXAMPLE OF DATA FLOW FROM IOT SENSING DEVICES TO SPARKS CLOUD.

Another interesting flow is the one followed to report data from the IPN Mouse to the SPARKS platform (see Figure 20). For this case, and due to the higher volume of data reported (almost 10 sensors with a 30Hz sampling rate) the integration is done via a RabbitMQ server and the AMQP protocol. In fact, two separate RabbitMQ servers are used, one under the SmartWork domain, and a second one provided by SPARKS. The first one is the original broker that the driver of the mouse uses to report the data of the mouse to SmartWork. The data there, comes in the internal format of IPN and need to be transformed so that they can be parsed by SPARKS. As a result, a service called a mapper is used to perform this transformation and report them to SPARKS in the appropriate format. This is a method used in other projects as well, when a proprietary protocol or data format needs to be integrated with the SPARKS platform. Once the data are appropriately formatted, they can be sent to SPARKS again using the AMQP protocol for storage under the specific user's identity and are distribution to all other SmartWork applications that need them (e.g., AI and ML modules).

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

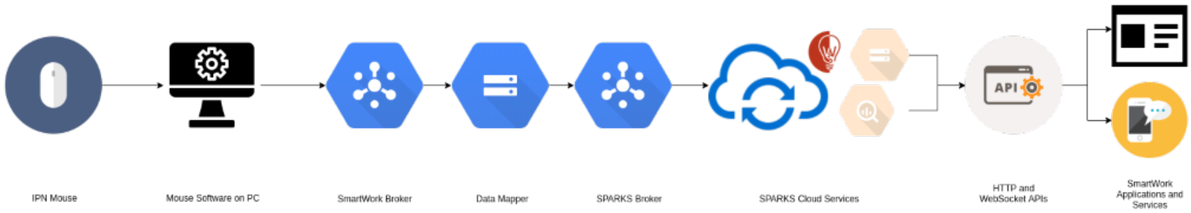


FIGURE 20: EXAMPLE OF DATA FLOW USING RABBITMQ FROM INTELLIGENT MOUSE TO SPARKS CLOUD.

6. Conclusion

Based on the thorough state of the art review presented in Deliverable 2.1: "*State of the Art (SOA) review and benchmarking of best practices*", this documented presents important decisions on the selection of sensor devices to be used in the SmartWork project (Section 3). Then, in order to get valuable insights in the day-to-day use of the sensor devices, this document presents the setup of the In-Lab Data Collection study, that also serves to generate the necessary data for the machine learning and modelling processes in Work Package 4 (Section 4). Finally, this document presents the sensor network architecture that is designed to be easy to use in a multi-party collaborative environment such as this European research project, facilitating easy development of front-end applications that will use the sensor data to provide value to SmartWork's end-users (Section 5).

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343

7. Bibliography

Quintas, J., Cunha, A., Serra, P., Pereira, A., Marques, B., & Dias, J. (2013). Indoor localization and tracking using 802.11 networds and smartphones. In *International Competition on Evaluating AAL Systems through Competitive Benchmarking.* Berlin: Springer.

Funded by Horizon 2020 Framework Programme of the European Union under Grant Agreement No. 826343