

# A Flexible Architecture for Mobile Health Monitoring

Mathieu Bagot, Pascale Launay, Frédéric Guidec

IRISA, CominLabs, Université Bretagne Sud

There is a growing need for systems that allow to monitor continuously the health condition of patients with chronic diseases, while allowing these patients to live their daily life as usual, at home as well as out of home. Developing such systems is now feasible based on currently available wireless transmission technologies and off-the-shelf wearable sensors, but most of the applications developed so far fall into the quantified-self movement, and can hardly be used for medical monitoring.

This paper presents a general architecture for mobile biophysical monitoring, covering all stages of data acquisition, transmission, and processing. This architecture has been designed so as to meet the expectations of the medical field (especially regarding confidentiality and dependability), while remaining open and flexible (i.e., new types of sensors or data processing algorithms can be incorporated as and when needed).

***Index Terms*—sensors, mobile, health, monitoring, m-health**

## I. INTRODUCTION

The European population is turning grey [1]. This is the consequence of the baby boom observed in Europe after World War II, combined with a rise of life expectancy, which over the last 50 years has increased by about 10 years for men and women alike. An aging population implies increased costs in healthcare expenditures, and special attention must be paid to preserve social models. Electronic health (e-health), which combines healthcare with Information and Communication Technologies (ICT), has already brought in solutions to increase healthcare efficiency, for example by avoiding redundant diagnostics, and by improving communication between all members of the medical personnel. New wireless transmission technologies and the miniaturization of electronic components now open up new perspectives, as the availability of a large variety of wearable sensors paves the way to mobile health (m-health).

During the last decade, Quantified-Self (QS) has emerged as an approach that uses technology to pro-actively acquire biological or environmental information about different aspects of a person's daily life, thus allowing this person to better understand his/her daily functioning. Dozens of QS applications (a.k.a. self-tracking apps) are now available for smartphones and tablets, and many of these applications can gather data from wearable monitors such as smartwatches, digital armbands, or heart rate monitors. The data thus acquired

are usually not kept on the smartphone or tablet: they are transferred to a remote site in the cloud, where they can then be stored indefinitely, and kept available for authorized users.

Although such applications are getting more and more popular, they can hardly be used in the medical field for a number of reasons. First, most of the sensors used in QS applications have never been certified for medical use [2]. Second, each model of sensor is usually distributed with its own dedicated application, which cannot gather data from any other kind of sensor. Collecting data simultaneously from different kinds of sensors —possibly produced by different vendors— is therefore not possible. Third, most applications are hard-coded so they can only transfer data to a predefined remote site. Users must therefore inherently trust the owner of this site when storing their data there, assume these data will be safe there, and most notably that they will not be disclosed or used without their consent. Finally, QS applications usually do very little processing of data, and simply allow to display nice curves or basic statistics based on these data.

In order to meet the higher expectations of the medical field, a health monitoring system based on wearable sensors and applications should be designed so as to meet the following requirements:

- **Versatility:** the system should be able to accommodate a large variety of sensors (ideally, certified ones), and run with heterogeneous sensors simultaneously;
- **Extensibility:** the system should be extensible, so new types of sensors or new processing algorithms can be plugged in whenever needed;
- **Confidentiality:** the system should be designed so as to prevent the fraudulent disclosure of data. This implies that all data collected by the system should be stored and/or transmitted in encrypted form, using state-of-the-art encryption algorithms (which may be replaced or adapted over time). Besides, when data must be transmitted to a remote site, this site should be selectable by the user (typically the data center of a hospital or medical institution);
- **Dependability:** the system should be usable anywhere and at any time. This implies that networking disruptions should be tolerated (without data loss or data alteration), when the patient stands or moves in “white areas” where no wireless coverage is available. Besides, the system should have sufficient autonomy to let the patient do his/her daily activities without having to recharge batteries too often.

Project SHERPAM<sup>1</sup> aims to provide a flexible system that meets all the above-mentioned requirements. The general objective of this project is to provide a solution to monitor continuously the health state of patients during their daily activities, while detecting anomalies automatically, and reacting accordingly. Each patient shall be equipped with a monitoring kit that is composed of a smartphone and wearable sensors. Since distinct pathologies may require different types of data, the monitoring kit is not limited to a pre-defined set of sensors. Instead it can be extended so as to be adapted to each patient's needs. The smartphone both serves as a controller for the sensors, and as a gateway since the data it collects from the sensors can be processed locally before being transmitted to a remote site for further analysis. Three application domains are considered as prime objectives in project SHERPAM: the monitoring of heart failure patients, the evaluation of the functional limitations of arteriopathic patients, and the recognition and quantization of the physical activity of patients. Yet, because of its flexibility the system designed in this project could be used to monitor other kinds of diseases.

This paper presents the software framework that has been designed and implemented in the context of project SHERPAM. Related work is discussed in Section II. Section III presents an overview of SHERPAM's system architecture, and it shows how this system is meant to meet the requirements listed above. Since the autonomy of the system is one of the keys to its usability, Section IV presents the results of preliminary experiments that have been conducted to evaluate the power consumption induced by several wireless technologies the system can rely on. In Section V we focus on the requirements about versatility and extensibility, and present a plugin mechanism that has been designed so the smartphone application can be extended with sensing or processing modules as and when needed. Section VI concludes this paper, and enumerates directions for future work.

## II. RELATED WORK

IT companies have largely entered the Quantified-Self (QS) market by providing dedicated applications (RunKeeper<sup>TM</sup>, MoodScope<sup>TM</sup>, etc.), platforms (Apple Healthkit<sup>TM</sup>, Google Fit<sup>TM</sup>, Samsung Artik<sup>TM</sup>, etc.), and hardware (Withings<sup>TM</sup>, Jawbone<sup>TM</sup>, etc.). The products they propose are either oriented toward fitness and wellness, or toward healthcare. Fitness and wellness solutions focus on sports training, diet planning, sleep monitoring, or more generally life logging, while healthcare solutions propose new services for both patients and physicians.

The study in [3] makes four interesting observations about these solutions, and especially about their usability for clinicians.

*Data capture:* Reliability of the data collected is crucial to establish a diagnosis. In order to produce relevant data, a sensing device must be well positioned, well calibrated, and

certified for healthcare usage. This is not the case of most of the devices distributed for QS usage [4]. Indeed, for healthcare applications sensing devices should be selected by a clinician based on the patient's particular condition, and the patient should be trained to manipulate these devices.

*Data access:* Users and clinicians are concerned by the possible disclosure of personal sensible data. Most QS applications can only upload data to a pre-defined, non-selectable server. Changing the server's address is therefore not possible, so the users must implicitly trust the owner of this server, assume that their data will be safe there, and that they will not be used without their consent [5].

*Clinical practice:* The doctor-patient relationship is evolving. According to [6], the paradigm of the old thinking 'My health is the responsibility of my physician,' [is] being replaced by the new thinking that 'My health is my responsibility, and I have the tools to manage it.'

*Situational constraints:* Clinicians are skeptical about the necessity to examine data provided directly by patients, considering that this analysis might be a waste of time. They are also worried about a data overload due to the amount of data generated by self-logging tools that may hide critical data.

One of the first project that considered using mobile networking technologies to develop innovative m-health applications has been project MobiHealth [7] (2001-2003). This project only considered using GPRS/UMTS transmissions, though. Since then many other projects have been conducted (a survey can be found in [8]), focusing either on mobile monitoring, on ambient assisted living, or on activity recognition, using either off-the-shelf sensors (e.g. Shimmer<sup>TM</sup>, Zephyr<sup>TM</sup>) or prototype sensors [9].

More recently, research efforts have been focused on the personalization of patients care. The PSYCHE system [10] has been designed to monitor patients affected by bipolar mood disorder. This system first trains itself by identifying the moods of a patient, afterwards it can predict mood state transitions for this patient. Another example of an adaptive system is presented in [11]. With this system a set of rules is defined to monitor each patient, and these rules can be dynamically and remotely modified by the physician in charge of this patient according to the current needs.

## III. OVERVIEW OF THE SHERPAM SYSTEM

### A. Usage scenario

Figure 1 presents a typical usage scenario of the SHERPAM system. This system encompasses all stages of data acquisition, transmission, and processing, and each of these operations shall be performed automatically and continuously, thus allowing the patient to leave a normal life, whether at home or out of home. In the scenario depicted in Fig. 1, a patient is equipped with a wearable kit (WK) that continuously acquires data about his health and general state (acceleration, rotation, heart rate or full ECG, temperature, location, etc.), as well as a smartphone that serves simultaneously as a controller of the Body Area Network (BAN) composed by the sensors, as a front-line data processing unit, and as a gateway between the patient and medical personnel. Transmissions in the BAN

<sup>1</sup>SHERPAM: Sensors for HHealth Recording and Physical Activity Monitoring



Figure 1. Usage scenario of the SHERPAM system.

rely on Bluetooth links. Data received from sensors can be processed locally by the smartphone, using algorithms that do not require too much computation power, so warnings to the patient can be triggered immediately if an anomaly is detected in his current state. The smartphone also serves as a gateway, as the data received from the sensors –or the results of local processing– can be transferred to an aggregation server (AS), which can be deployed anywhere in the cloud (in Fig. 1 we assume this server is located in a hospital’s data center (DC)). Communication between the smartphone and the aggregation server relies on wireless transmission technologies such as Wi-Fi and 2.5G/3G/4G standards. The aggregation server stores the data it receives from all patients, but selected data feeds can additionally be forwarded in real-time to back-line processing units capable of running resource-greedy algorithms that could not fit on a simple smartphone (e.g., algorithms requiring a lot of computation power, or access to large databases for pattern recognition). The whole system is meant to operate automatically, but if a processing unit detects a possible anomaly that requires confirmation by expert eyes, then a solicitation can be sent to medical personnel. In any case, visual and audible warnings can be returned to the patient’s smartphone through the aggregation server.

One of the major objectives of project SHERPAM is that patients with chronic conditions should be allowed to move as freely as possible at home, but also out of their home. A patient’s home is usually equipped with a Wi-Fi access point, and thus constitutes a hotspot (HS1) in which continuous connectivity between the patient’s smartphone and the remote aggregation server. Whenever the patient goes out, however, this connectivity becomes erratic. When out of home the patient may move through other Wi-Fi hotspots (HS2 and HS3), through areas covered by cell-phone networks (C1, C2 and C3), but he may also occasionally move through “white areas” where no wireless connectivity is available whatsoever. The SHERPAM system must keep operating in

such circumstances, and it must notably guarantee that no data is lost during connectivity disruptions. To achieve this goal the application running on the smartphone implements disruption-tolerant networking techniques: data collected from the sensors are assembled in “bundles”, that constitute transmission units between the smartphone and the aggregation server. Each bundle contains meta-information, such as the patient’s id, type of data contained in the bundle’s payload, timestamp, etc. Bundles are first stored on the smartphone, and are transferred to the aggregation server when network connectivity is available. Different strategies can be applied for bundle ordering, depending on the nature of the data they contain.

With this approach the SHERPAM system can easily tolerate network disruptions, and an interesting consequence is that it also helps preserve the smartphone’s battery, as shown in Section IV.

### B. Architecture of the software system

Figure 2 shows the main software components in the SHERPAM system. The left side of the figure shows the wearable kit composed of the sensors worn by the patient and the application deployed on the patient’s smartphone, and the right side shows the aggregation server and processing units deployed in the medical cloud. Arrows in the figure represent how data flow in the system.

Data acquired by sensors enter the smartphone via dedicated connectors. A connector is a software module that is specifically implemented to control a given type of sensor. The smartphone’s application has been designed in such a way that new connectors can be included at any time as software plugins. Extending the system so as to accommodate a new kind of sensor therefore comes down to implementing a new plugin for this sensor, as explained in Section V. A pre-processing unit performs operations on the data output by

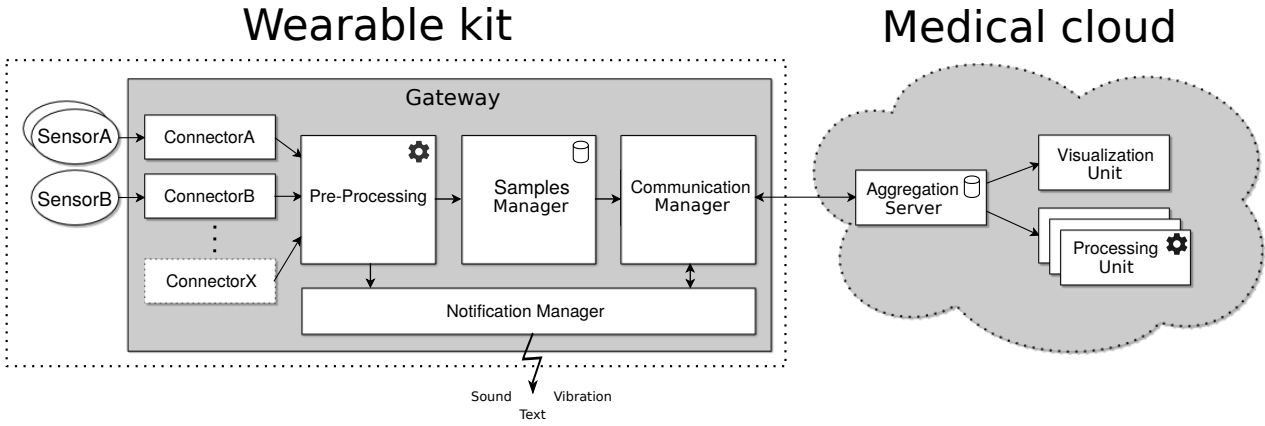


Figure 2. Data flow and logical components in the SHERPAM system

connectors. Pre-processing can include filtering out meaningless data received from a sensor, compressing data, or even running pattern recognition algorithms in order to transmit only important data to the aggregation server, or to generate user notifications locally. After being pre-processed, the data samples are fed to a sample manager so as to be stored in the smartphone's persistent cache before being transmitted to the aggregation server. At this stage, the data samples are aggregated in data bundles, which are then encrypted and signed. The communication manager is responsible for the transmission of bundles over the network. It manages Wi-Fi and cell-phone connections (choosing the most appropriate radio technology at any time, exploiting connection opportunities, switching from one technology to another, etc.), and sends bundles to the medical cloud whenever possible. The main data stream flows from the smartphone to the medical cloud, but the system also allows notifications to flow from the cloud to the smartphone. Incoming notifications are handled by the notification manager, that generates audible or visual signals for the patient.

Bundles received by the aggregation server are deposited in a database, whose content is then accessible for processing units. A basic Web server has been developed as a demonstrator of an interactive processing unit. This server is accessible from a standard Web browser, and it allows authorized users—typically medical staff—to browse the database maintained by the aggregation server, and to download data sets from this server. Specialized processing units can be developed in order to run advanced pattern recognition algorithms on data feeds in real-time, and notify a patient or medical personnel when a particular pattern is detected in a feed.

#### IV. RESULTS

Dependability is a major requirement for a monitoring kit that is meant to be worn by patients all day long. Indeed, wearing such a kit would hardly be accepted by patients if they were required to charge batteries several times in a day.

Wearable sensors can usually operate for at least one day before their battery is depleted, but the autonomy of smartphone must also be taken into consideration. The applications running on smartphones often show low duty cycles, so their host can

be in idle state most of the time. In contrast, a smartphone running the SHERPAM application will never be able to sleep, since it will have to receive data continuously from the sensors, process these data as soon as they are received, scan for available networks, and transmit data to the remote aggregation server whenever possible.

In order to verify that running the SHERPAM application on a standard smartphone is indeed feasible for several hours (and, ideally, for several days), preliminary experiments have been conducted in order to evaluate the energy consumption induced:

- when an application runs continuously on a smartphone, and thus prevents it to ever enter sleep mode;
- when an application uses TCP sessions to transfer data to a remote server, using either Wi-Fi, 2.75G (EDGE), 3.5G (HSPA+), or 4G (LTE) wireless links.

Note that in these tests we deliberately focused on the power consumption induced by medium-range to short-range transmissions (using Wi-Fi or cell-phone networks), and ignored that induced by short-range Bluetooth transmissions.

The tests have all been conducted using Motorola Android smartphones (model Moto G 4G, 2nd gen.), with no mobility and thus no need for the smartphones to switch dynamically from one wireless technology to another. During these tests the smartphones stood at a few meters from a Wi-Fi 802.11n access point, and at about 500 meters from a multi-standard cell-phone tower.

Three transmission modes (“idle”, “continuous” and “bundle”) have been tested for each type of wireless link. In each case a basic application was used to produce data at a constant rate, and transmit these data to a remote server through a TCP session. The data production rate was set at 25 kb/s. This is approximately the bitrate produced by a wearable ECG sensor. In “idle” mode the data were actually discarded instead of being transferred on the wireless link, meaning the wireless interface was up but unused. In “continuous” mode, data bytes produced by the application were sent immediately via the TCP session, without any attempt to bufferize these data at application level. In “bundle” mode the data produced by the application were first assembled into bundles, and a bundle was only sent every minute, each bundle thus containing 187.5

Standard \ Mode	“idle” mode	“continuous” mode	“bundle” mode
Radios disabled	288 h 35’		
Wi-Fi (802.11n)	209 h 53’	57 h 46’	200 h 12’
EDGE (2.75G)	123 h 36’	11 h 42’	46 h 10’
HSPA+ (3.5G)	218 h 21’	12h 54’	86 h 34’
LTE (4G)	103 h 13’	7 h 48’	85 h 30’

Table I

BATTERY LIFE OBSERVED WITH DIFFERENT TRANSMISSION MODES

kB of application-level data. Each test started with a battery level of 100%, and stopped when the battery level dropped below 20%. The screen was off at all times during the tests.

Results are presented in Table I. It can be observed that although the smartphone can run for about 12 days when all radio interfaces are disabled, its battery life drops to at most a week as soon as a radio interface is enabled (even is unused). With continuous transmissions through a radio interface, the battery life drops to half a day or even less for cell-phone technologies, but it is still over two days if using a Wi-Fi link. Finally, the “bundle” mode proves very effective at preserving battery life, as it allows the wireless chipsets to enter power-saving mode whenever the radio channel is not used for a while.

Lessons learned from these preliminary experiments are listed below:

- preventing a smartphone to enter sleep mode does not hinder its capacity to run applications continuously in the background for a long time, provided these applications are not too CPU-greedy and network-greedy;
- transmitting data episodically in “bundles” rather than continuously allows an application to benefit from the power-saving modes defined in wireless technologies, and with these modes the autonomy of a monitoring system based on a smartphone can be increased significantly;
- even when transmitting data in bundles, Wi-Fi transmissions consume a lot less power than cell-phones technologies, so Wi-Fi should be preferred whenever possible.

## V. HETEROGENEOUS SENSOR CONNECTORS

A health monitoring kit should be able to interact with a large variety of sensors and not be limited to a pre-defined set of sensors. Besides the medical staff should be able to prescribe the best suited sensors for each patient according to his/her needs.

Unfortunately, each sensor provider tends to define and implement its own data formats, protocols, and configuration procedures. Bluetooth HDP (Health Device Profile) constitutes an attempt to standardize these elements for Bluetooth health devices (in compliance with ISO/IEEE 11073 family standard [12]), but not all Bluetooth-enabled sensors comply with this standard yet. Besides other sensors rely on alternative transmission technologies, such as ANT+.

In order to deal with the lack of a widely accepted standard for wearable standards, our approach consists in designing in project SHERPAM a generic framework that supports interaction with heterogeneous sensors, this framework being extensible through the use of software adaptors that manage specific kinds of sensors.

### A. Plugin approach

The architecture of the SHERPAM app has been defined in such a way that new kinds of sensors can be included at any time, without changing the source code of the app itself, using a plugin approach. Software connectors, each designed to handle a specific kind of sensor, can be *plugged in* the SHERPAM app as when needed, thus enabling it to use a set of sensors those connectors are able to manage. Extending the SHERPAM app for a new kind of sensor therefore comes down to implementing the appropriate connector plugin. The core application is totally independent from the number and the kind of available connectors. Thus, when the SHERPAM app is installed on a smartphone for a target patient, only the connectors required for the sensors this patient will wear must be deployed on the smartphone.

Figure 3 presents the class diagram of the plugin mechanism developed for the SHERPAM app. It is composed of three major elements, and an API.

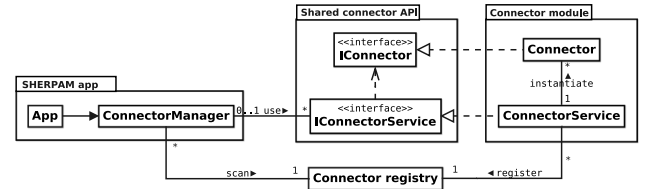


Figure 3. Class diagram of the plugin mechanism

A *connector module* must be defined for each kind of sensor. It acts as a software adaptor between the SHERPAM app and the sensor’s specific data formats and transmission protocols, by implementing code that conforms to the *shared connector API*. This API is itself composed of the *IConnector* and *IConnectorService* interfaces. *IConnector* describes how to connect to a sensor, and get data samples from this sensor. *IConnectorService* creates a connector instance for each sensor, and manages its lifecycle.

The *connector registry* allows the SHERPAM app to discover and access available connector modules. When a new module is installed on the smartphone, it spontaneously registers with the connector registry. By scanning the registry an application can thus discover the available connector modules.

At startup the SHERPAM app uses its *ConnectorManager* to scan the connector registry and binds to the *ConnectorService* of each connector installed. When the application has to interact with a sensor, a connector is instantiated by using the appropriate module.

### B. Configuration and payload descriptors

Connector modules allow the SHERPAM app to manage heterogeneous sensors. However, each sensor defines its own data format and configuration parameters because of the lack of a widely accepted standard for wearable sensors. In order to deal with this problem generic mechanisms have been defined for the SHERPAM app.

Each connector module defines two descriptors. A *configuration descriptor* is used by the plugin loader to identify

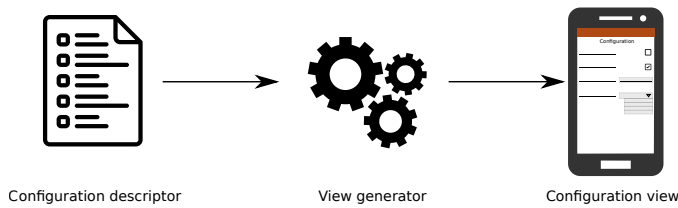


Figure 4. Generation of a specific configuration view from a configuration descriptor provided by the connector

the parameters that must be set to achieve the configuration of a sensor. This descriptor is used by the plugin loader to generate a configuration view at runtime, so that a user – typically medical staff – can specify what kind of data must be produced by the sensor (e.g., heart rate, 3-axis acceleration, etc.), and in what conditions (e.g., acquisition frequency). This configuration can be adjusted according to the target patient's needs, since all patients do not necessarily have to be monitored the same way.

Similarly, a *sample descriptor* describes the format of the data samples that will be generated by the sensors (e.g., data types, data units, etc.). This descriptor is used by the SHERPAM app to determine how to decode the stream of data samples it will receive from the sensor once data acquisition has started.

A patient can thus be equipped with different kinds of sensors, and the SHERPAM app can configure each sensor at startup according to the patient's profile, and then process the data streams it will receive accordingly.

### C. Current state of the system's implementation

To date an Android implementation of the SHERPAM app has been developed, and connector modules have been implemented that already allow this app to drive SHIMMER™ sensors (for ECG, EMG, GSR, acceleration, etc.), the ZEPHYR BIOHARNESS™ sensor (for heart rate and breathing rate, GPS, etc.), heart rate monitors that comply with the Bluetooth HRM health device profile, as well as sensors that may be embedded directly in an Android smartphone or smartwatch (e.g., acceleration, compass, GPS). Thanks to SHERPAM's plugin mechanism, other connector modules should be developed in the future in order to drive other types of Bluetooth HDP-enabled sensors (e.g., pulse oximeters, glucose meters, blood pressure monitors), as well as sensors that use ANT+ for transmissions.

## VI. CONCLUSION AND FUTURE WORK

In this paper we have described a flexible software system that has been designed to monitor continuously the health state of patients during their daily activities. This system covers all steps of data acquisition by wearable sensors, data gathering and pre-processing by an Android application, and data transfer to a remote aggregation server for storage and advanced analysis. Unlike other systems with similar objectives, the SHERPAM system is meant to be highly flexible and customizable. Data can be combined from different kinds of sensors simultaneously, and dedicated processing modules can be included in the system as needed, depending on the

patient's particular condition. Additionally, the system uses disruption-tolerant networking mechanisms in order to prevent data loss, and guarantee data integrity.

A one-year medical trial is planned in project SHERPAM, starting in 2017. During this trial the whole system will be used to monitor the health of patients with different kinds of chronic diseases. Until then the system still needs to be improved in several ways. Specialized pattern recognition modules will be included in the system, using the plugin mechanism described in this paper. Moreover context-aware algorithms shall be implemented in the Android app in order to drive data transmissions between the smartphone and the aggregation server, balancing automatically between the nature of the data to be transferred and the cost of using one or another wireless technology for this transfer.

At the end of project SHERPAM, the system will be distributed under the terms of an open-source license.

## ACKNOWLEDGMENT

This work is part of the SHERPAM project (2014-2018). As such it has received a French government support granted to the COMIN Labs excellence laboratory, which is managed by the National Research Agency in the "*Investing for the Future*" programme under reference ANR-10-LABX-07-01. Further information about this project can be found at <http://www.sherpam.cominlabs.ueb.eu/>.

## REFERENCES

- [1] European commission: The 2015 Ageing Report: Economic and budgetary projections for the 28 EU Member States (2013-2060). (March 2015)
- [2] Adam J., S., Neil Jay, S., Andrew, A.: The Challenges of Mobile Health Regulation. *Journal of the American Medical Association* (May 2016)
- [3] West, P., Giordano, R., Van Kleek, M., Shadbolt, N.: The Quantified Patient in the Doctor's Office: Challenges & Opportunities. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems 2016*, ACM (May 2016) 1–13
- [4] Haruka, M., Ryoko, K., Satoshi, N., Yoshio, N., Kazuko, I.T., Shigeho, T., Motohiko, M.: Accuracy of Wearable Devices for Estimating Total Energy Expenditure: Comparison with Metabolic Chamber and Doubly Labeled Water Method. *Journal of the American Medical Association* (May 2016)
- [5] Till, C.: Exercise as labour: Quantified self and the transformation of exercise into labour. *Societies* (August 2014) 446–462
- [6] Swan, M.: Health 2050: The Realization of Personalized Medicine through Crowdsourcing, the Quantified Self, and the Participatory Biocitizen. *Journal of Personalized Medicine* 2 (September 2012) 93–118
- [7] Van Halteren, A., Bults, R., Wac, K., Konstantas, D., Widya, I., Dokovski, N., Koprnikov, G., Jones, V., Herzog, R.: Mobile Patient Monitoring: The Mobihealth System. *Information Technology in Healthcare* (February 2004) 365–373
- [8] Alemdar, H., Ersoy, C.: Wireless Sensor Networks for Healthcare: A survey. *Computer Networks* 54(15) (2010) 2688–2710
- [9] Chuo, Y., Marzencki, M., Hung, B., Jaggernauth, C., Tavakolian, K., Lin, P., Kaminska, B.: Mechanically Flexible Wireless Multisensor Platform for Human Physical Activity and Vitals Monitoring. *IEEE Transactions on Biomedical Circuits and Systems* 4 (October 2010) 281–294
- [10] Lanata, A., Valenza, G., Nardelli, M., Gentili, C., Scilingo, E.P.: Complexity Index from a Personalized Flexible Monitoring System for Assessing Remission in Mental Health. *IEEE Journal of Biomedical and Health Informatics* (January 2015) 132–139
- [11] Siddiqui, A., Koch, O., Rabie, A., Handmann, U.: Personalized and Adaptable M-Health Architecture. In: *Wireless Mobile Communication and Healthcare (Mobihealth)*, 2014 EAI 4th International Conference on, IEEE (November 2014) 381–384
- [12] ISO/IEEE 11073-00103-2012: Health informatics-Personal health device communication Part 00103: Overview (November 2012)