

# OPAIEH: An Ontology-based Platform for Activity Identification of the Elderly at Home

Jorge Gómez Montalvo, Cristhian Carrillo,  
Luis Basto Díaz, Francisco Moo Mena, Víctor Menéndez Domínguez

Universidad Autónoma de Yucatán,  
Facultad de Matemáticas,  
Mexico

{jgomez, luis.basto, mmena, mdoming}@correo.uady.mx,  
cristhiancarrillo07@gmail.com

**Abstract.** Recently, the sector of the population older than 60 years of age is growing. The assistance for the elderly at their home allows to increase their autonomy and independence while they are living alone. Several fields have emerged to improve the quality of life of the elderly, and to develop environments that offer help, support and assistance during the realization of their daily activities. The identification of activities is a key piece for the provision of assistance to seniors who live alone. This work is focused on the activity recognition of the elderly who live independently at home. Here is presented OPAIEH, an ontology-based platform for activity identification of the elderly at home. The platform includes the ontological model, which allows a new activity characterization. Also, the platform includes a sensors network, a client device, and web server to perform the recognition of different activities that the elderly do inside their home. Furthermore, the platform generates a set of graphs that shows different statistics and behaviours of the users. In order to perform an experimental test on OPAIEH, a case study was developed as a proof of concept about the use of ontologies for the activity recognition task. The results encourages us to continue our work.

**Keywords.** Indoor activity identification, OWL ontologies, ambient assisted living.

## 1 Introduction

Several publications made by the “National Institute of Statistics, Geography, and Informatics” of Mexico [12, 13], as well as some reports made

by the World Health Organization (WHO) [20], show that the sector of the population that is older than 60 years of age is growing. Certainly, the life expectancy of people has increased in recent years, which has allowed the percentage of adults over 60 to grow. The WHO mentions that between 2015 and 2050, the sector of the population with more than 60 years will increase from 900 to 2000 millions.

From a social perspective, aging of the population implies new problems and challenges for any society, for example: offer health services focused on the elderly, provide transportation systems adapted to the mobility of the elderly, have information technology services aimed at the elderly [7]. Furthermore, the WHO points out that physical and social environments play a key role for the elderly while trying to maintain themselves healthy and independent [21].

The fields of Gerontechnology and Ambient Assisted Living (AAL) have emerged to solve some issues of the ageing population, improve the quality of life, and develop environments that offer help, support and assistance to elderly during the realization of their daily activities. The AAL is a multidisciplinary field that uses information and communication technologies (ICT) in personal health and telehealth system to counteract the effects of the growing population of older adults [15].

The AAL systems provide an ecosystem of medical sensors, computers, wireless networks and applications for health monitoring. The main goal of AAL solutions is extend the time during which the elderly can live independently in their preferred environment using ICT to preserve their personal health.

The assistance for the elderly in their home through smart sensors is a service that is currently becoming more import, since it allows to increase the autonomy and independence of elderly while minimizes the risks of living alone [4]. The identification or recognition of activities is a key piece in providing health care services and assistance to seniors who live alone [3], inasmuch as a precise activity recognition can provide the information required to perform an adequate assistance.

Regarding activity recognition at home, one important characteristic of AAL solutions is the intrusion factor. It means, AAL systems and platforms discourage their own utilization when they invade private aspects of elderly daily living. For example, the use of video cameras or microphones, in a bedroom or bathroom, to identify daily activities of an independent elderly person.

This paper presents OPAIEH, an Ontology based Platform for Activity Identification of the Elderly at Home. We present the ontological model named IAR-ONTO, which represents the activity definition proposed in this research work. OPAIEH platform includes a bluetooth sensors network, a client device (OPAIEHClient), and web server (OPAIEHServer) to perform the recognition of different activities that the users do inside their home. Also, it was developed a web client (OPAIEHClient-Web), whose main function is to provide a set of graphs that shows different statistics and behaviors of the users inside their home.

It is important to mention that this work presents a case study as a proof of concept about the use of ontologies in the activities recognition task, mainly for two reasons:

1. The experimental test developed in order to evaluate the performance of OPAIEH platform was not performed in seniors, inasmuch as it

is necessary that users use a smartphone and most older adults find it difficult to use such devices [1].

2. Carry a smartphone with yourself while displace around all the rooms of the home could be uncomfortable, so the use of a smartphone for OPAIEH was not thought as a final device.

The rest of the paper is organized as follows. Section 2 reviews related works about systems for the activity recognition and how they implement such recognition, both at hardware and software level; also we compare our platform with each of that systems. In section 3, an activity definition is proposed. Section 4 describes the design of the ontological model based on our activity definition. Section 5 explains the function of each component of our proposed platform. Section 7 presents the algorithm developed for activity recognition based on ontologies. Section 8 describes the experimental test performed as a proof of concept, as well as the results. Finally, Section 9 presents the conclusion of this work.

## 2 Related Work

In order to achieve indoor activity recognition, a vast majority of AAL systems use devices like camcorders, sensors, as well as Wi-Fi and radio signals. For example, in [2] a video-based system is proposed that allows recognizing activities by extracting the temporal and spatial characteristics of the data collected by a camcorder. The video-based systems are very useful for monitoring in real time, and provide very complete and accurate information, but they have a drawback, they expose to a great extent the privacy of the user. Our platform allows the caregiver to be informed about the elderly about the activities carried out by him trying to minimize the intrusion in his daily life.

A type of sensors used in the recognition of activities within the home are magnetic sensors. [17] describes a platform based on the use of a magnet by the user and the placement of magnetometers in the home, which sense the

changes in the magnetic field that occur when the user approaches the sensors. One advantage of this platform is the use of a magnet in the form of a bracelet, which is very practical. The disadvantage of this platform is based on the fact that there is a trade-off between the strength of the magnet to be used, if it is very strong it can attract metal objects, and if it is very weak it can not be detected adequately by magnetometers.

In [14], another platform is described that uses magnetic sensors but with a different approach, it is based on the fact that in a home there are various electrical devices, therefore attaching to the user a glove that has embedded Hall sensors and using decision trees as the engine to perform the analysis of the information, it is possible to recognize the activities performed by the user by sensing the magnetic fields that are generated when the user come into contact with the different electrical and electronic devices found in the home.

The work done in [16] describes a data acquisition system that uses a reader and tags of radio frequency identification (RFID), as well as a probabilistic engine and Bayesian networks, which from the data collected by the RFID reader (when interacting with "tagged" objects) allow to infer the possible activities that are carried out. The works done in [14] and [16] have the disadvantage of having to attach to the user large electronic devices. Indeed, the platform presented in this paper proposes a mobile application that must be installed on the user's smart phone, and certainly, the user must carry that device as he moves in his home, the justification is that the smart phones are devices that are already present in people's daily lives.

In addition to the use of sensors, different radio signals have also been used in internal environments for the recognition of activities, like the Wi-Fi [8,9] and radio frequency [11] signals. In [8] and [9], it is described a platform that uses Wi-Fi signal, in specific the Received Signal Strength Indicator (RSSI), and an algorithm that combines k-NN classifiers and decision trees in order to recognize the activities performed by a user in a room. This platform is based on the principle that the Wi-Fi signal is blocked or reflected in different ways according to activity carried out

by the user, therefore, for each of the activities, a unique RSSI mark is returned to the receiver, which is analyzed by the above mentioned classification techniques. This procedure allows to recognize which activity is carried out in a specific moment. The disadvantage of Wi-Fi-based platforms for activity recognition is that until now they can only recognize a few activities (like walk, stand up and sit down) inside the room where the transmitters and receivers of these signals are placed. On the other hand, the platform presented here is able to recognize all the activities that the user registers.

### 3 Activity Characterization

OPAIEH platform was developed taking as reference our characterization of activity which involves the following four parameters:

- *Room*. It is the place of the home where the activity is carried out (e.g. bathroom, kitchen, dinning room).
- *Day*. It is the day (or days) of the week in which the activity is performed.
- *StartTime*. It is the time at which the user begins to perform the activity.
- *Duration*. It is the time it takes the user to perform the activity. An activity starts when the user enters a room, and ends when the user leaves the room.

Therefore, by using these 4 parameters, an indoor activity may be defined as the action that is performed in a *room*, on one or several *days* of the week, with a *start time*, and with a *duration*. For example, an older adult performs the activity "have breakfast" in the dining room, from Monday to Saturday, at 7 o'clock, and she/he performs this activity for 20 minutes.

In this project, it is proposed that the elderly person must characterize, only one time, their activities through OPAIEH. Such activity characterization is necessary in order to avoid the use of devices with a high level of intrusion like video cameras or microphones for activity recognition.

By using OPAIEH, once the user has characterized his main activities, OPAIEH will be able to identify them.

Semantically speaking, it is important to say that the user is not giving us private information nor the specific name of his activities, but only the values of the 4 parameters of a “certain” activity. In other words, what we are proposing is that OPAIEH will be able to identify activities by measuring, in near real-time, the parameters proposed for the activity characterization.

## 4 Ontological Model

Accordingly to the proposed activity characterization, it was designed and built IAR-ONTO (Indoor Activity Recognition Ontology), that is the knowledge base for activity recognition in OPAIEH. For the design of IAR-ONTO, Protégé was used, a free open source platform that allows the creation, visualization, and manipulation of ontologies in various representation formats, including OWL [10].

An OWL ontology is composed by classes, properties (relations between individuals) and instances (individuals of those classes). There are 3 classes in IAR-ONTO, the central class *Activity*, and two auxiliary classes *DaysofTheWeek*, *Room*. The class *Activity* has 5 properties, *hasDay*, *hasRoom*, *hasStartTime*, *hasDuration* and *hasActivityName*, the first 4 properties correspond to the parameters of our activity definition, the last one add the identification (or name) of the activity. Also, two types of properties were created in IAR-ONTO, *Object properties* and *Datatype properties*.

The first type of properties link an individual to an individual, like the property *hasDay*, which links an individual of the class *Activity* to up to 7 individuals of the class *DaysofTheWeek*. The second type of properties links an individual to an XML Schema Datatype value, like the property *hasStartime*, which links an individual of the class *Activity* to an integer datatype, that represents the *StartTime* of the activity in minutes (a day has 1440 minutes, so the 7 A.M. o'clock is considered by IAR-ONTO as 420). Finally, two type of individuals were considered, *created in design time* and *created at runtime*.

The individuals created in design time are those that were instantiated just after the creation of IAR-ONTO, their instantiation is possible because any activity that is created by any user will require these. IAR-ONTO only has 7 individuals created in design time, that belong to the class *DaysofTheWeek*, which correspond to the 7 days of the week. The individuals created at runtime (corresponding to the *Room* and *Activity* classes) are created by each user and the values of their attributes are assigned explicitly by themselves through a client device. The IAR-ONTO class diagram is shown in Figure 1.

## 5 OPAIEH's Architecture

OPAIEH is composed by four entities: sensor's network, client (OPAIEH Client), server (OPAIEH Server), and web client (OPAIEH Client-Web). Figure 2 shows the architecture of OPAIEH. The interaction of all four components allows activity recognition at home, as well as the visualization of the statistics of the activities. Next, each of the components is described.

### 5.1 Sensors Network

Unlike external environments where GPS signal is broadly used in order to know the location of a person or device, when it comes to internal environments, it is not recommended to use the GPS signal, since it does not have enough precision to determine the location of a person inside a home [22]. For this reason, the sensors used in OPAIEH are *beacons*.

A beacon sensor sends a Bluetooth broadcast signal containing several parameters, but we are interested only in the ID and the Radio Signal Strength Indicator (RSSI), because they allow us to compute the indoor location of the user.

Indeed, depending on the size of a room, at least one sensor is installed in each room of the home, establishing a sensors network interacting with OPAIEHClient, which measures the strength of the broadcast signal, allowing us to compute how close is the receiver from each sensor. In this manner, OPAIEHClient is able to determine the location of the user inside his home, as the sensor

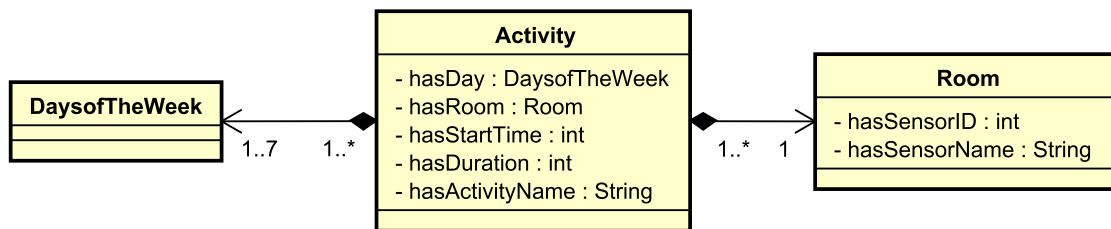


Fig. 1. Class diagram of IAR-ONTO

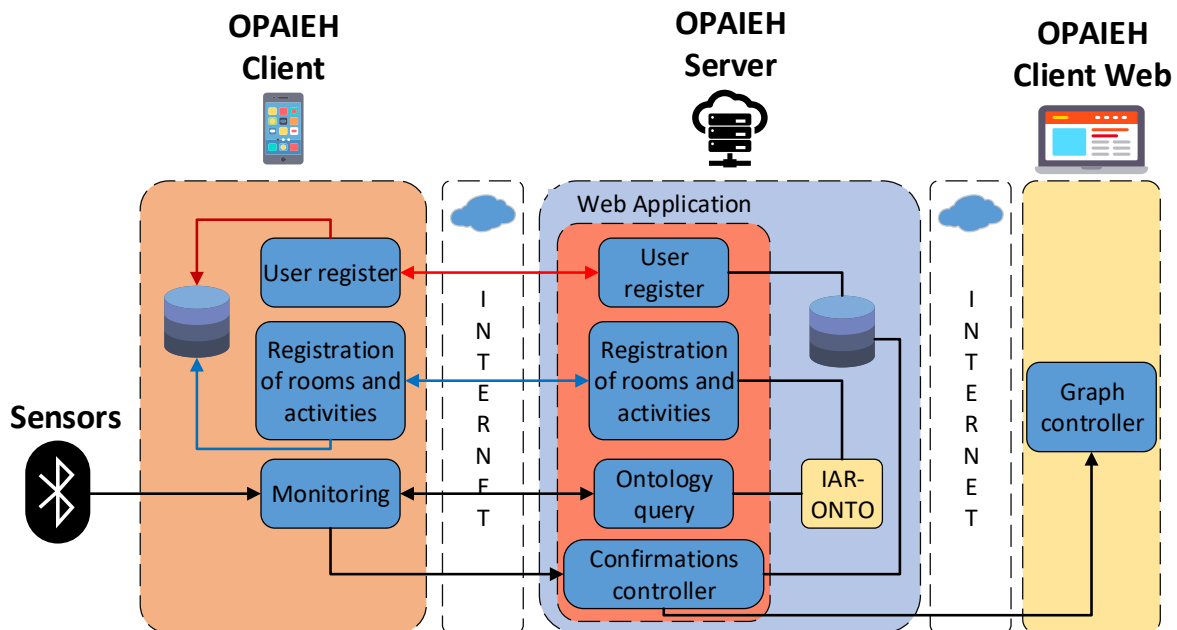


Fig. 2. OPAIEH architecture diagram

who sends the strongest signal is the one that is closest to the user.

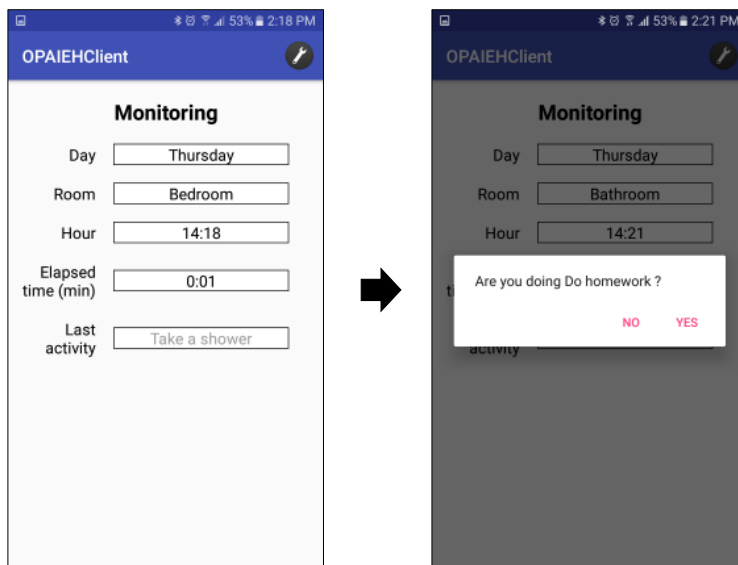
## 5.2 OPAIEHClient

OPAIEHClient is an Android application composed by three main modules:

1. *User's register.* This module allows the users to enter their personal data, which are: name, password and age. This information is sent to the server (OPAIEHServer), which stores it in a database.

2. *Registration of rooms and activities.* Through this module, the users can register their rooms by linking each room with its corresponding sensor. Also, the users can characterize the main activities that they will perform at home.

3. *Monitoring.* This module has three functions:
  - 1) It receives the Bluetooth signal emitted by the sensors and it determines in which room the user is inside;
  - 2) it sends information to the server about the activity that the user was doing (day, room, hour in minutes, and duration in minutes), the server analyzes that information (by means of our proposed inference algorithm) and returns to



**Fig. 3.** Screenshot of OPAIEH Client monitoring interface

OPAIEHClient the name or ID of the *possible* activity (inferred activity) that the user was doing; 3) it asks the user if the inferred activity is the same that the one he was actually doing (feedback), the answer of the user is sent to the server and saved in the database, we call this feedback process *activity confirmation*. Figure 3 shows a screenshot of the monitoring interface of OPAIEH Client, before it sends the data to the server (left side) and after sending the data (right side).

### 5.3 OPAIEHServer

OPAIEH Server is composed of three main modules: Web application, Database, and IAR-ONTO (see Section 4).

#### 5.3.1 Web Application

This module contains components that allow: 1) the registration of users, rooms, and (one-time definition of) activities; 2) execution of queries on IAR-ONTO; 3) The management of activities' information.

1. *User's register*. This component register the personal data of the user that receive from OPAIEHClient into the database of OPAIEHServer.
2. *Registration of rooms and activities*. This component manage the storage of the rooms and activities into the IAR-ONTO instance of each user. The most important feature of this module is the storage of the *predefined activities*, which in this project are defined as those that the people perform multiple times per day, in other words, they are activities in which the StartTime isn't a parameter that allow to identify them. In OPAIEH, they were considered two predefined activities: *washing dishes* and *toilet activities* (physiological needs and brushing teeth), both activities are storage into the replica of IAR-ONTO that corresponds to the user, when this one registers into OPAIEHClient the rooms *Kitchen* and *Bathroom*, respectively.
3. *Ontology query*. This component is responsible of receive the activity data which is sent by the *Monitoring* module of OPAIEHClient, with that information the Ontology Query component performs queries into the IAR-ONTO

**Table 1.** Table of confirmed activity

No.	Date	Room	Duration	Inferred activity	List activity	User activity
1	2018-01-23 09:28:42	Dinning room	7	1	0	Have breakfast
...	...	...	...	...	...	...
12	2018-01-30 15:49:35	Bathroom	9	0	1	Take a shower
...	...	...	...	...	...	...
21	2018-02-05 15:00:26	Bathroom	3	0	0	Tooth brush

replica of the user, the purpose of these queries is known what was the possible activity that the user was doing. Once this component obtains the activity, it will share it with the Monitoring module of OPAIEHClient. The process for making queries is explained in more detail below (see Section 7)

4. *Confirmations controller.* This module is responsible for saving the activity confirmations in the database, specifically into the table of confirmed activity (each user has a table). Also this module makes queries on the tables of confirmed activity to extract information what is requested by OPAIEHClientWeb, which uses that information to build graphs that shows different statistics of the activities performed by the users inside their home.

### 5.3.2 Database

OPAIEHServer database is composed by two kind of tables: *users table* and *table of confirmed activity*. The users table is responsible to storage the register data of the users (username, password), it helps in the process of login into OPAIEHClient and OPAIEHClient-Web.

The tables of confirmed activity (one table per user) are responsible to storage the confirmed activities. This table is composed by seven columns: 1) *No.*, it represents the id of the activity; 2) *Date*, it exposes the day and the hour when the activity starts; 3) *Room*, this column stores the room where the activity was performed; 4) *Duration*, this column stores the total time that the activity lasted; 5) *Inferred activity*, this column equals to 1 if the activity was inferred correctly, otherwise equals 0; 6) *List activity*, this column equals to 1 if the activity wasn't inferred correctly

but it was stored in the OPAIEHClient database, otherwise equals 0; 7) *User activity*, it represents the activity that the user confirm. Table 1 shows an example of a table of confirmed activity.

### 5.4 OPAIEHClient-Web

OPAIEHClientWeb is a web process that allows the caregiver of the older adult to visualize in a graphic way some statistics of the activities carried out by the elder person. OPAIEHClientWeb takes care of requests for information about the activities (table of confirmed activity) from the *Confirmations controller*, with that information, the *Graph controller* generates 5 different graphs, which are created using the libraries ChartJS [5], D3 [6], and SIMILE Timeline [19].

1. *Entries per room.* This vertical chart bar created with the ChartJS library, represents the number of times (frequency) that the user enters to each room. The horizontal axis is for the rooms and the vertical axis is for the frequency. Figure 4 shows an example of this kind of chart.
2. *Average duration of activities.* This horizontal chart created with the ChartJS library, represents the duration average of the activities (user's confirmations) in minutes. The horizontal axis is for the duration average and the vertical axis is for the activities. Figure 5 shows an example of this kind of chart.
3. *User's room path per day.* This dot chart created with the ChartJS library, represents the room path of the user per day. The horizontal axis is for the time and the vertical axis is for the rooms. Figure 6 shows an example of this kind of chart.

4. *Directed graph of the relationship between activities and rooms per day.* This directed graph created with the D3 library, represent the relationship between the activities sequence per day (Next) and the rooms where that activities was performed (Done in). Figure 7 shows an example of this graph.
5. *Timeline of the activities.* This timeline created with the SIMILE Timeline library, represents all the activities since the users started their monitoring until the access to OPAIEHClientWeb. Figure 8 shows an example of this timeline.

## 6 Operation of OPAIEH inside the home

Figure 9 shows the diagram of the operation of OPAIEH and the actors involved in that process. There are 4 operations on the platform: *putting the sensors, registering, monitoring, consulting.*

1. *Putting the sensors.* Either the older adult or his caregiver must puts a sensor in each room involved in the monitoring.
2. *Registering.* After putting the sensors through the home, the older adult (or his caregiver) can already registers his personal information, the rooms (link each room with the sensor inside of it), and rooms through the monitoring device (or device client).
3. *Monitoring.* Once made the register, the elderly can starts the monitoring. In this stage the monitoring device sends information to the server, that has embedded the inference algorithm.
4. *Consulting.* The elderly's caregiver can consult the information about the activities through the Web Client. This information could helps in order to look for patterns that indicate unusual behaviors of the older adult.

## 7 Inference Algorithm

In this section we explain how the queries are performed on the IAR-ONTO's replicas, the execution of the queries is based on the algorithm inference. The queries are created in SPARQL language [18].

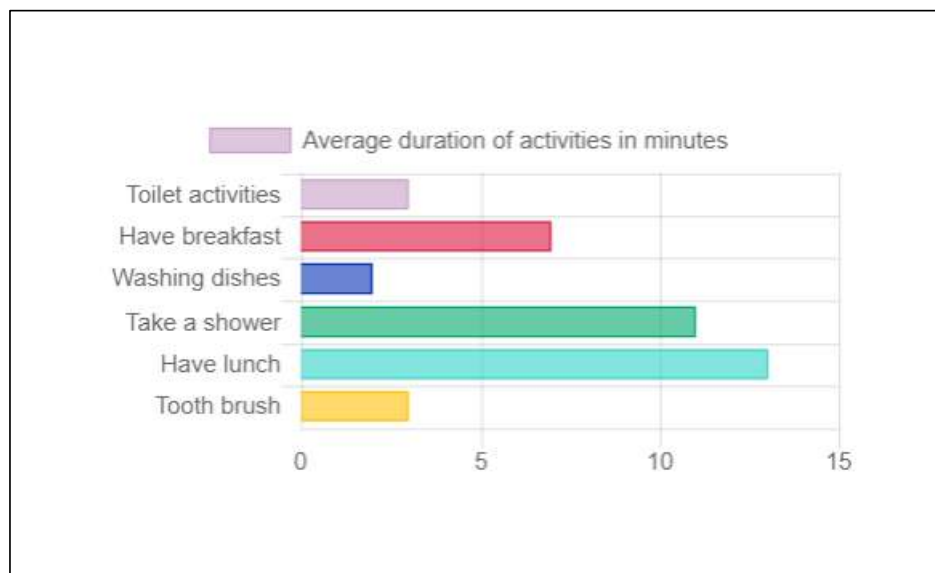
The inference algorithm is invoked by the *Ontology query* component each time that the older adult moves from one room to another. In every change of room, OPAIEHClient sends the data of the activity, that the user was performing, to the Ontology query component. The data consists of *Day, Room, StartTime* and *Elapsed time*. When the Ontology query component receives the activity's data, the inference algorithm starts, the first two queries created are Query StartTime and Query Duration:

- *Query StartTime (QST).* This query is based on the start time of the activity performed by the user. In order to do this query flexible, it's necessary to apply it an interval, which was fixed to  $\pm 10$  (minutes). Figure 10 shows QST in SPARQL format. In natural language, this query can be read as "*Select the activity performed on this day, at this room and at this interval of start time.*" The query will return the activity stored in the IAR-ONTO's replica of an specific user that match with that parameters.
- *Query Duration (QD).* This query is based on the duration of the activity performed by the user. In order to do this query flexible, it's necessary to apply it an interval, this interval depends on the room where the activity was performed, they were fixed three intervals,  $\pm 4$ ,  $\pm 5$  and  $\pm 6$ , for activities performed at the bathroom, performed at the kitchen, performed at the rest of rooms, respectively. Figure 11 shows QD in SPARQL format. In natural language, this query can be read as "*Select the activity performed on this day, at this room and at this interval of duration.*" The query will return the activity stored in the IAR-ONTO's replica of an specific user that match with that parameters.





**Fig. 4.** Entries per room chart



**Fig. 5.** Average duration of activities chart

Based on the results of QST and QD, four cases can be presented.

1. *Case 1.* If either the result of QST ( $R_{QST} \neq null$ ) and the result of QD ( $R_{QD} \neq null$ ) are

not null, the inferred activity will be the result given by QST ( $R_{QST}$ ).

2. *Case 2.* When QST has a result not null ( $R_{QST} \neq null$ ) and QD has a null result

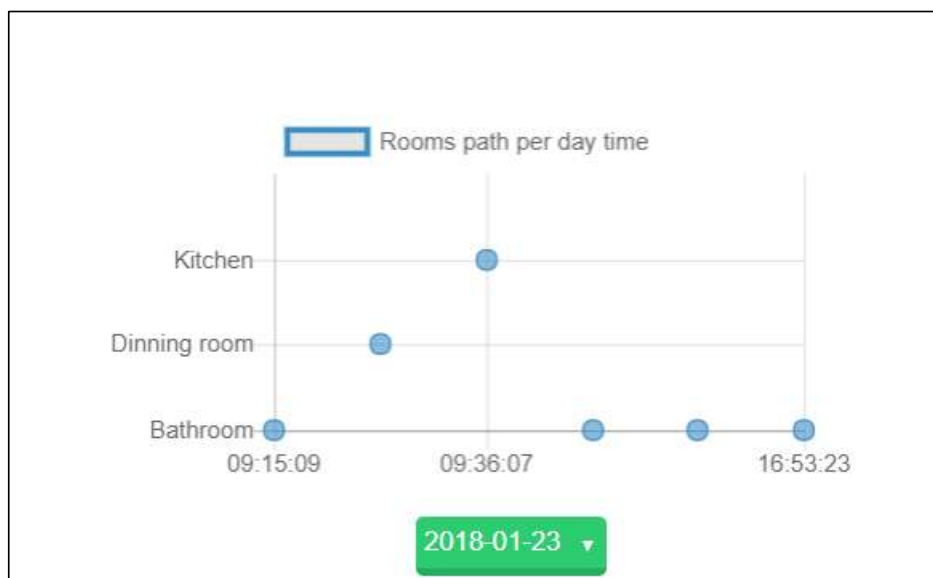


Fig. 6. User's room path chart

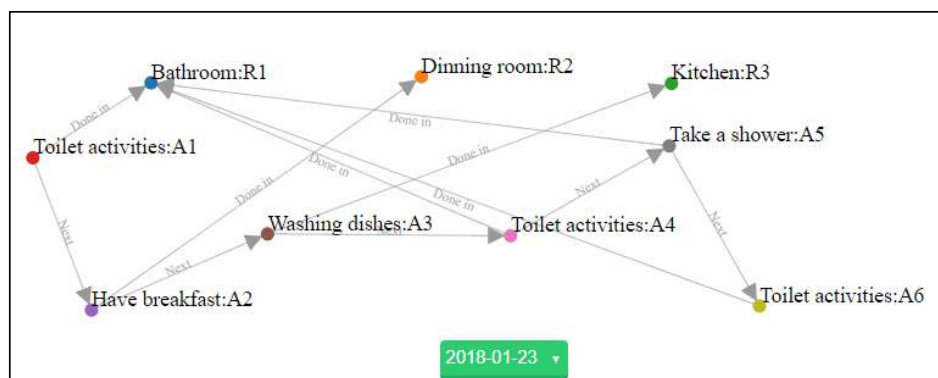


Fig. 7. Directed graph of the relationship between activities and rooms per day

( $R_{QD} = null$ ), the inferred activity will be the result given by QST ( $R_{QST}$ ).

3. *Case 3.* When QST has a null result ( $R_{QST} = null$ ) and QD has a not null result ( $R_{QD} \neq null$ ), it is necessary to perform another query, named Query Find StartTime (QFST), with the purpose of checking if the resulting activity of QD ( $R_{QD}$ ) is consistent with the activity performed by the user. For

example, in a specific IAR-ONTO's user replica are established the follow activities, *Have breakfast* that is performed at Dinning Room from Monday to Saturday at 7:00 (420 in minutes) with duration of 10 minutes, and *Have lunch* that is performed at Dinning Room from Monday to Saturday at 13:00 (780 in minutes) with duration of 20 minutes. One day the user performs an activity and OPAIEHClient sends the next parameters to

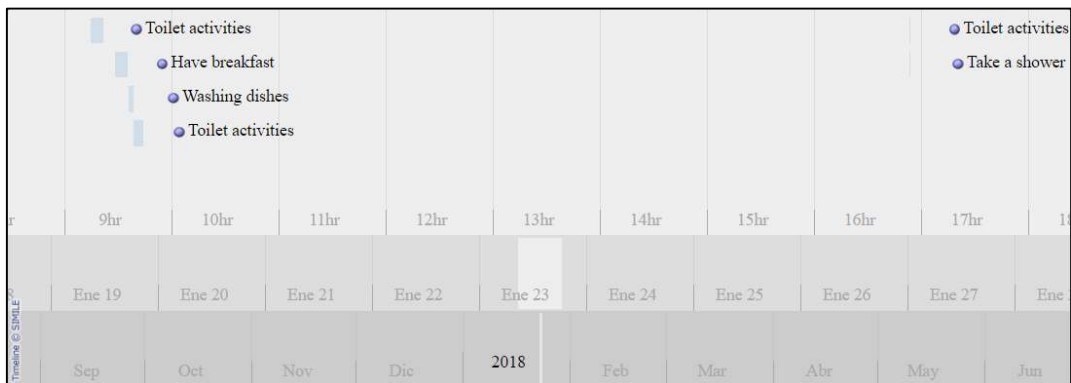


Fig. 8. Timeline of the activities

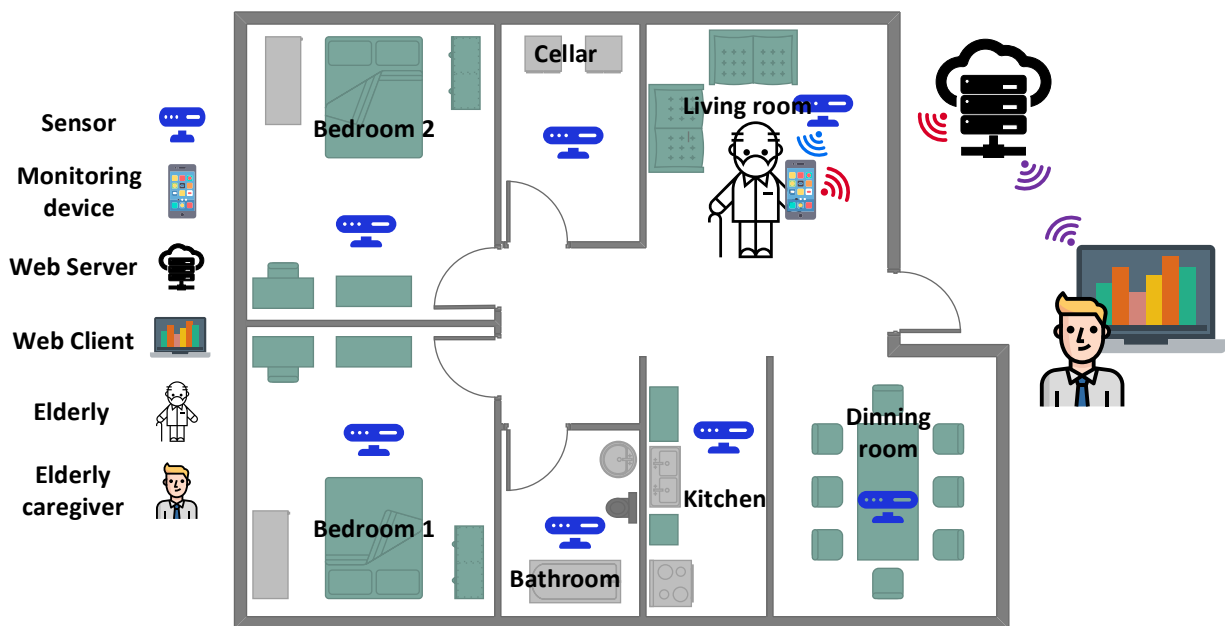


Fig. 9. Diagram of the operation of the platform

OPAIEHServer, Room = Dinning Room, Day = Wednesday, StartTime=440, Duration=16. With that information the result of QST will be null, because the StartTime of the activity Have breakfast isn't include in the StartTime interval of the activity performed by the user

(440±10 or 430-450), we need that all the parameters of both activities are equal (the one in IAR-ONTO and the performed by the user). The result of QD will be Have lunch, because the duration of that activity fits in the duration interval of the activity performed by

```
“SELECT ?X WHERE { ?X a:hasDay a:day . ?X
a:hasRoom a:room . ?X a:hasStartTime ?Y .
FILTER (?Y >= minRange && ?Y <= maxRange) }”
```

**Fig. 10.** Query StartTime (QST) in SPARQL format

```
“SELECT ?X WHERE { ?X a:hasDay a:day . ?X
a:hasRoom a:room . ?X a:hasDuration ?Y .
FILTER (?Y >= minRange && ?Y <= maxRange) }”
```

**Fig. 11.** Query Duration (QD) in SPARQL format

the user ( $16 \pm 6$ , or 10-22). If we select the result of QD as the inferred activity we have an inconsistent problem, Who has lunch at 7:20?. For that reason we perform QFST, which consists of consulting the StartTime of the activity resulting from QD ( $QFST(R_{QD})$ ), if the result of QFST ( $R_{QFST} = 0$ ) is equal to 0 (the activity is a predefined activity) or if the absolute value of the difference between the StartTime of the activity resulting of QD and the StartTime of the activity performed by the user is less than 120 (2 hours), the activity inferred will be given by the result of QD ( $R_{QD}$ ), otherwise it will be performed another query, that we will explain in Case 4. The Query Duration is useful in the search of predefined activities, due to its nature that they can be performed at any time of the day. Figure 12 shows QFST in SPARQL, in natural language this query can be read as “Select the StartTime of the activity with this name”.

4. *Case 4.* If either QST and QD have a null result or if the result of QD is inconsistent, we perform the last two queries named Query Nearest StartTime (QNST), one query is for forward the StartTime and the other is for Backward the StartTime. That queries have a similar structure of QST, with the difference that the parameter day is eliminated. We run both queries in a loop until one of them returns a result, that result will be the inferred activity. Figure 13 shows both QNST queries in SPARQL format.

Figure 14 illustrates the flowchart of the inference algorithm.

```
“SELECT ?StartTime WHERE { ?activity
a:hasStartTime ?StartTime . ?activity
a:hasActivityName “ResultQueryDuration” }”
```

**Fig. 12.** Query Find StartTime (QFST) in SPARQL format

```
“SELECT ?X WHERE { ?X a:hasRoom a:room . ?X
a:hasStartTime StartTimeBackward }”
```

(a)

```
“SELECT ?X WHERE { ?X a:hasRoom a:room . ?X
a:hasStartTime StartTimeForward }”
```

(b)

**Fig. 13.** Query Nearest StartTime (QNST) in SPARQL format, (a) Backward (b) Forward

## 8 Experimental Test and Results

In this section, we explain the experimental test, as well as the obtained results.

### 8.1 Experimental Test

The purpose of the experimental test was to determine the accuracy of OPAIEH for the activities inference. The accuracy is defined in equation 1.

$$Accuracy = \frac{Activities\ correctly\ inferred}{Total\ activities} \times 100. \quad (1)$$

Due to the feedback design of OPAIEHClient, its use by older adults would be complicated, for that reason the final user in the experimental test was a middle-aged person, with the ability to play OPAIEHClient without any problem.

In order to carry out the experimental test, some preliminary preparations were required: 1) place one sensor in each of the 3 rooms that the user established (Bathroom, Kitchen and Dinning Room); install OPAIEHClient on the user's smartphone; register the user on the platform, as well as register the rooms and activities,

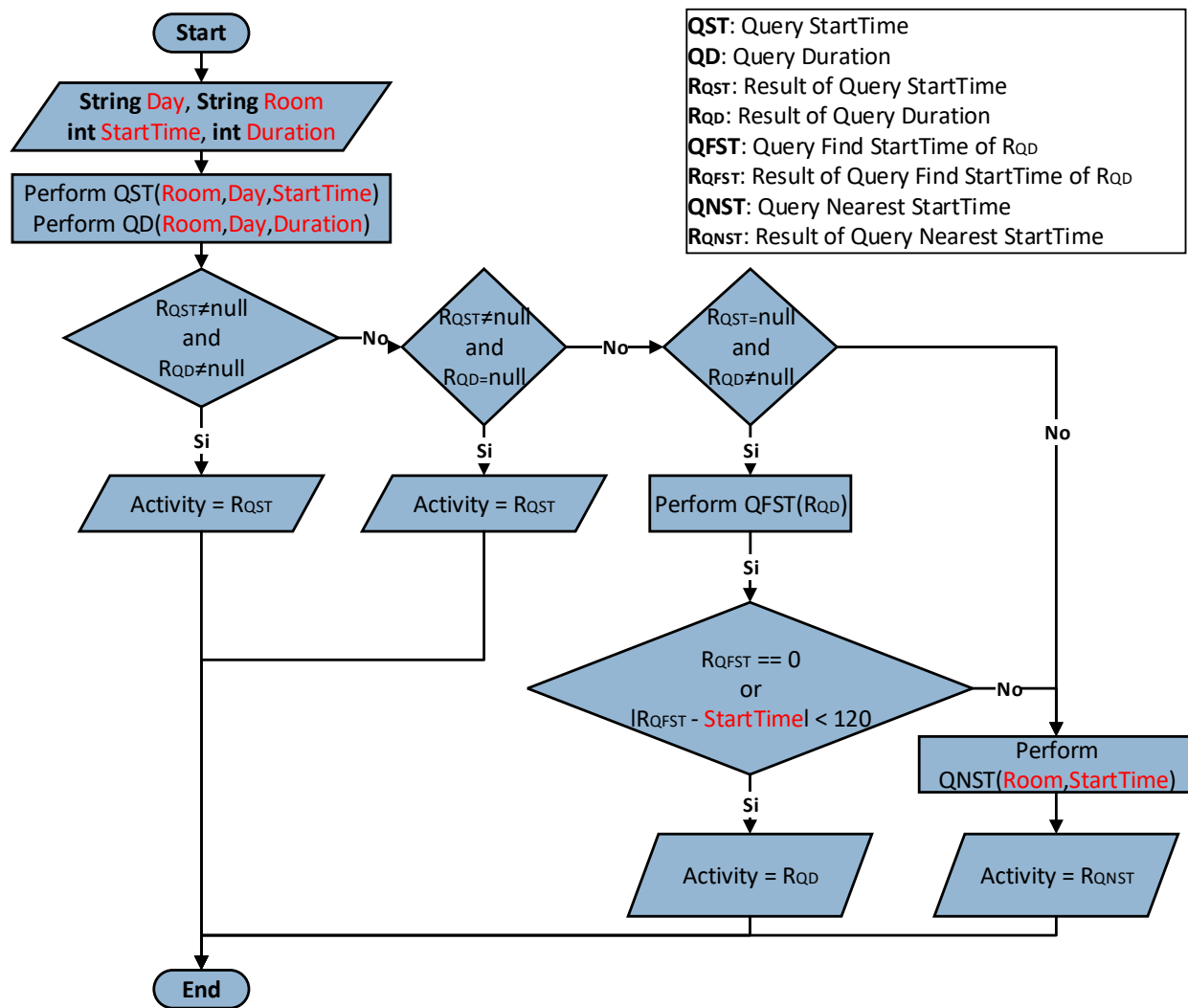


Fig. 14. Flowchart of the inference algorithm

the user registered 3 activities (Have breakfast, have lunch and take a shower) and the two predefined activities defined in this project was created (Toilet activities and Washing dishes) due the user registered the rooms Bathroom and Kitchen, respectively.

The experimental test consisted of the user having to use OPAIEHClient (monitoring and confirmation of activities) during 4 weeks (28 days) each time he was going to carry out some of

the activities that he registered from this mobile application, as well as the predefined activities.

### 8.2 Results

In order to better explain the results, it is necessary to have a good understanding about the structure of the table of confirmed activity, in specific the columns *Inferred activity* and *List activity*, based on these columns the activities can be grouped in three categories: 1) the activity was inferred correctly; 2) the activity was inferred incorrectly

but when presenting a list of activities to the user this activity was in it; 3) the activity was inferred incorrectly and it wasn't presented to the user.

The total number of confirmed activities after the 28 days of the experimental test was 147. Figure 15 shows the amount of confirmed activities by category.

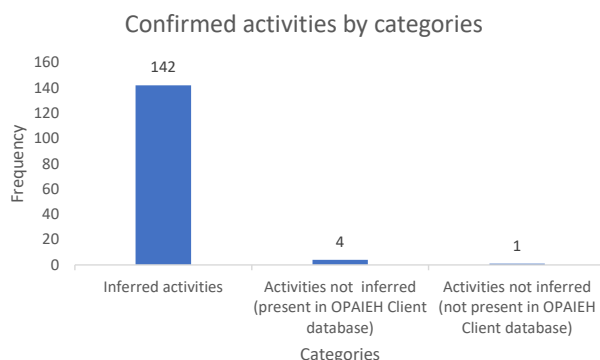


Fig. 15. Graph of confirmed activities

With the data provided by the Figure15 and applying the formula of the equation1, we are able to calculate the accuracy of OPAIEH for the inference of activities, the accuracy was 96.6%.

## 9 Conclusions and Future Work

In this paper we present the development of OPAIEH, an ontology based platform for activity identification of elderly at home. The use of OWL ontologies allowed to create easily and simply (thanks to the semantics they provide) a knowledge base (IAR-ONTO) based on the definition of activity proposed in this project, which allows storing the information about the activities that users perform inside their home. IAR-ONTO is the key piece of the activity recognition, since the inferences are made around this, by realizing queries in SPARQL format through the inference algorithm.

The results obtained by OPAIEH were satisfactory, since of the 147 confirmations made by the user, 142 were correctly inferred and only 5 were not, based on this it can be said that the accuracy of OPAIEH is 96.6%.

As future work, it is planned to make the interface of the client device more user-friendly so that it

can be used by older adults or use a smaller, portable device as a smartwatch. Also, this will allow monitoring throughout the day and not only for some specific activities, and therefore, we will obtain a greater number of records.

## Acknowledgements

We would like to thank to National Council of Science and Technology (CONACyT) and University Autonomous of Yucatan (UADY) for their funding support in the realization of this work.

## References

1. Abdulrazak, B., Malik, Y., Arab, F., & Reid, S. (2013). Phonage: Adapted smartphone for aging population. Biswas, J., Kobayashi, H., Wong, L., Abdulrazak, B., & Mokhtari, M., editors, *Inclusive Society: Health and Wellbeing in the Community, and Care at Home*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 27–35.
2. Al-Wattar, M., Khusainov, R., Azzi, D., & Chiverton, J. (2016). Activity recognition from video data using spatial and temporal features. *2016 12th International Conference on Intelligent Environments (IE)*, pp. 250–253.
3. Attal, F., Mohammed, S., Dedabrishvili, M., Chamroukhi, F., Oukhellou, L., & Amirat, Y. (2015). Physical human activity recognition using wearable sensors. *Sensors*, Vol. 15, pp. 31314–31338.
4. Bamis, A., Lymberopoulos, D., Teixeira, T., & Savvides, A. (2010). The behaviorscope framework for enabling ambient assisted living. *Personal and Ubiquitous Computing*, Vol. 14, pp. 473–487.
5. ChartJS (2018). Simple yet flexible JavaScript charting for designers & developers. Retrieved from <https://www.chartjs.org>; accessed August 2019.
6. D3 (2018). Data driven documents. Retrieved from <https://d3js.org>; accessed August 17th, 2019.
7. de Geriatría, I. (2010). Gerontecnología. *Perspectivas para el desarrollo de la investigación sobre el envejecimiento y la gerontecnología en México*, Secretaría de Salud, Mexico, pp. 180–184.

8. **Gu, Y., Quan, L., & Ren, F. (2014).** Wifi-assisted human activity recognition. *2014 IEEE Asia Pacific Conference on Wireless and Mobile*, pp. 60–65.
9. **Gu, Y., Ren, F., & Li, J. (2016).** Paws: Passive human activity recognition based on wifi ambient signals. *IEEE Internet of Things Journal*, Vol. 3, No. 5, pp. 796–805.
10. **Horridge, M. (2011).** A practical guide to building OWL ontologies using Protégé 4 and CO-ODE tools edition 1.3.
11. **Huang, X. & Dai, M. (2017).** Indoor device-free activity recognition based on radio signal. *IEEE Transactions on Vehicular Technology*, Vol. 66, No. 6, pp. 5316–5329.
12. **INEGI (2005).** Los adultos mayores en México. perfil sociodemográfico al inicio del siglo xxi. Instituto Nacional de Estadística y Geografía, México.
13. **INEGI (2010).** Perfil sociodemográfico de adultos mayores. Instituto Nacional de Estadística y Geografía, México.
14. **Maekawa, T., Kishino, Y., Sakurai, Y., & Suyama, T. (2013).** Activity recognition with hand-worn magnetic sensors. *Personal Ubiquitous Comput.*, Vol. 17, No. 6, pp. 1085–1094.
15. **Memon, M., Wagner, S., Pedersen, C., Beevi, F., & Hansen, F. (2014).** Ambient assisted living healthcare frameworks, platforms, standards, and quality attributes. *Sensors*, Vol. 14, pp. 4312–4341.
16. **Philipose, M., Fishkin, K. P., Perkowitz, M., Patterson, D. J., Fox, D., Kautz, H., & Hahnel, D. (2004).** Inferring activities from interactions with objects. *IEEE Pervasive Computing*, Vol. 3, No. 4, pp. 50–57.
17. **Popleteev, A. (2015).** Activity tracking and indoor positioning with a wearable magnet. *Adjunct Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2015 ACM International Symposium on Wearable Computers, UbiComp/ISWC'15 Adjunct*, ACM, New York, NY, USA, pp. 253–256.
18. **Seaborne, A. & Prud'hommeaux, E. (2008).** SPARQL query language for RDF. W3C recommendation, W3C. [Http://www.w3.org/TR/2008/REC-rdf-sparql-query-20080115/](http://www.w3.org/TR/2008/REC-rdf-sparql-query-20080115/).
19. **SIMILE (2018).** Timeline web widget for visualizing temporal data. Retrieved from <http://www.simile-widgets.org/timeline/>; accessed August 7th, 2019.
20. **World Health Organization (2015).** *World report on ageing and health*. World Health Organization.
21. **World Health Organization (2019).** Ageing and life course. Retrieved from <https://www.who.int/ageing/healthy-ageing/en/>; accessed August 2019.
22. **Zou, H., Huang, B., Lu, X., Jiang, H., & Xie, L. (2016).** A robust indoor positioning system based on the procrustes analysis and weighted extreme learning machine. *Trans. Wireless. Comm.*, Vol. 15, No. 2, pp. 1252–1266.

Article received on 29/10/2019; accepted on 11/03/2020.  
Corresponding author is Jorge Gómez Montalvo.