



AnAbEL: towards empowering people living with dementia in ambient assisted living

José Ginés Giménez Manuel^{1,2} · Juan Carlos Augusto^{1,2} · Jill Stewart^{3,4}

Published online: 27 November 2020
© Springer-Verlag GmbH Germany, part of Springer Nature 2020

Abstract

Ambient assisted living aims to support the well-being of people with special needs by offering assistive solutions. Systems focused on dementia increase the autonomy of people living with dementia by monitoring their activities. Topics such as activity recognition and specific solutions such as reminders and tracking users by Global Positioning System offer great advances in user safety and help them preserve a healthier lifestyle. However, these solutions are often addressed to secondary parties, providing them activity logs or alerts, but excluding the main user, the person living with dementia. Although the primary users are taken into consideration at some design stages using user-centred design frameworks, the final products tend to not fully address user needs. This paper presents an ambient intelligent system aimed at reducing this limitation by providing reminders and advice to the person living with dementia in the first instance. The system still involves caregivers if unusual or unhealthy behaviour continues. The solution is deployed in order to be validated by professionals from London city boroughs who work in housing and dementia related services, with an emphasis on enhancing healthy lifestyles by empowering the user in the early stages of dementia with autonomy. Through continued activity monitoring in real-time, the system can provide reminders and warnings to users to keep healthy routines. Continuous monitoring provides user behaviour tracking, and the context-aware logic used involves caregivers through alerts when necessary to ensure user safety. This article describes the process followed in developing the system, and covers previous concerns and practical feedback from health professionals over the deployment of the system in a real environment. Our approach also includes a novel indoor localization system to distinguish users and allows a more specific delivery of services in multi-occupancy scenarios.

Keywords Ambient assisted living · Smart homes · Dementia · Ambient intelligence · Activity recognition

1 Introduction

The current issues related to dementia are an important concern in today's society and involve several disciplines that address solutions to improve the well-being of People living with Dementia (PwD). The diseases associated with dementia are related to ageing mainly, which due to rising life expectancy, are growing and showing a worrisome forecast in future years [1]. One important concern about PwD are the self-harm behaviours which have a negative impact in their health. These behaviours can happen in a direct or indirect way, for example, performing basic activities of daily living (ADL) incorrectly or not performing them at all by omission. Some common ADLs such as eating, sleeping, drinking or bathing could be complicated to achieve by PwD. Others more directly harmful are related to mobility problems which can generate falls or cognitive deterioration which may lead to, for example, elopement (unexpectedly

✉ José Ginés Giménez Manuel
gbequer@gmail.com

Juan Carlos Augusto
j.augusto@mdx.ac.uk

Jill Stewart
j.stewart@mdx.ac.uk

¹ Research Group on Development of Intelligent Environments, London, UK

² Department of Computer Science, Middlesex University, London, UK

³ Centre for Public Health and Risk Management, London, UK

⁴ Natural Sciences Department, Middlesex University, London, UK

leaving the house). These can endanger PwDs well-being by getting lost in a city and other dangerous situations outside of the home at inconvenient times. Also the occurrence, performance, and duration of ADLs and behaviours can be important indicators of the PwD cognitive decline. Although it is not yet clear how they are inter-related, knowing these parameters can help to provide effective adjusted social care to the person's situation [2].

Although at initial dementia stages it is recommended that PwD remain at home, and it is preferred by them as well, they need help and support in daily routine activities as well as monitoring from caregivers in order to guarantee their safety. This situation has been discussed in terms of ethics since initial stages and middle dementia impairment persons are aware they need help but they do not want to lose their autonomy or be confined [3]. Thereby the value of autonomy versus the need to prevent harm and distress clashes with the relationship between caregivers and PwD. Many documented cases describe how caregivers' well-being is affected by stress from their jobs [4–6] and they ask for more public and institutional support.

Thereby it seems necessary to address this issue where autonomy versus dependency creates different conflicts in early stages of the dementia process.

1.1 Related work

Work in ambient assisted living (AAL) is experiencing rapid growth in connection to assistive technology and smart home environments, including new devices and techniques to support people with special needs, especially addressing PwD concerns. AAL work covers several interconnected topics, such as activity recognition (AR), the Internet of things (IoT), user-centric design (UCD) and co-design.

AR research presents many approaches to detecting ADLs in PwD's daily lives at home [7] and some addressed to nursing homes [8]. Regardless of where AAL systems are deployed, their goal is the continuous monitoring of problematic situations, with ADLs recognizing and categorizing unhealthy habits for clinical observation in order to design personalized interventions. The techniques used by these systems range from non-intrusive sensors [9–11] and wearable devices [12] to cameras and microphones [13]. Using unobtrusive sensors is considered most appropriate for PwD, since these sensors respect privacy and do not provide technical challenges for the user. Some new work using radio-frequency identification (RFID) tags has been conducted [10, 14] to categorize small activities such as picking up a certain cap or medicine jar. Other work such as [9] focuses on higher level activities such as movement detection in a room, turning an appliance on or off, etc. Although these present an easier architecture to deploy in a real environment, they are less accurate at inferring activities.

Other current techniques to improve activity detection and classification accuracy use various learning methods [15, 16] or ontologies [17] in a multi-sensor environment. Some newer devices and techniques improve AR in other aspects, for example, user localization inside a home or distinguishing the activities of other residents [18, 19]. In general, AR is improving in its accuracy of detecting activities in increasingly complex situations. Regardless of the pros and cons of each approach, the final aim is to display the information gathered to secondary users (caregivers, doctors, nurses or relatives) for analysis.

Other solutions focused on specific problems claim to prolong users' healthy lifestyles and autonomy, by alerting caregivers in the case of an anomalous situation. Examples include detecting falls [20, 21], tracking users out of home using Global Positioning System (GPS) [22] and, more user-centred, reminder solutions which interact with PwD by sending alerts to the user's mobile [23, 24] or other systems [17] in order to keep their daily routine healthy. In particular, reminders using mobile devices show positive results, helping users modify their behaviour when prompted [25]. Many commercial solutions can be found on the market, including items focused on specific cases of dementia such as reminders, adapted phones, watches or GPS [26].

Based on our analysis of previous AR techniques and approaches, we offer a scalable system which recognizes the most common situations of interest for PwD, takes into account other users in the same house, and gives pre-eminence to the guidance of the PwD in the style of a virtual lifestyle coach, keeping contact with secondary users mostly as a backup option.

1.2 Current AAL limitations

Although some current AAL solutions use a UCD approach to development, many are not aimed at the primary user, the PwD, as the direct user, but focus on secondary users such as caregivers, nurses, doctors or relatives. This limitation is present in solutions within AR and those which involve the user such as fall detection, reminders or GPS location. Although they are valuable in covering some problems, they are based on alerting secondary users to take charge of the situation, and reminders can often be ignored by PwD. Therefore, these systems do not guarantee that the user accomplishes an activity or task. Solutions which offer more certainty, such as cameras or microphones that infer user activity, face ethical concerns regarding privacy and are rejected by users as very invasive. However, they are finally addressed to the person who monitors the PwD, that is the secondary user.

The acceptance of technology used to support the elderly is currently increasing within that population [27], however

current systems seem to omit, in some ways, the primary users.

In general, these solutions provide important innovations and approaches to topics, but new developments are not exploited to include the empowerment or autonomy of PwDs as their first priority.

1.3 Motivation

Building on previous work in AR focusing on ADLs combined with ideas for reminders, the present work suggests an approach that empowers PwD in the early stages by increasing their autonomy within the AAL systems, making them the main recipients of the system advice.

We deploy an ambient intelligent (AmI) system able to detect anomalous dementia-related ADLs and behaviours performed at home and to warn, in the first instance, the primary user about the situation, for example skipping a meal. Since the system continues to work in the AR process, it can determine whether the user is revising that behaviour and performing the activity, for example by eating. In the case where the situation is not resolved by the primary user, the system can involve secondary users through an alert keeping them updated about the situation and possible risk, allowing them to make a suitable intervention if required.

Summarizing, the system proposed consists of continuous user monitoring using AR technologies to:

- Notify and remind the primary user about tasks, empowering their autonomy;
- Alert caregivers when needed, reducing their workload and ensuring primary user safety.

This work presents a system which aims to achieve ambient assisted empowered living: AnAbEL. This is an AmI system addressed to people with dementia or cognitive decline, focused on their well-being. The goals are to enhance PwD autonomy and coach them in preserving a healthier lifestyle.

This paper describes the general principles which guide this project as well as the deployment of the final system:

- Detecting ADLs and behaviours related to dementia, on time, using non-intrusive sensors that provide useful information about daily life routines;
- Implementing a user-friendly interface to configure the system, which allows setting and adjusting the system to each user and environment in order to evaluate activity;
- Providing primary user and secondary user interactions using technology to give notifications and reminders;
- Presenting a scalable smart system environment able to integrate newer technologies such as Bluetooth low energy (BLE).

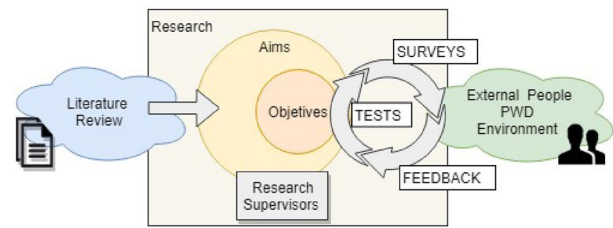


Fig. 1 Methodology used in this research based on U-CIEDP

2 Methodology

This work extracts initial knowledge from a literature survey of dementia health and AAL. This literature comes from many sources such as Alzheimer’s associations, MDX My Library search, Google Scholar and Basesearch. Recent work in AAL, smart homes and AmI gives great consideration to a user-centric approach to the design and development of systems and proves useful in defining methods. In particular, the present work is inspired by the user-centred intelligent environments development process (UCIEDP) proposed in [28]. We apply a UCIEDP methodology in a simplified version, as shown in Fig. 1.

2.1 Study design

An initial survey was conducted to define an initial system approach based on previous research. It addresses people with experience in dementia such as caregivers, relatives, nurses, doctors, etc. Using an on-line survey it was possible to reach more people, reduce costs and facilitate the data processing. In order to secure meaningful answers the survey was sent to various contacts at dementia research centres in London, and the invitation to take part was accompanied by a summary of the project. Finally, 14 surveys have been collected from people with more than one year of experience working with PwD.

The survey poses questions to help us understand, for example, which ADLs and behaviours are more commonly missed and important to monitor based on the respondents’ personal experience. Other questions ask which ADLs are easier for PwD to carry out without help, and their relationship with technology. Further questions ask about the role of caregivers and their years of experience in the position.

Additional feedback comes from prototype presentations to professionals from the UK health care and environmental health sectors, remarking on the usefulness and need for this sort of tool.

In summary, the concerns focus on privacy, security and well-being, while the expected gains focus on enhancing PwD autonomy and supporting caregivers.

The surveys, questionnaires and other methods in this research have been approved by the Computer Science Research Ethics Committee of Middlesex University¹.

2.2 System architecture

This section describes the basic parts needed to build the AnAbEL system as well as their design based on requirements from AAL, AmI systems and nursing work.

2.2.1 Detecting ADLs and behaviours (AR)

AmI systems have to be integrated and embedded in users' daily environments, providing ubiquity. They have to exploit the useful contextual and situational information from this environment (context awareness) [29]. In this research approach, ubiquity is realized using non-intrusive sensors which do not alter the environment and respect privacy concerns [30]. Sensors are a crucial part of context awareness, which is inferred by the temporal reasoning [31] proposed for this system. This provides an efficient logic implementation that retrieves environmental information and produces real-time outcomes about user activity.

In order to select a reasonable initial number of ADLs to monitor, the selection process is based on the most important activities. Eating, drinking, sleeping and bathing are considered more important to monitor than, for example dressing, since they are crucial for achieving a healthy lifestyle, but also because deviations in these habits are relevant measures of cognitive states related to disorders in PwD [30]. According to the survey described in Sect. 2.1, eating, drinking (hydration) and sleeping are considered the most important activities to monitor in PwD's daily routines, and so seem the best choice for initial consideration. They are also identified as the easiest to do by PwD without external help, which means the system avoids intervening in users' more complex activities which could affect their well-being.

Among the common behaviours related to dementia, this project focuses on those which imply physical activity. Other common behaviours associated with dementia, such as incongruent speech or repeated questions, are not considered. Wandering behaviour is an important behaviour to address [32, 33] since it is one of the most troublesome behavioural problems related to dementia. Our surveys find that more than 90% of respondents have witnessed wandering episodes in their personal experience.

Another common behaviour is leaving the house unexpectedly or running away from the building, called "elopement", which is important to control since it implies a

potential risk especially if leaving the house at unusual times. This behaviour is also pointed out by Lay et al. [32] and Steinhauer et al. [33].

The last criterion for selecting ADLs is that the system implemented should adapt to the current environment (see Sect. 3). The initial sensors available for this research are non-intrusive and can cover many types of detection, such as motion sensors, reed sensors for doors and windows, pressure sensors for beds, sofas or chairs, energy sensors for appliances such as table lamps, and switch sensors for room lights. The ADLs finally selected to incorporate into this initial system are eating, sleeping, wandering and elopement. Human behaviour, being complex to monitor using the current technology, which is a limitation of AAL systems, has been remarked upon by previous works such as [23]. Thus, this article considers ADLs and behaviours detected as merely pieces of evidence about behaviours, even if they are written in the form "the user is eating". However, the work by Steinhauer et al. [33] describes how reasoning about timing is able to offer a framework that reflects human activities. One problem derived from this kind of system is that strongly defined behaviours prevent user adaptation. This drives the development of a user interface to configure behaviour and obtain a system better adapted to the user's environment.

2.2.2 Interface

The user interface is important to define the AmI system characteristic of providing personalization and user adaptation [29]. Thus, we decide that the AnAbEL system would provide options that adapt to its users. We propose a user interface to configure parameters related to user activities, building characteristics and the system logic.

Each user performs their daily routines at a different time, and the time spent executing activities also differs, thus the initial interface approach focuses on covering these variations. The first aim is to allow the user to configure daily timetables, wherein each activity usually happens, assuming the opposite situation to be unusual and worth attention. It is important to understand why ADLs and behaviours are healthy/unhealthy, usual/unusual or safe/risky, so that the system can provide support. It is accepted that ADLs must be adjusted to user routines and that deviations in the time they are carried out or the time spent performing an activity can be symptoms of something going wrong [30, 34].

Consequently, once timetables are provided for each activity, the system needs to assess under what circumstances to warn the user. For example, if the user usually goes to sleep around 10 pm, but if one day the activity happens at 11:30 pm, this behaviour can be considered unusual, and could mean the user is feeling disorientation or anxiety [30]. However that conclusion is left for a qualified

¹ <https://unihub.mdx.ac.uk/study/spotlights/types/research-at-middlesex/research-ethics>.

caregiver who can appraise the situation. Also, an activity could happen at an unusual time. Leaving the house during the day could be normal for some users who go for walk, but if this action happens at an unsafe time, such as midnight, it could be related to user disorientation, then the system should take action by comparing the activity with the user's schedule. Timetable configuration is an essential element of user personalization.

Threshold parameters define when the AnAbEL system intervenes, by warning the user and alerting the caregivers upon detecting unusual activity. For example, depending on the user's habits, the user might start the activity later or spend more time than other people on the activity, but it can be perfectly well carried out without system warnings or alerts. Thus, two sorts of thresholds are proposed for each behaviour: one indicating when the system has to warn the primary user about unusual activity, and the other to alert the secondary user if the primary user has not resolved the situation alone.

As each person has a different manner of communicating, depending on culture, environment, personality, cognitive impairment, etc, an adaptable accessory is needed to offer a proper prompt to the user [35, 36]. Some initial ideas include using sounds (music or familiar voices) or lights to interact with and guide the primary user, but we finally chose the mobile device described in the next section. For this reason, the AnAbEL interface has the option to select the means of interacting with the primary user and can be configured to get feedback from the primary user in order to fit the communication context and reduce misunderstanding and PwD anxiety [36, 37].

A parameter suggested by wandering detection relates to the user's physical condition and house design, that is the time taken by the user to go from one room to another (room distance). Other parameters cover other areas of user customization based on lifestyle or preferences (schedules and question/answers), the system logic (time to issue an alert) and the system adapting to the environment (time spent going between rooms).

2.2.3 System actions

One AmI characteristic is being anticipatory [29], which means the system intervenes without the user's deliberate mediation. Thus, AnAbEL's takes anticipatory action addressed to the user to achieve some task autonomously. Once user activity is detected, assessed and catalogued as "healthy/unhealthy or usual/unusual", what can AnAbEL do to improve user autonomy? It seems reasonable that by reducing caregiver surveillance and intervention, it enhances user autonomy. To achieve this, AnAbEL should cover basic caregiver tasks, preventing PwD harm.

In the initial stages of dementia and cognitive impairment, preventing harm by reminding the user to do tasks or to stop an unusual behaviour is crucial. Nursing work shows how using correct words with PwD can be effective to remind them of a task or amend a behaviour [7, 18]. Reminder systems show good results in rectifying user behaviours through mobile messages. Natural language interaction between the user and the system is difficult, due to the limitations of current technology. Advances in the Internet of Things, artificial intelligence and language processing are promising, and open a huge number of possibilities. Some research describes interventions with PwD using artefacts such as lights or sounds [37, 38], although, always under the supervision of the caregiver who can control the reaction. These actions can be performed easily in current smart environments which are able to control lights and smart devices such as TVs, radios or other appliances. Even when a primary user warning system is effective and increases PwD autonomy, they cannot lose the security of continuous caregiver monitoring. Hence, AnAbEL takes the caregiver into consideration by enabling the system to alert them when needed. Figure 2 shows the whole process flow of the AnAbEL system.

As represented by (5) in Fig. 2, all information gathered from ADLs, behaviours, alerts and user feedback are stored. This way, AnAbEL provides a wealth of information, as do other AR systems addressed at professional assessment, tracking the evolution, adaptation or modifying of routines or interventions for PwD.

3 System infrastructure

Previous sections describe some of the key features and concepts of the system in a generic way. The next section explains the technical aspects at a higher level of detail, as well as the environmental infrastructure wherein the system is deployed.

3.1 Intelligent environment

AnAbEL deployment has been performed within the Smart Spaces lab at Middlesex University. Part of the Smart Spaces lab is set up as a smart house for research and experiments on sensing supported systems by the Research Group on Development of Intelligent Environments². Figure 3 shows an accurate map of the lab with hardware elements installed inside, such as a server, distribution of used sensors and the smart hub (Vera Plus model³). This system is developed

² <http://ie.cs.mdx.ac.uk>.

³ <https://getvera.com/products/veraplus>.

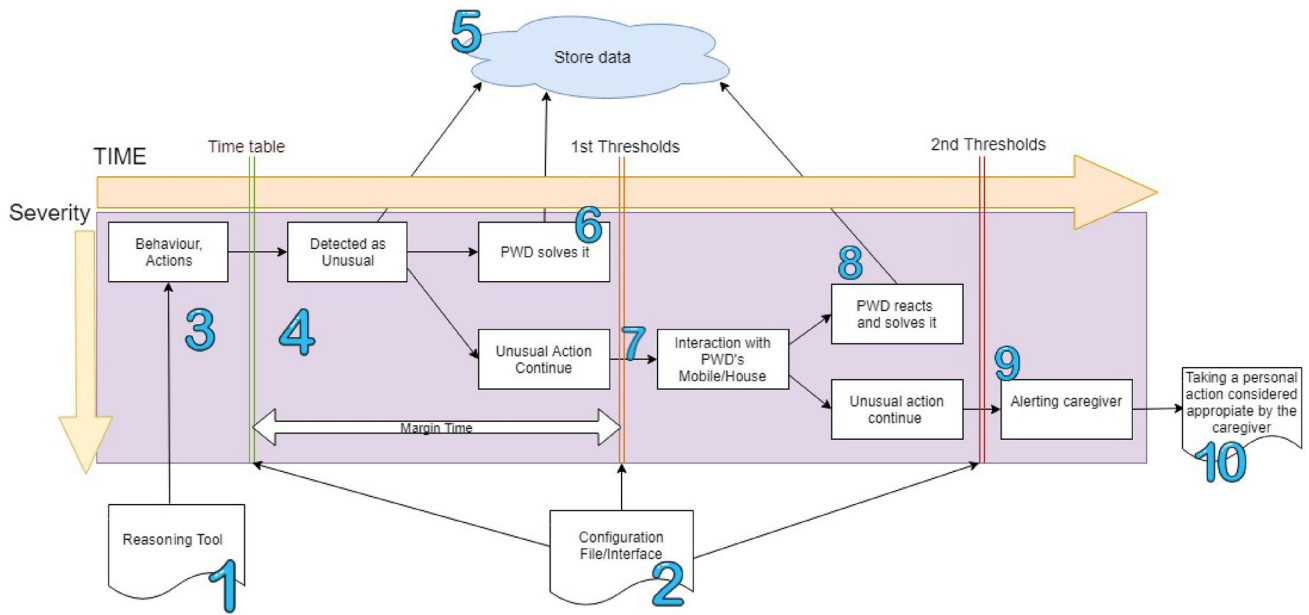


Fig. 2 Overview of system flow. The system (1) works on AR (3) using user configured timetables (2). When the system detects something unusual happening (4), either the user resolves the situation before a certain time (6) or the user continues the unusual behaviour

(7), then the system notifies the user. The user may react to the alert and change the behaviour (8), but if not, the caregiver is alerted (9) to take human action (10). The server (5) is where all activities and behaviours are stored

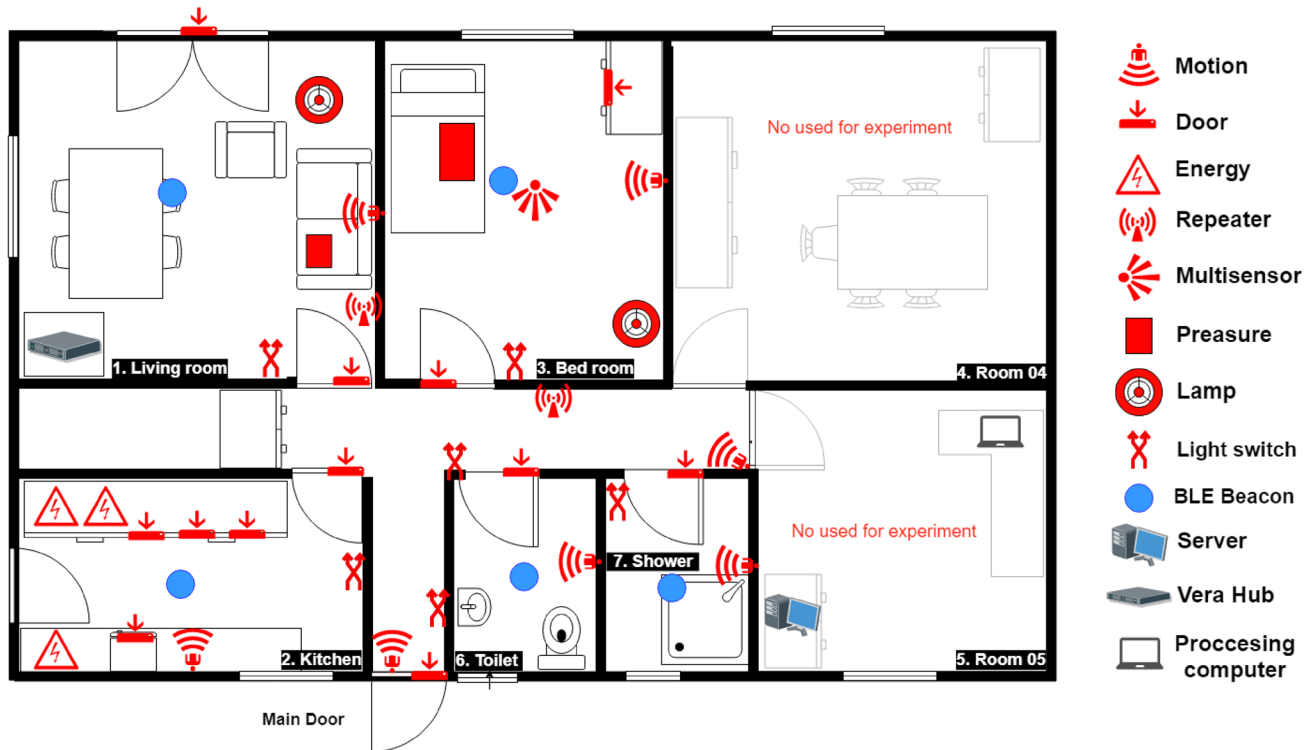


Fig. 3 Lab map with sensor deployment. In this research, the processing computing is integrated into the server

based on the SEArch architecture that is deployed in the lab. Although this architecture is described in Augusto et al. [39], the next section explains the parts related to this work.

3.2 Sensing environment

The lab is equipped with a Vera Plus hub device to manage the main sensing environment. The hub provides a Z-wave network to connect devices and request properties (state, battery level, etc.) as well as to change their properties.

Devices can be sensors or actuators depending on the function. Sensors are those devices which can transform a physical dimension into a digital signal, for example the presence of light into a digit 1, and actuators are those which can transform a digital signal into a physical dimension, for example a digit 1 commands a light to be turned on. We define the signal value in both cases as a sensor state, which represents a meaningful value about an environmental event. For example, a motion sensor has a trigger property which defines the state of the sensor, so, for example, when it detects movement in a room this property takes the value 1, and 0 in the case of no movement detected. Changing an actuator state means modifying the value of this property, for example, turning a light OFF or ON implies modifying the trigger actuator property to 0 or 1, respectively. Note that an actuator can also be configured so that devices can work as sensors and actuators in the same scenario, for example, the lights can be modified by a user, giving information about user action (sensor), or by the system when it sends a command to turn light OFF/ON as a response to a previous event (actuator).

The devices installed in the lab and managed by Vera are:

1. PIR sensor: This detects movement using infrared variations. The state of this sensor can be 1 = movement detected or 0=no movement. PIR sensors reset automatically to 0 a certain time after they are triggered. This value can be modified using the Vera interface. For this project, all PIRs were set to the minimum time allowed to reset (5 s). This configuration seems reasonable to manage them since, by using a temporal reasoning application, it is easier to determine whether the user is not in a room anymore using this tool than waiting for PIR default reset.
2. Energy device: For whatever appliance is plugged into this device and connected to the electrical installation, when the sensor trigger property is 1 (ON) it lets the electricity pass and the appliance work. However, the appliance can be OFF and the device ON giving a false positive of the real state. Therefore, the system requests another property; it checks the Watts usage to determine whether the appliance is working (ON) or not (OFF).

This is a case of the “state” concept at work. The device works as a sensor and actuator.

3. Reed sensor: This sensor consists of two separate magnetic pieces. When they are in proximity, the internal circuit is closed and the value of the property state is 0. If the pieces are separated the sensor state value is 1. These sensors are used for the doors and windows of rooms, cupboards, lockers or fridges, to determine whether they are open or closed.
4. Bulb device: This works as sensor and actuator. As a sensor it gives info about whether the bulb is ON or OFF, and as an actuator it can block or allow the electricity to pass to the bulb.
5. Pressure sensors: The original pressure pads were modified by adding a reed sensor to connect with Vera throughout the Z-wave network. The sensor detects the pressure of somebody standing on it, and is used on beds, chairs or sofas.
6. Switch sensor: This works similarly to a bulb sensor, but installed on the walls like normal light switches. They monitor the state of the light and can change the value (ON/OFF).

Figure 4(7) shows a tagged BLE beacon. These are not Vera Z-wave sensors, but provide environmental information through a communication protocol different to that used by Vera. Section 3.6 explains this technology.

3.3 Server

The lab hosts a server to manage the databases, web API and reasoning applications used for AR functions. The operating system running the server is Windows 10 and it uses Internet Information Services (IIS) as a web server. MySQL is the database management system installed on the server because, among other reasons, it is open-source and offers the scalability and flexibility that research development needs. The database is used by the various systems working in the lab to retrieve and store data.

The web server manages a RESTful API developed in PHP which provides a layer between external applications and databases allowing it to manage requests. It hosts web pages for various applications such as the user interface for this project. Despite the research being carried out in a closed environment, connections between the server and external elements use HTTPS with a self-signed certificate (used for deployment stages), as well as database encrypted with user passwords and other basic measures related to security such as a firewall, operating system and IIS. The implementation of security measures is aimed at developing an environment as real as possible.

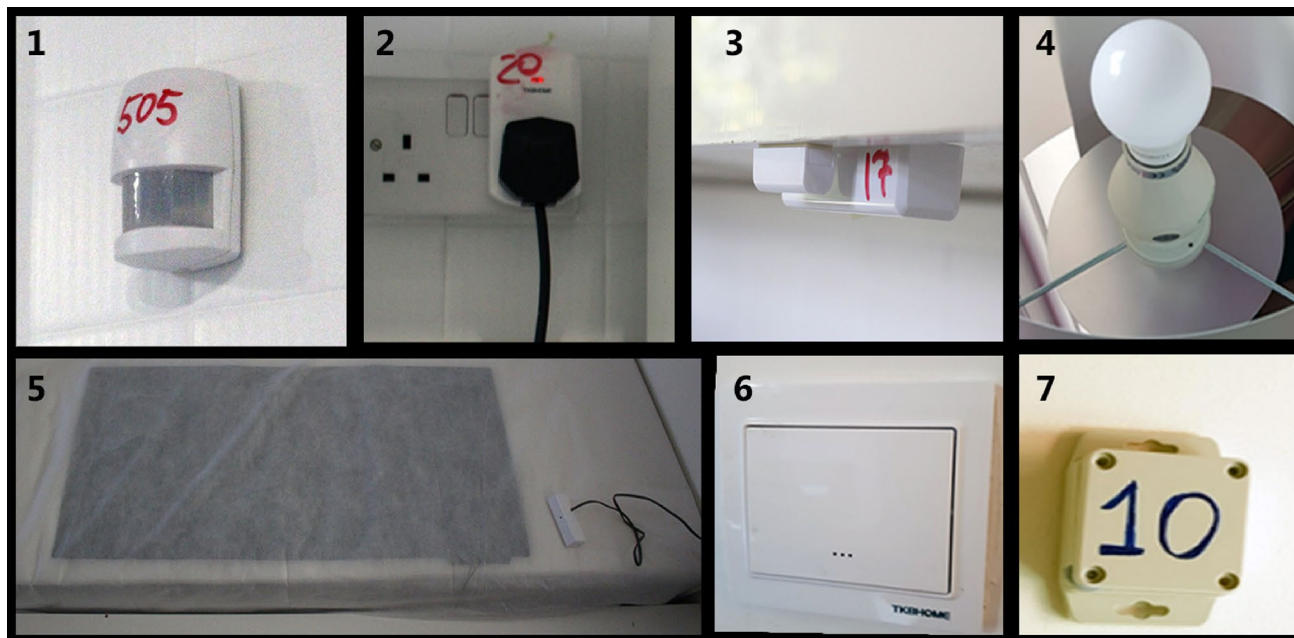


Fig. 4 Examples of sensors installed in the lab

3.4 MReasoner

The application used in this project to perform the activity recognition task is MReasoner(MR) [40]. This reasoning tool is developed in Java and provides the mechanisms to retrieve and infer information from sensing environment in real-time. This characteristic allows to define more accurate actions or activities which are a limitation in other AR systems which have a poor representation of time as Hoey et al. work explains [11].

3.4.1 MReasoner language

MR is a rule-based temporal reasoner. Its language allows rules to be triggered based on conditions met at specific times or lasting for some length of time. In our system, there are conditions related to the states and their changes captured by the sensors. The MR rule structure is *SSR((antecedents)-) consequent*: when the conditions in “antecedents” are TRUE the rule is triggered after which “consequence” becomes TRUE. SSR type rules apply this effect immediately at a logic level. MR allows another type of rule with delay effects, however these are not used in this project. The MR atomic element (variables) which form “antecedents” and “consequent” are called “states”. Since there are various layers in the system, these should be not confused. For example, MR state and Vera state, although related, are different. While in Vera, a state is the value of a property, in MR it is the name of a variable.

We say they are related because MR is able to associate a state with a sensor, then the value of the state in a code execution depends on the value of the sensor. MR distinguishes between two types of states:

- **Independent states** are those which do not change their value as consequent in a rule (e.g. motion sensor is represented by an independent state which sets its value in a function of the motion sensor value, but this value is independent of any MR conclusion).
- **Dependent states** are those states which are the consequent in some rule, but also can be antecedents in other rules. For example, “if movement is detected in the kitchen then the user is in the kitchen (userInTheKitchen)” and “if the user is in the kitchen then put the kettle on”. This example is not very practical but illustrates that “userInTheKitchen” is a dependent state.

The next example illustrates the translation of a basic action into a rule: “if movement is detected in the living room then turn on living room light” and “if no movement is detected then turn off living room light”.

```
SSR((MotionLivingroom)- > LigthLivingroom);
SSR((#MotionLivingroom)- > #LigthLivingroom);
```

MReasoner manages time conditions for a state in different ways based on the present assessment time. This means, an antecedent state can be evaluated across a period. The operators to manage time are:

- Absolute time coding by the operator “[-]”, e.g. light is on for the last 30 s which is translated as [-][30s.] LightOn;
- Relative time coding by “(-)”, e.g. Light was on at least once during the last 30 s: (-)[30s.]LightOn;
- Time interval: the previous operators can be used with time periods. For example: “between 7 PM and 8 PM the light is on” is similar to:

SSR((-)[-][19 : 00 : 00 – 20 : 00 : 00] lightOn)- > ...)

These are the basic operators of MR, but it provides many other commands, some of which are explained in future examples where used.

3.4.2 MReasoner working

MR polls the external systems each second, requesting the current state of the values declared in the rules. MR examines the Vera system each second, getting an updated picture of the whole home situation. The states’ values are updated in the process and, according to the rules which model the activity, produce conclusions. These conclusions can change internal states or actuators which modify the Vera sensor values. Since IoT solutions are growing and offering new technologies covering new issues or improving previous ones, this AmI system should offer an easy way to add more systems, enhancing its scalability. For example, this project poses the need to distinguish the primary user from other house occupants, but Vera does not support any technology to do this. The challenge of adding a new sort of device, distinct from Vera, with that aim, such as BLE localization, is resolved by using ad-hoc middleware (MW) developed for this project. The MW allows MR to manage other technologies and systems in an easy way. Actually, the MR-Vera communication is managed through this MW.

3.5 Middleware

The approach for MW development in this project is based on URL requests to various applications to get or set information, regardless of the platform or language used. This MW is based on XMPP⁴ protocol ideas but adapted to this environment.

Most commercial systems offer protocols to access information stored in databases or files, for example Vera is accessible using HTTP commands. Other non-commercial systems demand some development to save or get data which implies the use of a database. In this case, the database is easily accessed by developing a simple public API. The proposed MW can retrieve data through an API from

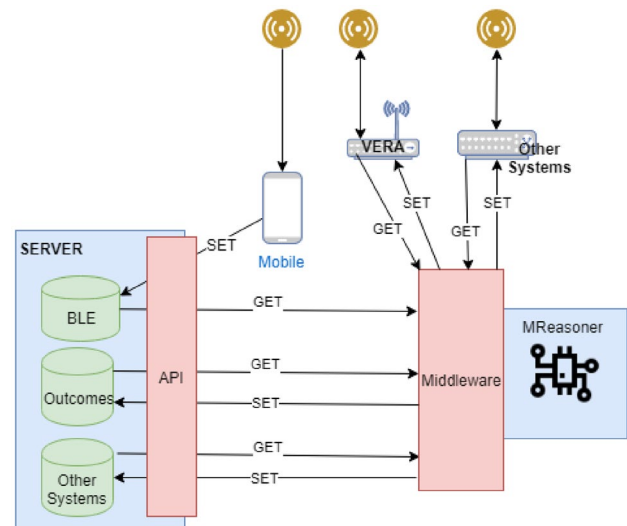


Fig. 5 Middleware diagram

many different systems, as can Vera which provides the API. However, the BLE technology used here just provides the board device to emit a Bluetooth signal, and thus it is necessary to make adaptations such as including a database with the current user position (see the BLE section). This offers access to the databases through an API, meaning the MW can retrieve the information easily. Other sorts of systems can be added to the MW and can be polled for information, adding functionality to the system. Figure 5 shows a schematic of this environment.

The MW does not currently have an interface to facilitate the addition of new systems. It offers several abstract java classes which allow the addition of a new system class to manage it by implementing basic methods to “get” and “set”, among other things, the attributes of IP, URL format, services, etc. The addition of a new system class loads the basis configuration necessary to request information. Once the new class is defined to communicate with the new system, each sensor belonging to the new one has to be defined in the database, although there is an inherited method which allows all sensors provided by the new system to be loaded automatically. The information provided relates to the properties of each system.

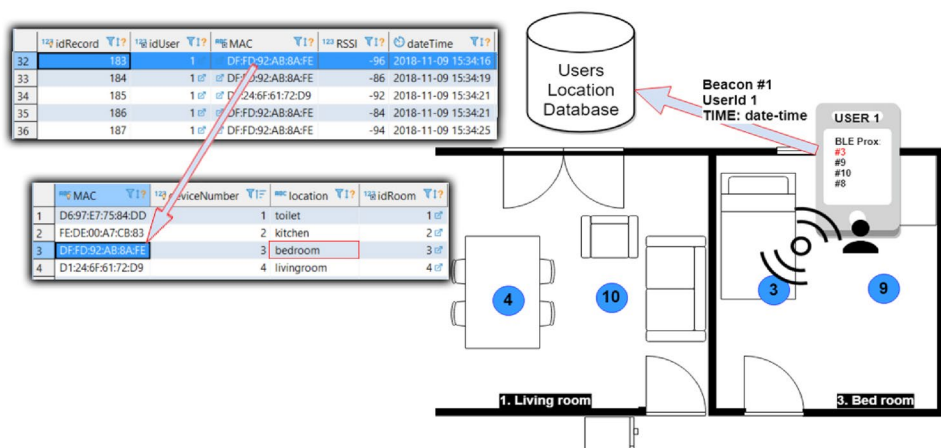
Although the MW looks limited because no interface supports it, other systems have been tested working together. The MW is used to incorporate AnAbEL into the user’s indoor localization, based on BLE beacons.

3.6 Localization system: BLE beacons

Although in the UK 60% of PwD live at home and around 14% are estimated to be living alone, this work uses multi-user architecture for a realistic environment, to account for

⁴ <https://xmpp.org/>.

Fig. 6 General illustration of localization system using BLE beacons and user's mobile



the presence of caregivers, relatives, friends and the 86% of PwD not living alone [1]. Trying to cover the whole spectrum makes it necessary to develop a system to distinguish between the residents. These limitations are present in many ARs which points to necessary future work. Several works show methods or technologies to get an accurate user's position indoors which also allow differentiation among users performing tasks by examining the proximity to other devices. Ultra-wideband (UWB) seems to be the most accurate technology for this, but implies an additional device such as wearable sensors to connect the user's mobile and UWB device [18]. Other technologies use pre-installed WiFi networks around the environment to avoiding installing new devices [41]. Since there is no consensus on whether WiFi or Bluetooth devices give better results, this project implements BLE beacon technology, based on Sora et al. work [42].

The BLE beacon deployment around the house, as well as a related Android application developed for this technology to detect the user, were tested with several configurations of beacon placement. The main BLE problem faced is the variation in beacon signal strength caused by objects such as walls, lockers and, more importantly, the user's body. This issue causes changes in the nearest BLE detected by the Android device, even if the Android device remains still, and generates continual jumps between the nearest detected beacons.

The most stable configuration found is two beacons per room separated by about 1–1.5 m. This way, the jumps happen statistically more often between beacons in the same room. Even if the mobile device detects a more powerful beacon signal for a time from another room where the user isn't, this time is insignificant compared to the time for the two BLEs in the same room. Thus, it is possible to create an effective filter and clean these outliers. In addition, the Android application checks the beacon signals using the device's accelerometer only when the user is moving, reducing the non-significant data originating from the

jumps. Figure 6 summarizes how the localization system implemented in the lab works.

4 System development

Although MW and BLE beacon systems were developed as part of the AnAbEL project, this section focuses on those system parts which were mentioned at Sect. 2.2. Since these components are strongly interconnected, order of the following sections is not related to how they were developed but they are explained to provide a better understanding.

4.1 Managing contextual information

As a user performs normal activities inside the house, sensors are activated according to the activity. Information from the Z-wave sensors is managed by the Vera hub and information from the BLE beacons is managed by the user's mobile which stores the current user location in the database. MR uses the MW to retrieve data from the various systems and uses that data to update the states which drive the rules. The consequence of the rules related to the activities is saved in the "Outcomes" scheme. Thus, MR connects to Vera, BLE and Outcomes.

The Outcomes scheme is formed of an API and database, wherein are defined the information and structure related to the activities eating, sleeping, wandering and elopement. Here, the defined states used by the MR rules are managed as the Vera sensors are through the MW. Since Outcomes is added to the MW, when MR manages states from Outcomes, the changes are updated automatically, as are the Vera actuators (see Fig. 5).

For example, the activity "eat" is defined in Outcomes, which represents whether the user is eating or not. So "eat" is a state name used by MR in the rules which sets "eat"

Fig. 7 Screenshot of outcomes table with significant events saved

datetime	ABC context	ABC type	ABC state	123 value
2019-05-07 22:30:00	sleeping	2	sleepAlertUser	1
2019-05-07 20:46:42	eating	1	eat	0
2019-05-07 20:16:28	eating	1	eat	1
2019-05-07 19:15:55	elopement	3	ElopementAlertCarer	1

Fig. 8 Screenshot of the states table wherein the Outcomes scheme is defined

ABC state	123 type	ABC context	ABC text
ElopementAlertCarer	3	elopement	[NULL]
ElopementAlertUser	2	elopement	[NULL]
eat	1	eating	[NULL]
elopement	1	elopement	User goes out after hours
goIn	1	elopement	User is going in
goOut	1	elopement	User is going out
irregularSleep	1	sleeping	[NULL]
mealAlertCarer	3	eating	[NULL]

=1 when it detects the user is eating, and saves it in the database (Fig. 7).

The states are defined in the table “states” in Outcomes, with further attributes. This table is used by the MW to load those used by MR in the rules. Figure 8 shows an example. A “state” belongs to a context, “eating”, “sleeping”, “wandering” or “elopement”. It keeps the “states” grouped by activity or context, helping the subsequent process of analysis by producing statistics and charts. A state can be set as “state” (internal state), “warnUser”, “alertCaregiver” or “userState”. Thus, the “eat” state can be considered an internal state because it is just updated and saved (it can be used for graphs and statistics as we show further on), while “warnUser” and “alertCaregiver” are types which indicate that the system should generate alerts. For example, assuming the user is not going to sleep during the time set on interface timetables:

```
SSR((sleepSchedule ^ #sleep)->sleepAlertUser)
```

The change in the consequent “sleepAlertUser”, defined as a “warnUser” type, is used by the system to know if it has to warn the user. If the user leaves the house at an unusual time that triggers “elopementAlert”, which is defined as an “alertCaregiver” type. This means the system alerts the caregiver when the “elopement” state is set to 1. Lastly, “userState” represents feedback from the user, and this is associated with user responses from the mobile device.

Note, at this point, the states can be called different things and be updated depending on different criteria, but updating a state name in Outcomes implies updating the text in

the rule, otherwise MR does not relate the state name with Outcomes.

The Outcomes log stores the changes of state by saving the “state” name, the context, the type, the new value, the time when it changed and an extra “info” field used for user responses.

4.2 User interface

The interface is developed to be easy to manage and understand. It is developed using HTML5, JavaScript and the JQuery library, which dynamically generates the interface by requesting the interface database.

Since the interface can be configured by the primary user, we apply design recommendations focused on being clear and easy to understanding for the elderly, including the font style and size, layout and elements of colours and contrast [43]. For example, black letters on a white background were chosen initially, and the opposite to distinguish sections for high contrast. Since it is likely a secondary user will configure the system using the interface, it should not be difficult to understand, assuming the secondary user does not have strong knowledge of managing applications. Figure 9 shows the basic parameters, such as schedules, thresholds and responses.

Since the system stores the information about the ADLs, it is important to show the activity information in an understandable way. The interface provides a tab which displays an ADL graph developed using Highchart.js library. Figure 10 shows one possibility for how the information can be shown. This graph displays data grouped by ADL or behaviour across time and by default loads the user’s timetable from the interface as schedule. These intervals are shown

User Interface V0.1

MONITORING
EATING
SLEEPING
ELOPEMENT
WANDERING
Save
Cancel

Schedules

From to ✕

From to ✕

From to ✕

Add period

Message

Message to user

Reponses available to the user

✕

✕

✕

✕

Add response

Alerts

Alert to caregiver after seconds

Alert to user after seconds

Fig. 9 Interface with an example of “eating” activity configuration

User Interface V0.1

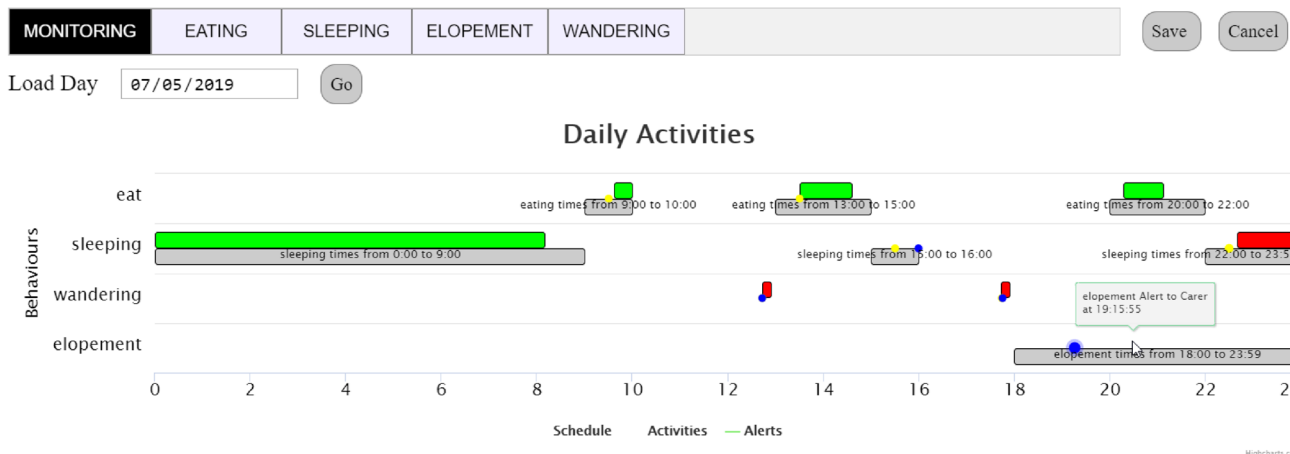


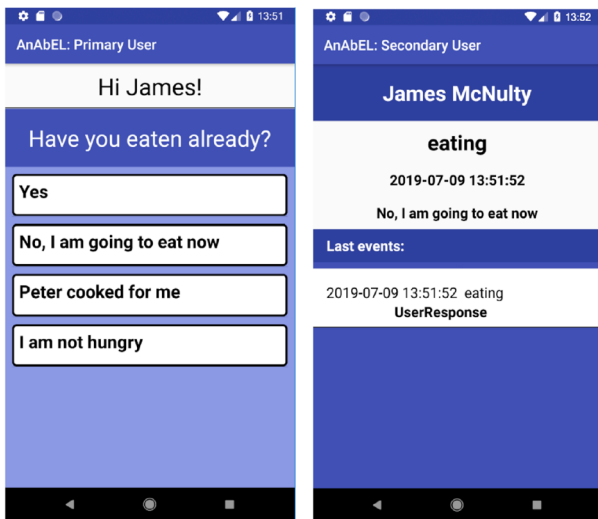
Fig. 10 Interface tab showing the daily activities of the user

by the grey bars in Fig. 10. The information related to daily activity recognition is loaded by selecting a day, and overlaps a little onto the previous one. This information is represented by a green background when activities are done inside the usual times, and red when there is some deviation from the normal routine. It displays warnings to the user with specific points on the graph, and a different colour for alerts to the caregiver. Figure 10 shows warnings in yellow and alerts in blue. This helps to evaluate if they happen at an appropriate time and whether the warning system is coaching the user. More statistical graphs and charts can be extracted according to the requirements of the users, and once the sensitive information is stored, it is easy to create new forms of data display. For example, Lazarou et al. [7] shows interesting graphs and charts.

4.3 Mobile application: interacting with users

Mobile technology is mentioned in a previous section as a good way to interact with the user, although other alternatives could be considered, such as using music or lights. Nevertheless, as Orpwood et al. [44] points out, PwD struggle to learn new technologies, so the devices used should be familiar. Since the elderly population in developed countries are familiar with mobile phones [27], it seems reasonable that AnAbEL should rely on mobile technology for warnings and alerts.

The mobile application (APP) design is simple and focuses on ease of understanding, by providing basic information. Since the layout is the biggest challenge facing



(a) Primary user GUI. (b) Secondary user GUI.

Fig. 11 **a** APP GUI interface for primary user showing alerts when the system registers the state “warnUser” with value 1; **b** secondary user interface showing an alert when primary user responds to a question or when the system sets the state “alertCareviger” to 1. It also keeps a scroll list which shows the latest alerts received

applications designed for PwD, the main concern is showing information as clearly and completely as possible.

The APP offers the possibility of users logging on and showing a different interface (GUI) depending on the user’s role (primary user or secondary user). If it is launched as primary user, the phone receives alerts related to the activity by showing the text message for this activity in the user’s interface (Fig. 11a) and the list of predefined responses. When the user selects a response, it is sent to the server and creates a row type “userState” in Outcomes (Fig. 7). When the APP is launched as a secondary user, the phone receives an alert with the primary user’s response (Fig. 11a). This way, the secondary user can evaluate the situation based on his/her own experience and knowledge of the primary user’s daily life and characteristics. Regardless of whether the primary user has given a feedback, if the unusual situation continues the system creates an alert for the secondary user when the threshold time “alert caregiver” has elapsed (Fig. 11b), and the information about the activity is loaded by the secondary user’s phone as an alert. The secondary user’s screen displays a list of the recent events that have occurred, giving a wider picture of the situation.

4.4 Activity recognition and assessment

Although the previous section explained how the MReasoner rules work, the artificial intelligence involved in the process could seem complex. The next section explains how ADLs and behaviours are transcribed into rules following some

ideas from previous work such as Tran et al.’s[45]. Among the many combinations of rules which can be modelled, the situations described here are simple, but it can become complex by adding more possible scenarios for each activity. Besides, the SEArch architecture [39] provides a learning process to generate rules based on real user activities [46]. Thus, this system working in a real environment could create rules for each user.

These rule clusters address scenarios of practical usefulness, and provide an understandable way for developers to write their own rules. This task can partially be achieved by automated machine learning techniques, however, in our opinion, they still need a human in the loop to ensure their correct final deployment. There are pros and cons to this approach as with other AR work, but comparing methodologies at such a high level is not part of the present document. AnAbEL achieves a satisfactory performance to the extent that the SEArch architecture, from which AnAbEL’s reasoning core is formed won the 2019 British Computer Society Machine Intelligence Competition⁵. In this competition our team presented AnAbEL, together with the learning and preference systems within the SEArch architecture.

4.4.1 Eating activity

An approach to obtaining “evidence the user is eating” could be: the user is in the kitchen using some appliance such as microwave or kettle, and opening the fridge or a closet with food. This translation of the “eat” activity into MReasoner rules, including user position using BLE (called here “userKitchen”, which means the user’s localization is detected in the kitchen) is:

1. SSR((Cupboard4 ^ KitchenMotion ^ userKitchen)->eat);
2. SSR((Cupboard6 ^ KitchenMotion ^ userKitchen)->eat);
3. SSR((FridgeDoor ^ KitchenMotion ^ userKitchen)->eat);
4. SSR((Microwave ^ KitchenMotion ^ userKitchen)->eat);
5. SSR((Kettle ^ KitchenMotion ^ userKitchen)->eat);

Another important point is the needed to reset the “eat” state, which means the user is not eating (#eat) or finished eating. This logic is based on an estimated time included in the rules that indicates the user is not in the kitchen doing something related to eating any more. In this case,

⁵ <http://www.bcs-sgai.org/micom/pastcomps.php>.

it is assumed that one minute after all sensors involved in “eat” do not show activity and the user position is “not in the kitchen”, it could be evidence that the user has finished eating:

1. `SSR((-) [60s.]#Cupboard4 ^
[-] [60s.]#KitchenMotion ^
#userKitchen)->#eat);`
2. `SSR((-) [60s.]#Cupboard6 ^
[-] [60s.]#KitchenMotion ^
#userKitchen)->#eat);`
3. `SSR((-) [60s.]#FridgeDoor ^
[-] [60s.]#KitchenMotion ^
#userKitchen)->#eat);`
4. `SSR((-) [60s.]#Microwave ^
[-] [60s.]#KitchenMotion ^
#userKitchen)->#eat);`
5. `SSR((-) [60s.]#Kettle ^
[-] [60s.]#KitchenMotion ^
#userKitchen)->#eat);`

Now the system can monitor whether the user is eating and when he/she finishes, which are stored in Outcomes and shown in the interface monitoring graph (see Fig. 10). Afterwards it is necessary to evaluate the activity “eat” and catalogue it as usual or unusual within the eating context. Next, the rules show the periods designated for the user to “eat” in their daily routine which are loaded from the interface and converted into rules by MR (see Fig. 9):

1. `SSR((clockBetween(00:00:00–06:59:59))
-> #eatSchedule);`
2. `SSR((clockBetween(07:00:00–9:00:00))
-> eatSchedule);`
3. `SSR((clockBetween(13:00:00–14:30:00))
-> eatSchedule);`
4. `SSR((clockBetween(14:30:01–18:59:59))
-> #eatSchedule);`
5. `SSR((clockBetween(19:00:00–21:00:00))
-> eatSchedule);`
6. `SSR((clockBetween(21:00:01–00:00:00))
-> #eatSchedule);`

Here, the command `clockBetween()` is introduced, which sets a consequent value according to the current computer time. MR works with intervals the same day, for which reason rules 1 and 6 are added in the code above. Based on

“eatSchedule” and the “eat” activity, it is possible to manipulate the rules by modelling different situations depending on what is searched for, that is “assess the situation”. For example, let’s assume an unhealthy case where the user is eating after hours, then the caregiver is alerted through “eatingAfterHours” an Outcomes state in the context “eating” and type “alertCaregiver”:

1. `SSR((#eatSchedule^eat)->eatingAfterHours);`

In order to illustrate a scenario that embraces the user and caregiver, if the user is not eating after half an hour (configured at the interface) within the defined time for doing so, the system reminds the user to eat and alerts the caregiver 1 hour later if the user has not yet eaten:

1. `SSR((-) [1800s.]eatSchedule ^
[-] [1800s.]#eat)
->unhealthyEatingWarnUser);`
2. `SSR((-) [3600s.]eatSchedule ^
[-] [3600s.]#eat)
->unhealthyEatingCarer);`

Above, the times (1800 and 3600 s) are loaded from the interface “unhealthyEatingWarnUser” a “warnUser” type and “unhealthyEatingAlerCarer” an “alertCarer” type. These two rules evaluate the situation according to the user configuration and notify the users of the situation.

This is how MR rules work in a real case. When there is no evidence of the user eating within a schedule, the system changes the state type “warnUser” to make known to the user’s device (mobile App) that the user has a warning, and the same for the caregiver’s device.

4.4.2 Sleeping activity

The process for sleep follows the same logic used to detect the “eat” activity. First the evidence that the user is sleeping in the current environment is defined. It is easy to suppose that if the pressure pad on the bed is activated (false), then the user could be sleeping and vice versa. However, in a more complex situation, suppose that the user is on the bed reading using the light (“BedroomLighOn” in line 2 below) or sitting on the bed doing some activity (e.g. folding clothes) which involves “BedroomMotion” (line 3 below). To recognize this activity a pressure pad placed on the bed, a

movement sensor and the switcher in the bedroom are used. Here, the user localization is omitted, however “userBedroom” can be added in lines 1–4 below, analogous to the state “userKitchen” in the “eat” activity.

Finally, the system infers “the user could be sleeping”, if there is no movement, the light is off and the user is on the bed (BedroomBedPressure is 1/True if no pressure is detected, and #BedroomBedPressure is 0/False if it detects pressure):

```

1. SSR((#BedroomBedPressure^#BedroomLight^
      #BedroomMotion)->sleep);
2. SSR((#BedroomBedPressure^BedroomLight^
      #BedroomMotion)->#sleep);
3. SSR((#BedroomBedPressure^#BedroomLight^
      BedroomMotion)-> #sleep);
4. SSR((BedroomBedPressure)->#sleep);
5. SSR((-) [1800s.] #sleep ^
      [-] [1800s.]sleepSchedule)
      -> sleepAlertUser);
6. SSR((-) [3600s.] #sleep ^
      [-] [3600s.]sleepSchedule)
      -> sleepAlertCarer);
7. SSR((clockBetween(00:00:00-09:00:00))
      -> sleepSchedule);
8. SSR((clockBetween(09:00:01-14:59:59))
      -> #sleepSchedule);
9. SSR((clockBetween(15:00:00-16:00:00))
      -> sleepSchedule);
10. SSR((clockBetween(16:00:01-21:59:59))
      -> #sleepSchedule);
11. SSR((clockBetween(22:00:00-23:59:00))
      -> sleepSchedule);

```

More situations can be defined, such as alerting the user when he/she sleeps late or does not get up at a reasonable time:

```
SSR((sleep^#sleepSchedule)->sleepAlertUser)
```

The system was tested for an irregular sleep pattern in which the user gets out of bed several times during the night. However, this information could be monitored by checking the “sleep” state during night on the graph, for example, whether to alert the user or caregiver could be decided by:

```

1. SSR((-) [30s.]BedroomBedPressure ^
      sleepSchedule)
      -> getOutBed);
2. SSR((-) [5s.]#BedroomBedPressure)
      ->#getOutBed);
3. SSR((<->[60s.]getOutBed ^
      <->[60s.]#getOutBed ^
      sleepSchedule)
      ->pattern1);
4. SSR((-) [60s.]pattern1 ^
      <->[60s.]getOutBed ^
      sleepSchedule)
      ->pattern2);
5. SSR((-) [60s.]pattern2 ^
      <->[60s.]getOutBed ^
      sleepSchedule)
      ->pattern3);
6. SSR((pattern3) -> irregularSleep);

```

The above rules work analogously to a counter. If the pattern “getOutBed” repeats three times during the night (coded by sleepSchedule), the system deduces an “irregularSleep” situation. If the state for this is an “alertCaregiver” type, the caregiver is alerted about the anomalous situation. This pattern is interesting to analyse due to sleep interruption being considered an important behaviour for the sleep habits of PwD as many researchers point out, such as Hanford et al. [38].

4.4.3 Wandering behaviour

Although, this behaviour implies various situations [32], this work detects when users walk around the house with various PIR sensors activated by changing locations continuously. The rules used to describe this behaviour include the PIR and BLE sensors. The approach is that, each time there is a change of room, the state “pattern” is activated (lines 9–18, below). These rules load 30 seconds from the interface (similar to Fig. 12), as the time spent by the user moving between rooms. The “pattern” state continues as active (value true or 1) while the user is moving to two or more rooms and it is reset if this action is no longer detected (lines 19–23, below). But if “pattern” stays activated for enough time, this means the user is changing rooms many times in a short period, which could be evidence of wandering (line 25, below). Consequently, the user is warned and the caregiver alerted according to the time interface (lines 26–27, below) (Fig. 12). Note, there is no schedule rule for wandering because the times are not configured in the user interface, hence the behaviour

is “unusual” at any time. The following example describes the previous situation and shows another way to express user localization using the PIR sensors and BLE together (lines 1–8 below):

```

1. SSR((KitchenMotion ^ userKitchen)
      -> isKitchen);
2. SSR((#KitchenMotion ^ #userKitchen)
      -> #isKitchen);
3. SSR((LivingroomMotion ^ userLivingroom)
      -> isLivingroom);
4. SSR((#LivingroomMotion^#userLivingroom)
      -> #isLivingroom);
5. SSR((BedroomMotion ^ userBedroom)
      -> isBedroom);
6. SSR((#BedroomMotion ^ #userBedroom)
      -> #isBedroom);
7. SSR((ToiletMotion ^ userToilet)
      -> isToilet);
8. SSR((#ToiletMotion ^ #userToilet)
      -> #isToilet);
9. SSR((<->[30s.]isLivingroom ^
      <->[30s.]isKitchen)
      -> pattern);
10. SSR((<->[30s.]isLivingroom ^
      <->[30s.]isBedroom)
      -> pattern);
11. SSR((<->[30s.]isLivingroom ^
      <->[30s.]isToilet)
      -> pattern);
12. SSR((<->[30s.]isLivingroom ^
      <->[30s.]isShower)
      -> pattern);
13. SSR((<->[30s.]isKitchen ^
      <->[30s.]isBedroom)
      -> pattern);
14. SSR((<->[30s.]isKitchen ^
      <->[30s.]isToilet)
      -> pattern);
15. SSR((<->[30s.]isKitchen ^
      <->[30s.]isShower)
      -> pattern);
16. SSR((<->[30s.]isBedroom ^
      <->[30s.]isToilet)
      -> pattern);
17. SSR((<->[30s.]isBedroom ^
      <->[30s.]isShower)
      -> pattern);
18. SSR((<->[30s.]isToilet ^
      <->[30s.]isShower)
      -> pattern);
19. SSR(([-] [30s.]#isKitchen ^
      [-] [30s.]#isBedroom ^
      [-] [30s.]#isToilet)
      ->#pattern);
20. SSR(([-] [30s.]#isLivingroom ^
      [-] [30s.]#isBedroom ^
      [-] [30s.]#isToilet)
      ->#pattern);
21. SSR(([-] [30s.]#isLivingroom ^
      [-] [30s.]#isBedroom ^
      [-] [30s.]#isToilet)
      ->#pattern);
22. SSR(([-] [30s.]#isLivingroom ^
      [-] [30s.]#isKitchen ^
      [-] [30s.]#isToilet)
      ->#pattern);
23. SSR(([-] [30s.]#isLivingroom ^
      [-] [30s.]#isKitchen ^
      [-] [30s.]#isBedroom)
      ->#pattern);
24. SSR(([-] [30s.]#pattern ^
      wandering)->#wandering);
25. SSR(([-] [30s.]pattern ^
      #wandering)->wandering);
26. SSR(([-] [1s.]wandering)
      -> wanderingAlertUser);
27. SSR(([-] [20.]wandering)
      -> wanderingAlertCaregiver);

```

4.4.4 Elopement behaviour

Since this behaviour implies leaving the house, the first step is to define what constitutes “goOut” and it is also interesting to define when the user comes back to the house “goIn”, which is important in order to reset the “goOut” state (if the user is in) and to monitor how long the user is out (lines 5 and 4, below). The state “userCorridor” is given by a BLE beacon placed in the corridor next to entrance door. Figure 9 shows the database rows affected by this action.

AnAbEL - User's Interface

Fig. 12 Interface with an example of “wandering” activity configuration

1. SSR((clockBetween(10:00:00–19:00:00)

 -> elopementSchedule);
2. SSR((#clockBetween(00:00:00–09:59:59)

 -> #elopementSchedule);
3. SSR((#clockBetween(19:00:01–23:59:59)

 -> #elopementSchedule);
4. SSR(((->[3s.]FrontdoorMotion ^

 [-][1s.]#EntranceDoor ^

 EntranceDoor ^

 userCorridor) ->goOut);
5. SSR(((->[10s.]EntranceDoor ^

 [-][1s.]#FrontdoorMotion ^

 FrontdoorMotion ^

 <->[3s.]userCorridor)

 -> goIn);
6. SSR((([-][1s.]#goOut ^ goOut ^

 #elopementSchedule)

 -> elopementAlertUser);
7. SSR((([-][30s.]#goOut ^ goOut ^

 #elopementSchedule)

 -> elopementAlertCarer);
8. SSR((goOut ^ [-][1s.]#goOut)->#goIn);
9. SSR((goIn ^ [-][1s.]#goIn)->#goOut);
10. SSR((([-][1s.]elopementAlertUser)

 -> #elopementAlertUser);
11. SSR((([-][1s.]elopementAlertCarer)

 -> #elopementAlertCarer);

5 Test and evaluation

Once all the AnAbEL components were deployed and working together, it was tested and adjusted until we achieved the best possible accuracy. Previously, each element was subjected to basic software testing processes focusing more

on reliability than, for example, performance, although that was taken into consideration. Also, as the work by Augusto et al. points out [47], there are gaps in context testing in intelligent environments. The context is outside classical development testing, yet it is a crucial element of these sorts of systems. In this research, various contexts for each ADL or behaviour and some action related to them (e.g. going out) were tested following the methodology proposed in [47]. Some AR assessment examples can be seen at the Figshare repository⁶. Although, in this paper activity rules are presented separately, they are merged into a template loaded into MReasoner using the parameters from the interface and automatically setting the schedule and alerts times. This makes AnAbEL work as one integrated tool instead of, as it could seem from the explanation above, tools working separately.

Tests showing the success of the software and AR components and validation of the user experience from PwD are still desirable.

It was not possible to validate the system with PwD in a real home environment due to the health and safety regulations of the campus. However, the system was presented to a group of users, 22 in total. Their average age was around 30 and they were all undertaking a BSc or MSc in environmental health. The aim was to collect impressions from people with an interest in and knowledge of ageing and housing. At the same time, being students, they could pose concerns about current technology. They were invited to the laboratory to receive an extensive explanation of the system and

⁶ https://mdx.figshare.com/articles/Detecting_real-time_behaviour_inside_the_house_using_non-intrusive_devices_within_people_with_dementia_context_/7406360.

datetime	abc context	abc type	abc state	123 value	abc info
2019-07-01 18:00:35	elopement	4	UserResponse1	1	I go to Doctor
2019-07-01 18:00:26	elopement	2	ElopementAlertUser	1	[NULL]
2019-07-01 18:00:25	elopement	1	goOut	1	[NULL]

Fig. 13 Elopement sequence in database. The system detects the user going out, so the system warns the user immediately. The user responds to the warning by selecting “I am going to the Doctor”. The

caregiver receives this information and decides whether the situation needs intervention

a demonstration. Afterwards, they filled in an anonymous survey related to their experience, giving positive feedback and valuable personal impressions.

Around 76% of the responders were sure that these sorts of systems could improve the lives of people with cognitive impairment in the early stages. While the rest had some concerns, nobody rejected the idea, remarking that AAL technology could be decisive in the future. However, they expressed doubts about the technology being accepted by PwD. As other studies mentioned here describe, more than half (66%) considered the system as being able to strongly enhance user autonomy by guiding them and encouraging a healthy lifestyle. These findings support the idea of further cooperation with other professionals to focus on improving this system by creating real test environments (Fig. 13).

Empowering user autonomy and supporting caregivers is the main goal of the AnAbEL system. The students considered these points important and were positive about the technology, particularly for the early stages of dementia when PwD still relish their autonomy and independence and have the capacity to make decisions and understand advice. Also, they strongly agreed that the system could be very helpful in both enhancing autonomy and improving user safety. One important concern expressed by many participants in the survey was a possible misunderstanding about the user information offered by the system, which could provoke a bad intervention or diagnostic. This issue has been pointed out in previous AAL research, where a sensor malfunction can generate undesirable outcomes. This concern should guide future work on system reliability.

Finally, they expressed doubt about the ratio of system cost to real benefits for PwD. This issue could be alleviated as the growth of the IoT brings costs down. The current cost of the equipment used is approximately £600, which represents good value for money considering the importance of the service provided.

It is important to note, that several live demonstrations in the laboratory were also attended by professionals and representatives of London Borough Cities of Finchley and Croydon. They showed great interest in the system and firm up its utility. They also contributed to refining some of the system’s requirements and services. These expert feedbacks

has been very important to overcome the limitation of testing the system with PwD.

6 Conclusion

The state-of-art in AAL shows great advances in supporting people with dementia. However, AAL systems still have limitations identifying human behaviour with precision. These issues are accentuated for PwD, whose behaviour can be even more difficult to understand and anticipate. Nonetheless, the consensus is that enhancing PwD autonomy is a great step forward, which would benefit all people involved.

The system described in this article focuses on people experiencing symptoms of early stages of dementia. The system has been developed and designed centred on their autonomy and self-esteem, including a more direct and personalized interaction. Considering the current Activity Recognition limitations in detecting activities and bearing in mind user safety, our system keeps the caregivers involved while helping to reduce the burden of their role. We present the final pilot of AnAbEL, which has been deployed and addresses shortcomings in the state of the art.

The system shows good accuracy in detecting situations and appropriately manages reminders to users and alerts to caregivers with good user localization. Also, the positive external feedback of health professionals working with dementia who participated in real demonstrations makes the approach presented an exciting point of for further research. The system shows effective performance in real life situations, as recognized by the validation of experts in dementia services working for London councils.

The AR process could be improved by adding other approaches. Using the learning process associated with the MReasoner, it is possible to define specific activities for each user. It is possible to add more ADLs, such as taking medication, drinking, or performing any activity as part of treatment. By installing complementary sensors, the number of ADLs and the detection accuracy could be improved.

Finally, a stronger User-centred Design approach involving PwD and co-design techniques could develop fitter solutions for the interface and mobile application. It could also

lead to new user interactions and parameters that increase user adaptation. All these issues offer interesting future possibilities to improve the quality of life of users in the early stages of dementia and older adults in general. Despite current concerns on users' acceptance of technology, the next generation of older adults is the currently middle-aged population who are more used to technology in their daily lives. Therefore, after the initial infrastructure has been achieved, the focus at present is to put more emphasis on creating more intelligent environments for people with dementia and similar conditions.

Compliance with Ethical Standards

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

1. Alzheimer's research uk. statistics about dementia (2018). Accessed 10-07-2019
2. Kaufmann, E.G., Engel, S.A.: Dementia and well-being: a conceptual framework based on tom kitwood's model of needs. *Dementia* **15**(4), 774–788 (2016)
3. Smebye, K.L., Kirkevold, M., Engedal, K.: Ethical dilemmas concerning autonomy when persons with dementia wish to live at home: a qualitative, hermeneutic study. *BMC Health Serv. Res.* **16**(1), 21 (2015)
4. Almvik, R., Rasmussen, K., Woods, P.: Challenging behaviour in the elderly—monitoring violent incidents. *Int. J. Geriatr. Psychiatry* **21**(4), 368–374 (2006)
5. Astrom, S., Karlsson, S., Sandvide, A., Bucht, G., Eisemann, M., Norberg, A., Saveman, B.-I.: Staff's experience of and the management of violent incidents in elderly care. *Scand. J. Caring Sci.* **18**(4), 410–416 (2004)
6. Morgan, D.G., Crossley, M.F., Stewart, N.J., D'Arcy, C., Forbes, D.A., Normand, S.A., Cammer, A.L.: Taking the hit: Focusing on caregiver "error" masks organizational-level risk factors for nursing aide assault. *Qual. Health Res.* **18**(3), 334–346 (2008)
7. Lazarou, I., Karakostas, A., Stavropoulos, T.G., Tsompanidis, T., Meditskos, G., Kompatsiaris, I., Tsolaki, M.: A novel and intelligent home monitoring system for care support of elders with cognitive impairment. *J. Alzheimers Dis.* **54**(4), 1561–1591 (2016)
8. Fernández-Llatas, C., García-Gomez, J.M., Vicente, J., Naranjo, J.C., Robles, M., Benedi, J.M., Traver, V.: Behaviour patterns detection for persuasive design in nursing homes to help dementia patients. In: 2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 6413–6417 (2011)
9. Stucki, R.A., Urwyler, P., Rampa, L., Muri, R., Mosimann, U.P., Nef, T.: A web-based non-intrusive ambient system to measure and classify activities of daily living. *J. Med. Int. Res.* **16**, 175 (2014)
10. Li, H., Wan, C.-y., Shah, R.C., Sample, A., Patel, S.N.: Idact: Towards unobtrusive recognition of user presence and daily activities, pp. 1–8, 04 (2019)
11. Hoey, J., Ploetz, T., Jackson, D., Monk, A., Pham, C., Olivier, P.: Rapid specification and automated generation of prompting systems to assist people with dementia. *Pervasive and Mobile Computing*, vol. 7, pp. 299–318, 06 (2011)
12. Lara, O.D., Labrador, M.A.: A survey on human activity recognition using wearable sensors. *IEEE Communications Surveys Tutorials*, vol. 15, pp. 1192–1209 (2013)
13. Pirsivash, H., Ramanan, D.: Detecting activities of daily living in first-person camera views. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2847–2854 (2012)
14. Kostavelis, I., Vasileiadis, M., Skartados, E., Kargakos, A., Giakoumis, D., Bouganis, C.-S., Tzovaras, D.: Understanding of human behavior with a robotic agent through daily activity analysis. *Int. J. Soc. Robot.* **11**, 437–462 (2019)
15. Cook, D., Feuz, K.D., Krishnan, N.C.: Transfer learning for activity recognition: a survey. *Knowl. Inf. Syst.* **36**, 537–556 (2013)
16. Nef, T., Urwyler, P., Büchler, M., Tarnanas, I., Stucki, R., Cazzoli, D., Muri, R., Mosimann, U.: Evaluation of three state-of-the-art classifiers for recognition of activities of daily living from smart home ambient data. *Sensors* **15**(5), 11725–11740 (2015)
17. Fernando, C., Crispim-Junior, C., Bremond, F., Joumier, V.: A multi-sensor approach for activity recognition in older patients, 09 (2012)
18. Zhou, Y., Law, C.L., Guan, Y.L., Chin, F.: Indoor elliptical localization based on asynchronous uwb range measurement. *IEEE Trans. Instrum. Meas.* **60**, 248–257 (2011)
19. Filippoupolitis, A., Oliff, W., Takand, B., Loukas, G.: Loction-enhanced activity recognition in indoor environments using off the shelf smart watch technology and ble beacons. *Sensors* **17**, 6 (2017)
20. Gupta, P., Dallas, T.: Feature selection and activity recognition system using a single triaxial accelerometer. *IEEE Trans. Biomed. Eng.* **61**, 1780–1786 (2014)
21. Ozdemir, A.T., Barshan, B.: Detecting falls with wearable sensors using machine learning techniques. *Sensors* **14**(6), 10691–10708 (2014)
22. Paiva, S., Abreu, C.: Low cost gps tracking for the elderly and alzheimer patients. *Procedia Technology*, vol. 5, pp. 793–802, 2012. In 4th Conference of ENTERprise Information Systems—Aligning Technology, organizations and people (CENTERIS 2012)
23. Pollack, M.E., Brown, L., Colbry, D., McCarthy, C.E., Orosz, C., Peintner, B., Ramakrishnan, S., Tsamardinos, I.: Autominder: an intelligent cognitive orthotic system for people with memory impairment. *Robot. Autonom. Syst.* **44**, 273–282, 03 (2003)
24. McNaull, J., Augusto, J.C., Mulvenna, M., McCullagh, P.: Flexible context aware interface for ambient assisted living. *Human-centric Comput. Inf. Sci.* **03**, 2192–2962 (2014)
25. Yu, Z., Liang, Y., Guo, B., Zhou, X., Ni, H.: Facilitating medication adherence in elderly care using ubiquitous sensors and mobile social networks. *Comput. Commun.* **65**, 1–9 (2015). *Mobile Ubiquitous Sensing from Social Network Viewpoint*
26. Alzheimers.net. technological innovations for those with dementia (2018). Accessed 10-07-2019
27. Wang, J., Du, Y., Coleman, D., Peck, M., Myneni, S., Kang, H., Gong, Y.: Mobile and connected health technology needs for older adults aging in place: Cross-sectional survey study. *JMIR Aging* **2**, e13864 (2019)
28. Augusto, J., Kramer, D., Alegre, U., Covaci, A., Santokhee, A.: The user-centred intelligent environments development process as a guide to co-create smart technology for people with special needs. *Univ. Access Inf. Soc.* **17**, 115–130 (2018)
29. Acampora, G., Cook, D.J., Rashidi, P., Vasilakos, A.V.: A survey on ambient intelligence in healthcare. *Proc. IEEE* **101**, 2470–2494 (2013)
30. Alzheimer's association report 2015: Alzheimer's disease facts and figures. *Alzheimer's & Dementia*, 11(3): 332–384, (2015)

31. Guesgen, H.W., Marsland, S.: Spatio-Temporal Reasoning and Context Awareness, pp. 609–634. Boston, MA: Springer US (2010)
32. Lai, C.K., Arthur, D.G.: Wandering behaviour in people with dementia. *J. Adv. Nurs.* **44**(2), 173–182 (2003)
33. Steinhauer, H., Chua, S.-L., Guesgen, H., Marsland, S.: Utilising temporal information in behaviour recognition. 01 (2010)
34. Alberdi, A., Aztiria, A., Basarab, A.: On the early diagnosis of alzheimer's disease from multimodal signals: A survey. *Artif. Intell. Med.* **71**, 1–29 (2016)
35. Braley, R., Fritz, R., Van Son, C.R., Schmitter-Edgecombe, M.: Prompting Technology and Persons With Dementia: The Significance of Context and Communication. *The Gerontologist* **59**, 101–111, 06 (2018)
36. Boyd, H.C., Evans, N.M., Orpwood, R.D., Harris, N.D.: Using simple technology to prompt multistep tasks in the home for people with dementia: An exploratory study comparing prompting formats. *Dementia* **16**(4), 424–442 (2017)
37. Lamparero, J.: The use of music in dementia care : A literature study. Master's thesis, Novia UAS. Vaasa, Finland (2015)
38. Hanford, N., Figueiro, M.: Light therapy and alzheimer's disease and related dementia: past, present, and future. *J. Alzheimers Dis.* **33**, 913–922, 06 (2018)
39. Augusto, J.C., Quinde, M.J., Gimenez, J.G., Oguego, C.L.: The search smart environments architecture. In: 15th International Conference on Intelligent Environments (2019)
40. Ibarra, U.A., Augusto, J.C., Goenaga, A.A.: Temporal reasoning for intuitive specification of context-awareness. In: 2014 International Conference on Intelligent Environments, pp. 234–241 (2014)
41. Torres-Sospedra, J., Fernandez, O., Belmonte, Silva, G. Mendoza, Montoliu, R., Puertas-Cabedo, A., Rodríguez-Pupo, L. E., Oliver, S. Trilles, Calia, A., Benedito-Bordonau, M., Huerta, J.: Lessons Learned in Generating Ground Truth for Indoor Positioning Systems Based on Wi-Fi Fingerprinting, pp. 45–67. 01 (2019)
42. Sora, D., Augusto, J.: Managing multi-user smart environments through ble based system. In: Intelligent Environments 2018-Workshop Proceedings of the 14th International Conference on Intelligent Environments, pp. 234–243 (2018)
43. Faisal-Mohamed-Yusof, M., Romli, N., Yusof, M.M.: Design for elderly friendly: mobile phone application and design that suitable for elderly. *Int. J. Comput. Appl.* **95**, 28–31 (2014)
44. Orpwood, R., Gibbs, C., Adlam, T., Faulkner, R., Meegahawatte, D.: The design of smart homes for people with dementia—user-interface aspects. *Univ. Access Inf. Soc.* **4**, 156–164 (2005)
45. Tran, A.C., Marsland, S., Dietrich, J., Guesgen, H.W., Lyons, P.: Use cases for abnormal behaviour detection in smart homes. In: Y. Lee, Z. Z. Bien, M. Mokhtari, J. T. Kim, M. Park, J. Kim, H. Lee, I. Khalil, (Eds.), *Aging Friendly Technology for Health and Independence*. pp. 144–151, Springer, Berlin (2010)
46. Ali, S.M.M., Augusto, J.C., Windridge, D.: Improving the adaptation process for a new smart home user. In: *Artificial Intelligence XXXVI* (M. Bramer and M. Petridis, eds.), (Cham), pp. 421–434, Springer International Publishing (2019)
47. Augusto, J., Quinde, M., Oguego, C.: Context-aware systems testing and validation. In: 10th International Conference Dependable Systems, Services and Technologies (2019)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.