An Intelligent Multi-Agent Memory Assistant

Ângelo Costa
Paulo Novais

Universidade do Minho, Departamento de Informática

acosta@di.uminho.pt

pjon@di.uminho.pt

Abstract   World population is ageing and increasingly scarce resources are required to cover the needs of everyone adequately. Medical conditions, especially memory problems, restrict the daily life of a broad slice of the elderly population, affect their independence. To prevent this, providing the right care and assistance while having in mind the costs implicated is essential. One possible path is to work with resources that we already have today and create innovative solutions to achieve the required level of support. There are not many solution either technological or not to prevent memory loss. In this work we present a possible solution aimed at restoring or maintaining the independence of elderly people, through the use of so-called Memory Assistants. We thus present an Intelligent Multi-Agent Memory Assistant designed to help people with memory problems remember their events and activities. The implementation of an event manager, free time manger, medication remainder and a sensory system, to manage and monitor the user, we aim to improve their quality of life and increase their independence.

1   Introduction

"How will our life be when we get old?"

We pose this question in our daily life almost every day. The answer is: there is no answer. In fact, each technological evolution adds another variable to our possible future. At most we, as Humans, envision a future that is not too broad or too restricted. The technology that surrounds us changes our life for better or for worse. In fact, if we seek back 20 years, the computer field was complex and restricted to trained engineers and technicians. Nowadays, even a child can efficiently operate a computer. Thus, it is not wrong to say that our life is shaped by technological evolution. Going along this flow, the elderly community is now prone to accept new technologies and is in fact prepared to work with new devices.

1.1 Motivation

The last census made by the United Nations revealed that most of the European developed countries show a decreasing rate of new-borns and, at the same time, are the ones with a greater percentage of elderly persons, as can be seen in Fig. 1. This phenomenon is also seen in several other developed countries around the globe. The main causes that can be pointed out are the quality of current health-care services and better and healthier life styles. At the same time, these countries also have now a culture oriented to have fewer children, with these children having a better quality of life [1].

The provision of social services, especially health care, to a growing percentage of elderly is not attainable in the future with the structure of current society. It is our main objective to provide an answer to typical problems that arise in people when they are in an elderly stage. The social implications imply a response to the problem.

In our perspective, the answer must be a sustainable solution, economic and socially, supported by technology – electronic devices and computer applications. We propose a solution based on devices fairly cheap that can provide several features in a small and discrete manner. The adoption of a technological solution means that the persons that will benefit from it, the elderly ones, can have access to it with small costs involved, and because it is a combination of software and hardware, upgrades and new features are easier to implement.
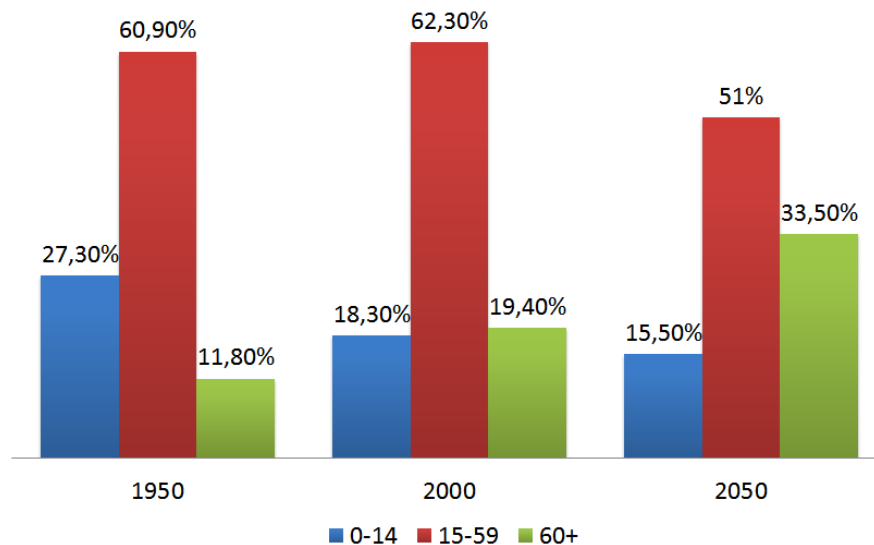

Fig. 1. Population growth (expected).

1.2 Aging Factors

With age, memory problems tend to arise. Recent studies show that we tend to start having memory losses at around 50 years of age [2] [3], with a trend to worsen as age increases. In the path to total memory loss, several stages can be identified. The stages are well known in the psychology and medical areas and are depicted in Fig. 2 [4]. To determine the current stage of a particular patient, practitioners generally rely on tests performed to that patient. The results of the tests are then compared to the ones of a regular person, and a diagnosis is achieved. These tests consist in routines, questions and activities related to the user, organized in two common groups: comprehensive tests and brainwave analysis. There are three possible outcomes: cognitively impaired (normal daily experiences), mild cognitive impairment (daily life experiences mildly affected) and severe cognitive impairment (needs constant supervision).

MCI, or Memory Cognitive Impairment, is the measure of the memory loss steps. The bigger the index, the worse the condition of the patient. Figure 3 depicts the evolution of the disease in the United States [5]. In this figure, it can be seen that there is an increase of 200% in 10 years only.
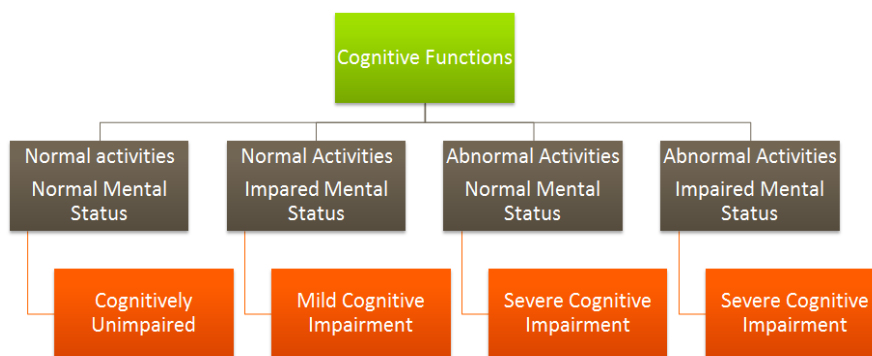
Fig. 2. Memory degeneration steps [4].

Currently, modern societies are not prepared to give support to the increased elderly population, although some have cultural behaviours that advocate the care of the elderly. Nevertheless, at this moment, virtually every country urgently needs innovative solutions and approaches to accommodate and take care of the elderly.

1.3 Possible Paths

Given the urgency of the problem, there is a need for approaches that can be tested and implemented presently and available to full use in a near future. The services

available nowadays to help elderly in their daily tasks are very scarce and are not nearly as focused on the actual problems as they should.

The development of technological solutions that make an integrated use of these already existing services is the best approach. In fact, software and hardware solutions are the best way to easily improve and implement tools to assist the users essentially because they are relatively cheap, easy to upgrade, fully configurable and adaptable, and allow the implementation of new functionalities.
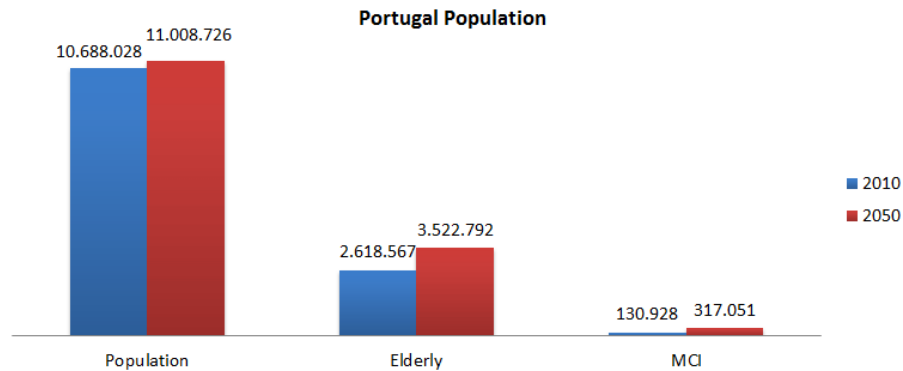

Fig. 3. Graph of the expected evolution of MCI in the Portuguese population

In terms of already developed solutions to the memory problems, the medical community essentially suggests [6] the use of games and tests to enhance the patient's brain activity and the implementation of activities that break everyday routines. As an example, tests and playful activities are now used as part of the treatment to Alzheimer's disease. This disease still has no known cure and the best that can be done is to prevent or slow down the loss of the remaining mental abilities [7].

Another important fact, revealed by a study carried on the University of Harvard [6] is that the part of the brain that we use to store the long term memories is also responsible for the creation of new thoughts and ideas, an ability commonly known as *envisioning*. Consequently, the loss of saved memories has as one of its consequences a decrease in the ability to project new activities. Another problem is the loss of a simple and very common skill: *the cause and consequence*. This ability is what tells us that doing a given task will probably have a given outcome, by having done a similar task in the past or by simple logic. An example can be: when opening a faucet, water will pour through that faucet. This is a simple example but it causes impact in our everyday life. Without this skill, whenever a new complex task is learned, for example by viewing another person executing it, in the next day the patient will not be able to infer the consequences of its execution, if requested to.

1.4 VirtualECare Project

1.4.1   AAL - Ambient Assisted Living

Ambient Assisted Living (AAL) is a relatively new paradigm that emerged in the 80's. It is a junction of domotics with intelligent and proactive software applications, aimed at implementing intelligent systems inside common domestic environments, in order to assist the patient and control the environment and the devices around him [7].

AAL is currently an active topic of research, with new projects arising to complete previous ones or to present new functionalities that were not developed before. Actually, the AAL paradigm can take several forms, shaping itself according to the devices at the user's home or even attached to/in the user's body, to fully fledge top-level projects.

1.4.2   VirtualECare Objectives

The VirtualECare project (VEC) focuses on a high-level view of the Ambient Assisted Living field. It assures the complete panoply of services to aid and assist elderly persons or people in need. It spans all the layers ranging from the user and the sensors in his home environment, to high level Group Decision Support Systems. It aims at a complete solution to a fairly recent and under-explored problem: the provision of health-care services directly at the home of the elderly [8].

VEC is also acting as an enabler for complementary smaller-level projects, being developed on the basis of service-sharing and functionality complementarity. One of them deals with the implementation of sensors in an environment, aimed at monitoring its users and acquiring important information, in order to create and refine an individual profile for each one of the known users. The project started with an initial implementation of a simulation platform that could either realistically simulates sensors or actually be connected to real ones. Having received a positive feedback from the scientific community, the project is under on-going development to implement several other interesting features. Specifically, researchers are now dealing with the determination of the user's mood and with the challenge of connecting this platform with other higher level projects, such as the VEC (Figure 4).

In terms of core functionalities, the data collected is used to adjust the actions and decisions implemented in the user typical profile in order to comply more accurately with the user needs. The profiles are very important as they are useful to create an initial picture of the user's typical preferences. In fact, personalization is an important feature for two main reasons: it allows achieving better resources assignment and higher levels of user comfort.

In terms of resources assignment, the system should be able to, through the use of a personalization map, infer what feature is most important and what actions are best to execute, in order to respond to specific problems. If the user likes a room colder than the usually programmed, the system should take into account that information since the first day of implementation. This is also useful in scenarios in which there is a conflict of user's definitions, which happens when two or more users are in the same room and have different needs or preferences, with the system having to reach a compromise between both of the user definitions.
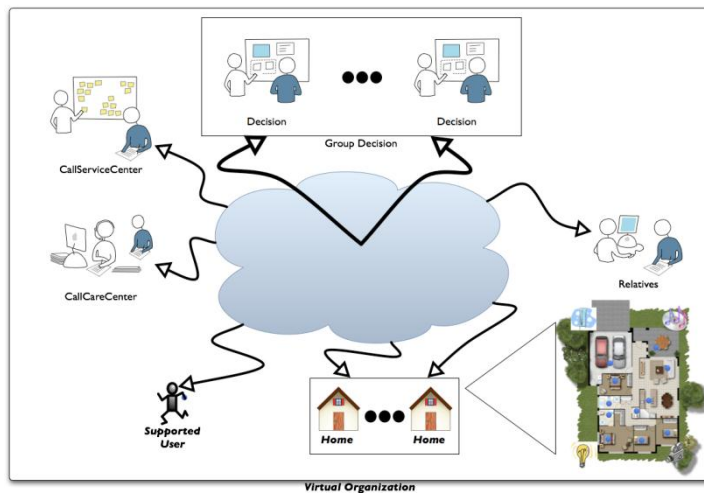

Fig. 4. VirtualECare Areas of Interest.

In terms of user's comfort, the system should be able to take decisions not in a generic way but in a rather focused one. The ability of taking actions that were carefully adjusted to the user is important so that the user does not have to correct the system periodically. In this sense, it is also important that the system has the mechanisms to determine if the user has special conditions (e.g. physical impairments) and adjust its actions accordingly.

The memory assistant area is very recent and some projects arise with the objective to solve inherent social problems that are clearly identified [7]. In fact most of the current developments have many flaws. Whilst the social problems are clearly identified, the solutions aren't, and this means that the solutions that are available, although in the core functions give an answer to the detected problems, are not well accepted by the community that will be using them. The *form following function* of the projects available nowadays have to be heavily reconsidered. For instance, most of the projects available to an elderly community do not have visual interfaces that cope with usual imparities that are common at an older age. Other common situation is that applications and projects are made by younger people, and it is in our opinion that they do not have the real perception of how the

world is viewed and operated by an elderly person. In the memory assistants' area these problems are heavily amplified, and being such a broad area, the projects are scarce and spread apart. As being a part of the AAL, the memory assistants have a large area of operation, this means that there are benefits as well problems, such as providing different solutions to different questions, but not having a streamlined solution and interpretation of each question means that every project can provide a solution unique from each other, and any of them cannot respond correctly to the user demands.

This document is organized as follows: Section 2 describes the Memory assistants' area and other related projects. Section 3 the technical overview. Section 4 presents the architecture and the several modules of this project. In section 5 the application scenarios and results of the tests, and finally, section 6 shows the conclusions and future work.

2 Memory Assistants

The memory assistants' area is a fairly new one, thus having innovative solutions to new problems. This also means that the projects available are very recent and have very different focus on the direction the solutions should be really heading. A presentation of different projects that are populating the memory assistants' area follows, exposing their features and flaws.

2.1.1 HERMES

HERMES [9] is a project developed together by CURE - Center for Usability Research and Engineering, INGEMA Foundation, IBM Haifa Research Lab, University of Bradford, Athens Information Technology and TXT e-Solutions. It is supported by the EU under the Framework Programme 7 and aims at providing an independent living and social participation to elderly persons, by improving the quality of life. The main objective is to develop software and hardware tools that will reduce the negative impact of declining cognitive capabilities, particularly the memory-related ones. An important feature of this project is the implementation of a monitoring system which should be able to record every action and associated choices of one user. This will later be used to build a map of associations of the different events and, based on that map, create a pattern that emulates the human memory mechanisms.

Despite its ambitious goals, HERMES is still in a very early stage of development. Thus, the developed tools are still very simple, with low effectiveness and an amount of problems that wave no known time of resolution. For instance, the speech recording tool must be executed in certain limited parameters in order for the speech to be converted to text and the system still needs to be manually trained for each user.

2.1.2 SenseCam

SenseCam [10] is basically a wearable camera that passively records photos, without the user's intervention. The objective is to report, through images, the user's daily activities and the places he has been to. The hardware contains even some degree of automation, e.g. light-intensity and light colour sensors, body heat detector and motion sensors. It was designed to be used as a pendant on a necklace but it may also be clipped in pockets or belts, or attached in clothing. The key idea is that the user, at the end of the day or on a posterior day can remember his past activities by watching the pictures taken while doing them.

Although the Microsoft Research Center claims it is a memory assistant for elderly, it has in fact no significant on-the-fly abilities for remembering its users of events and/or tasks. It is merely a so-called time machine, like a video camera used to record what the user has done. However, it serves the purpose of remembering the user what he/she has done, in a specific time or day. It has not any kind of associated intelligence and it is merely useful for users with total memory loss, like Alzheimer's.

2.1.3 M4L: Memories for Life

Developed by the Engineering and Physical Sciences Research Council (EPSRC), United Kingdom, M4L has as objective to use technological solutions to assist the user's memory [11]. This project focuses on five different fields: health, private life, education, entertainment and science.

Currently it is proposed a raw data archive centre with a significant capacity. This way, it can store a wide range of information about every user. The access to the data can be performed by handheld devices or computers that have a constant connection to the server. The healthcare field is currently designed to store all clinical information about the user. The private life field consists in saving all the items that are dear to the user, like his photos or financial data for example. The education, entertainment and science fields have as aim to save all the data available so that it can later be processed and selected by the user. The end result is an attempt to eliminate all the paper used nowadays and, at the same time, interconnect all the people involved in a seamless way. As it is currently used as an undifferentiated memory bank, the objective is to make a collection of well selected memories of the user.

As the project is still in a research and implementation phase, the current results are somewhat disappointing. Basically, what is holding this project back is the large amount of storage space needed for maintaining all this information. Thus, significant and practical breakthroughs are only expected in the long term, as the price of storage decreases.

The projects presented are valid in essence but miss some points. From a high level point of view, it may be said that the main problems are these: in the case of the

SenseCam the problem is interactivity; in the case of Hermes is the ability to map all the necessary connections and the resources available; and in the case of M4L is the storage and retrieval of major information.

With very different scopes the presented projects stand apart from each other, namely the HERMES project from the other ones. The HERMES project was a solid and futuristic point of view of the possible conceptions that should be available to the user. Also the complexity is at a completely different level, involving computer science and medical science, namely the neuroscience. The same cannot be applied to the other presented projects that are more simplistic but their goals are easily achievable.

3   Technology Overview

This section presents an overview of the technologies used in this project. The following descriptions underline their use, how they shape the way the project is constructed and it´s behaviour, and the benefits from its usage.

Multi-Agent Systems

From our point of view, the best paradigm that can fulfil the development requisites of Memory Assistants is the Multi-Agent System one [12]. The use of MAS in the development of an application platform enables modularity, expansibility, standardization, platform independence and the implementation of intelligent behaviours (e.g. negotiation, argumentation). Generally, agent platforms support the interaction between the agents by means of communication protocols and standards, facilitating the development process and laying the path for cross-platform compatibility. Intelligent agents can work in collaboration or individually, depending on their goals.

Agents can also be completely decentralized and are typically independent from each other. Given their flexibility, agents are easy to restructure and modify without having to change the entire platform or even having to change the communication methods. Well-developed communication ontology at the beginning of the project allows the system to use it for several functions, even if the agents are changed along the way. The implementation will follow the Wooldridge & Jennings definition of strong agent [13], an agent that is generally referred to as a computer system that is either conceptualized or implemented using concepts that are more usually applied to humans.

The fault-tolerance provided by the MAS is also crucial. If an agent has problems, the platform can continue to operate, even if at that time the services provided by that agent are not available. This can be solved by launching a new agent to take over the operations of the one that failed, given that the state of the agents is maintained outside the scope of the agents.

Another important characteristic for the development of a software platform is the possibility to develop the agents in offline mode, i.e., agents can be constructed in a replicated platform that operates in an offline mode. If they pass the stability tests, they can be launched without any loss in the main platform. The use of a MAS is one of the main characteristics of this project as it shapes its architecture. All the functionalities are provided by agents, thus the system inherits all the characteristics mentioned above.

JADE

The JADE Framework is an elegant and simple way to develop a MAS. Working over the JAVA platform means that it benefits from the use of object-oriented programming, has a large developing community and can be ported to virtually any hardware. The JADE Framework is already well-known to the Intelligent Systems and Artificial Intelligence fields, with significant projects relying on its use [14].

Communications

The communication layer is also supported by the JADE Framework. It follows a strict ontology, with the content expressed in XML language. The use of XML guarantees both human and machine readability and the correct tagging and field processing. The content is encrypted and is transferred in messages that rely on the HTTP protocol.

The proposed platform is a distributed one, so it is essential to provide the necessary mechanisms for cooperation and coordination in order to allow the agents to proceed correctly with their tasks. In that sense, the communication between the several existing agents can be established over several protocols, namely Ethernet, WIFI, GSM and UMTS. Moreover, the communication protocol complies with FIPA's (Foundation for Intelligent Physical Agents) Agent Communication Language (ACL).

Prolog

The Prolog computing language is a logic inferential one. This type of programming guarantees the mathematical treatment needed in the decision processes implemented. With the use of Prolog, which is a mathematical logical language, there is the absolute certainty of the outcome of the functions. As it is rooted in formal language it takes advantage in being correct, fast and easy to implement, thus making the decision easier to implement and fault proof.

Android

With the adoption of JAVA/JADE, the development and implementation of the tools and interfaces in mobile devices, namely the Android-based ones, is easy and seamless. The implementation of distributed memory assistants in mobile devices is an advantage as users gain mobility and can make use of the functionalities provided independently of the location. An advantage of the Android platform over the others available mobile platforms (iOS, Windows Mobile, Blackberry OS, etc.) is that is open-source and it is free and easy to develop new applications.

4   iGenda - A Multi-agent Memory Assistant

The depicted social problems, the fact that there are unfocused resources and the assumption that current projects are not providing quite the expected solution or having no actual implementation, constituted the motivation for the start of the iGenda project. This was one of the projects that emerged out of the VEC, and that was interesting enough to be decoupled and developed stand-alone, although in cooperation.

This project intends to be a response to a gap in the still undeveloped memory assistant field. Being this a relatively new field, several application scenarios can be envisioned. One of them is its use in a hospital environment to improve the physician routines and clinical duties, also connecting the patients, whether they are in the hospital or in their homes [15]. Another possible scenario is its use of active body sensors to register the vital signs in order to monitor the user's health condition and schedule, in both user's and physician's agendas, an appointment and an event. Ultimately, the system should be sufficiently autonomous to request an urgent medical response. In fact, the modular nature of the iGenda architecture allows using the same base architecture and extending it to such application scenarios, through service and functionalities reuse [16]. And, an important point is that this can be done without interfering with the functionalities that are already implemented.

As the projects presented in this field are scarce, this is our effort to accomplish the objectives proposed and fill this gap. Our objective is to apply the above presented memory assistant under a new vision, the intelligent agendas [17].

Moreover, we are also concerned with the patients and their relatives as the social support is of utter importance, mainly in the cases of elderly living alone. Therefore, the iGenda allows patient's relatives or friends to register as users of the system, to be able to access information about the patient. For example, a relative of the user can login to see when the next appointment is scheduled. The sharing of events is also supported among the group of users of a patient. In that sense, it is possible to create shared calendars and activities that are public to users of the group, incorporating this information into the agendas already being used by the users seamlessly.

In this section we highlight its architecture and main modules.

4.1 Architecture

The architecture of iGenda is based on the agent paradigm [18]. Every core action has an individual agent to support the activity and the decision, as it can be seen in Fig. 5. This way, it is safer and more immune to possible flaws. And, should they appear, it is easier to track them down and correct the problem.
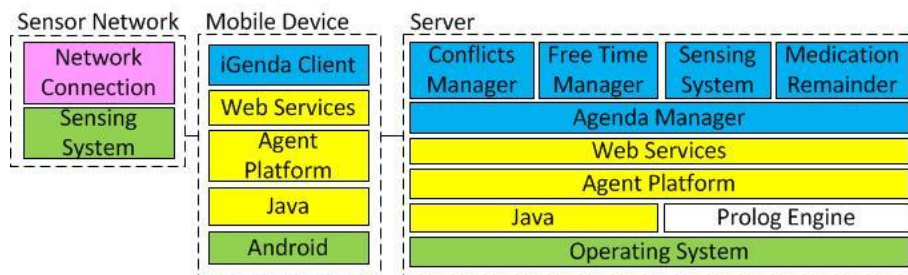

Fig. 5. The iGenda architecture.

The architecture presented on Figure 5 represents a multi-platform project. The server will create most of the decisions, the system being used to provide a mobile interface to the user and serve for the connection to the sensor system. In the figure are depicted the different operating systems, interpreters and agents. The agents assure the decision and establish the communications between the remaining agents and platforms.

All the decisions are independent logic implementations. These logic implementations are done in Constraint Logic Programming, which means that there is a close proximity with the logic rules. This approach translates in the types of predicates being taken into account as affirmations of truth. The affirmations are then tested accordingly to the algorithms implemented: if they fit, the test is successful; otherwise, the predicates are negated. This approach translates into a broad acceptance of predicates and variables, not being restricted to one input only. In lay terms, the algorithms are not a cadence of "ifs", but are "if-not", the inversion means that more tests can be made, not stressing the system by using a large amount of tests.

At the beginning of the project this approach was not implemented and, very often, tests failed because if the result was not as expected, the logic engines could not process them, thus narrowing the spectrum of cases that could be processed.
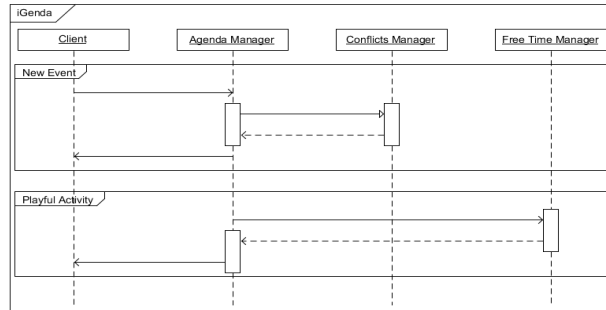
Fig. 6. Communication workflow.

The agents may communicate over different protocols (e.g. Ethernet, WIFI, GSM, and UMTS, among others). The communication protocol complies with the FIPA-ACL standard, using JADE as the agent platform and the protocol XMLCodec for coding the content (Fig. 6). All the agents are compliant with the ACL standard. Moreover, agents were also developed in JADE-Leap so that they can be implemented on mobile devices with limited resources. The support of the Health Level Seven International standard (HL7) is also a feature to be implemented, as the direct connection to hospitals is in study. This well established language will provide bidirectional communications between the institutions and the iGenda.
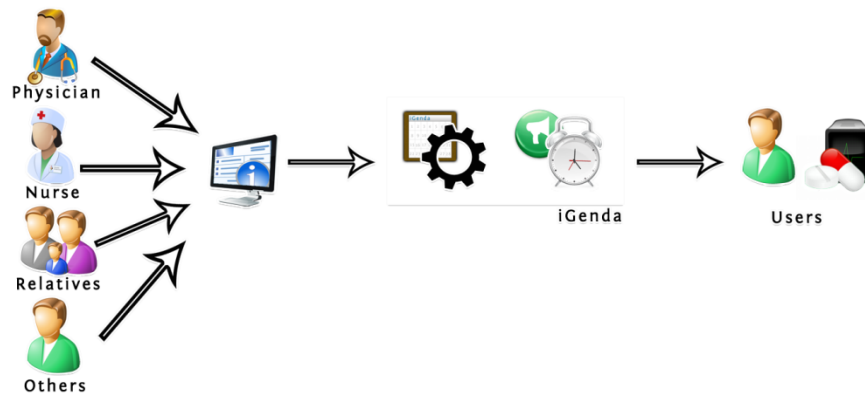

Fig. 7. A high level view of iGenda.

Initially, the project had only a little part dedicated to mobile devices. The approach was to have a mobile terminal that communicated with the user to notify him of the most important tasks. Now, as the project evolves, it is clear that a great deal of the functionalities must somehow be supported by mobile devices. This is enabled by seamless and location-independent connectivity to networks and enhanced computational features, such as GPS and sensors availability. Moreover, as

it can be seen in the description of the remaining modules (Figure 7), the mobile device is currently the best alternative to make the bridge between the users and the actual application: it is always present, it is lightweight, and it is always connected. This also represents a shift from a centric architecture to a distributed one, in which the emphasis is placed on the user nodes, making the application closer to the personal context of its users.

4.1.1 Core Modules

The iGenda is not a static system common to every user as each user has its own personalized iGenda instance, as seen in Figure 8. In the development of this project, a centralized management encompassing all users was not considered as it brings problems concerning scaling, performance and stability. The independent processing of each user is more reliable and, if an instance of iGenda crashes, it only crashes one user platform while the remaining keeps running. This approach also makes it easier to process the incoming connections and to manage the authorized users that can access the main user calendar. Let us now depict the core modules that make up the iGenda architecture.
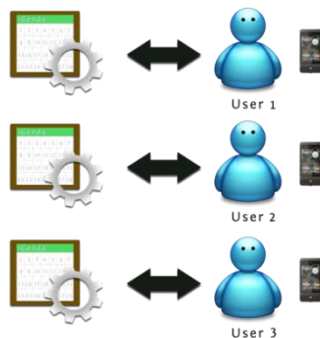

Fig. 8. One iGenda for each User

Agenda Manager

The Agenda Manager (AM) connects and controls the conflicts manager system and the scheduling system, using the communication infrastructure available to receive and send requests. As a result, the AM is responsible for the communication and security of the whole platform.

In that sense, the AM is the communication gateway. It can assume either a passive execution mode or a dominant one, depending on its configuration or demands. The passive mode constitutes the standard execution mode. In this mode, messages can be received and processed, and respond properly to the involving

agents. In the dominant mode, the AM controls the other user's iGenda sub-platforms. This agent is configured directly on the agenda and sends demanding messages to the iGenda's users (Fig. 8). The AM also ensures that the user's friends and relatives are in touch and know what activities he is or will be doing in the near future.

Conflicts Manager

The Conflicts Manager (CM) agent has as main objective to ensure that there is no overlap of activities and events. This module schedules or reorganizes each event ensuring that it is in accordance with the remaining ones. The CM is essentially a sleeping agent that only activates when the AM sends him a message. His task is to receive the incoming messages and process them.

The agent is divided into two main parts: the JAVA component receives the messages containing a potential new appointment and respective content to process while the logic engine decides how that appointment should be scheduled, taking into consideration factors like other appointments and respective priorities. When a new appointment arrives, several tests are done by the logic engine in order to determine the position in the agenda in which it should be placed. At this moment, 16 different tests are performed to see if the events fit. For instance the simplest test available is the one that verifies if an event fits in an empty space, or if it collides with existing events.

Each event received comes with an associated priority value. This value is attributed either by the user that is scheduling it or by the system. In the cases in which it is the user setting up a new high priority event, his role is verified to determine if he has the privileges to schedule that type of events. The priority values are numbered from 1 to 50, although it is not expected to the values to pass from 10. In this scale, a value of 1 denotes health-related events and is reserved to be used by physicians and medical staff only.

Given the complexity of dealing with temporal values, mainly due to the different units that compose a date value, the system uses a value in milliseconds instead of a traditional representation of a date. This makes time-related calculations much easier and gives a precision of milliseconds, rather than the precision of seconds of traditional date representation.

This process can be organized into three major steps: the initialization, the testing and the conclusion.

- Initialization: the CM JAVA component sends the calendar and the new event to the Prolog engine. This information contains all the activities to be executed in a period of time and additional information such as the priority of each activity or the persons involved.
- Testing: the Logic engine builds an internal interpretation of the calendar and tests the new event against the already scheduled ones. In the end, the outcome

can be a new calendar if the process ends successfully or an error message otherwise.

• Conclusion: the Logic engine builds a new calendar and sends it back to the JAVA component, flagging it as complete, if the calendar was built successfully. Otherwise, it sends an error message that the JAVA component can interpret and notify the involved persons.

Free Time Manager

The main functionality of the Free Time Manager (FTM) is to schedule playful activities in the free time of the user. In fact, these activities are paramount for a healthy active aging as they promote culture, education and conviviality, based on an individualized plan.

This agent can be fully configured to work in a way that maximizes each user's satisfaction. The FTM is activated by a trigger sent by the AM; this trigger is activated when its configured time is achieved. The process is relatively simple: the logic engine selects a possible action from a database using an algorithm that normalizes the choices and introduces a level of randomness to ensure that it does not always do the same choice.
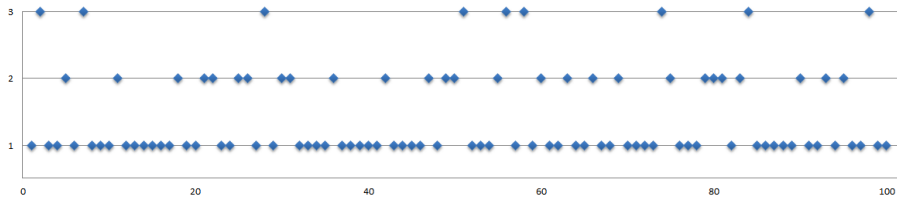

Fig. 9. Results of 100 tests and the chosen event.

$$prio = \{m_{prio} \ldots n_{prio}\}$$

$$a = \sum_{i=m_{prio}}^{n_{prio}} i$$

$$b = \sum_{i=1}^{m_{prio}} i$$

$$f(prio) = -1 * Rand(a) * b$$

Eq. 1: Free Time Manager choosing function

In Figure 9, the dots represent the activity selected, out of a group of 3 possibilities, as it can be seen, the activity 1 has the most hits, but the other activities are also selected along the way.

As mentioned before, each event has a priority value, which is also considered by the algorithm, represented in Eq. 1. The events selected are the ones that can fit the free space in the calendar. The *prio* series contain the events and the correspondent priority values; *a* is the sum of the priority all the possible events; *b* is the sum of all the possible events; and *Rand* is the function that results a random number in the *a* range.

Additional characteristics like overhead time (e.g. time to travel from current location of appointment location) are also described in the new activity so that the logic agent can accurately determine the time it takes to prepare an activity. The Logic module handles a large database of available activities and it does a selection of available events that can fit the available space. The selection algorithm picks one activity, and that event is saved in the calendar. This task is done for each free space, so multiple iterations are done in each execution.

4.1.2 Interface

In terms of the user interface, a basic one has been initially developed, merely to test the functionalities and the agent's interactions. This first interface is now going under major restructuring and was enriched with new visual and new functionalities, encompassing SMS and Voice Mail services. These two new functionalities were developed mainly thinking of the elderly people who are not comfortable with the use of more recent technological developments, so that technology does not become a barrier. The deprecated user interface is shown in Fig. 10.
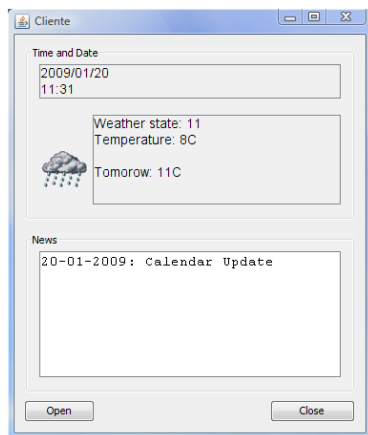

Fig. 10. Snapshot from the deprecated user interface.

The new interface was developed according to a new paradigm: the use of portable devices. The use of PDAs is the best way to grant portability and mobility to the application, allowing the user to move freely and, nonetheless, continue monitored and connected to the system. The new mobile interface can be seen in Fig. 11.

The interface was developed to be similar to standard Android interfaces so that users can quickly familiarize with it. In that sense, the learning curve of iGenda is reduced. The main objective was to have an intuitive interface that flows under the fingers of the user and that requires little or no training to use.
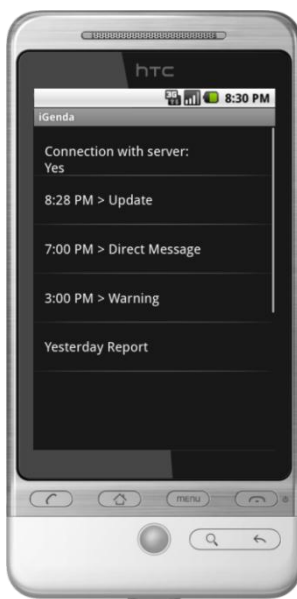

Fig. 11. New Android interface.

It is important to refer that the typical users should be used to operate devices such as PDA's, or require a minimal amount of training and adaptation. The Adaptive User Interface (AUI) is also an interesting feature to be added in a short term period. It consists on creating interfaces that can be adjusted to the user needs and preferences in terms of visualization and context. Given the target population of this project, the solutions given by the AUI are fairly interesting to serve correctly the results of the user, the way they were meant to be interpreted. Being an engineer, the developer sometimes forget that the user can only work with the interface provided and this is a core factor, and at the same level as all the modules and logical engines. But, as referred before, being the target population an elderly one, we must not forget that there are also various problems that affect commonly the elderly population. Problems like loss of vision and hearing are also very common at an older age, and interfaces provided by the operating system of the mobile de-

vices are not ready for those problems. Studies being done [19] in the visual and hearing problems suggests interfaces that provide, among other solutions, a high font resolution and audio/voice commands.

The tests made by projects that solve problems in the referred areas, that use computer applications, report bad results when used by persons that are less computer literate and substantial improvements when the users are used to operate computers, even if they have vision and/or hearing impairments. The roadmap of our project includes modular interfaces and pre-recorded commands, such as voicemails, to notify the users of the activities to perform. It is not our main objective to provide more advanced interfaces, such as active voice commands, as the main focus of the project is a technical solution, being the interfaces provided clear and simple as possible. We recognise the fact that the acceptance by the users can rely solely on the interface presented, but in an initial phase computer literate users will be our main users. This is done so they can surpass the possible interface problems and test the remaining modules of the iGenda system.

4.2 Medication Reminder

One of the worst consequences of memory impairments is a patient not remembering to take his drugs at the right time, or not remembering it at all. In that sense, a Medication Reminder module was developed. It consists of three agents that remind the user of the drugs he has to take, at the right time. Although this is apparently a simple problem, it is not so trivial to implement. Specifically, the agents consider the time to take the medication and the possible event of the user leaving the home, in which case an early notification must be issued in order for the patient to take the drugs with him. This module was developed independently rather than being integrated into the main iGenda application for two main reasons. The first is that the action of taking drugs is usually very brief and, thus, there is no need to schedule an event. The second reason is that the notifications should be different than the normal ones, with a higher degree of importance or urgency.
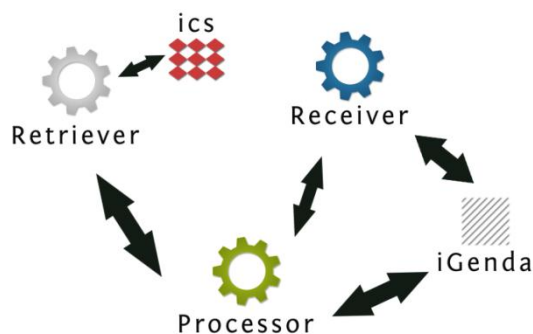

Fig. 12. Medication Reminder platform.

This module relies on 3 agents (Fig. 12): the Receiver, the Processor and the Retriever, with its functionalities as follows. The Receiver receives new schedules of drugs that must be taken by the user. This is done directly by the healthcare provider. The Retriever fetches the upcoming drug-related events and puts them in memory so that the Processor can perform correctly: it is the bridge between the database and the Processor. Finally, the Processor receives the information delivered by the Retriever and warns the user when the right time comes, prepares the next warning.

4.3 Sensing System

The Sensing System is a module that interfaces with sensors used directly on the user, acquiring information from his vital signs and health status [20]. This module can detect and prevent any issues that may threaten the health of the user, as it can detect problems that are occurring in real time. In fact, acquiring information about the patient's vital signs is the fastest and more reliable way to assess his health status. Therefore, it allows the system to know if the patient is under stress or in danger and also if he is able to perform the events scheduled. It also opens the door to intelligent and proactive scheduling. This means that, in a scenario in which the patient's health status is deteriorating over time, the Sensing System can connect to the iGenda and ask for an automatic scheduling of a potentially urgent medical appointment. The iGenda schedules the request in both the patient's and physician's agendas.
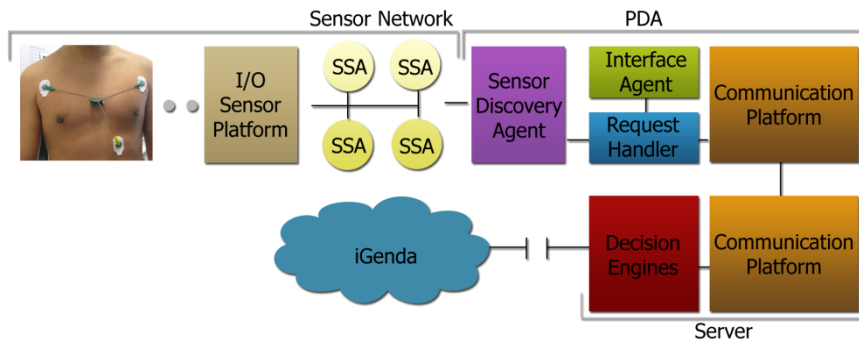

Fig. 13. The current architecture of the Sensing System

Moreover, this sensing system also makes use of GPS sensors, possibly available in the mobile device. The key idea is to have access to the current location of the user for two main purposes: determine travel overheads for efficient response in case of emergency.

This module is constituted by several sensors that capture the patient's vital signs, electrocardiogram, blood pressure, oximetry and temperature, processes them in order to be readable for Humans. From this point on, the data is interpreted and the system configured to read the data and, by means of well-defined clinical guidelines, trigger appropriate actions if necessary.

Currently, the sensing system in an advanced stage of development (Fig. 13), having already a beta version constituted by both software and hardware modules. It thus collects data for processing and interpreting the health status in order to create rules that can be used by the iGenda Core to better implement, schedule and manage the user's daily life and respective activities.

4.4 Centralized Management System

The main objective of the Centralized Management System (CMS) is to create a super iGenda system that supports multi-person scheduling and multi-layer calendars [21] [22]. It is being developed to work in medical environments such as hospitals and should not be seen as a sub-module of iGenda but rather as a fork..

The main objective is to control the routines of physicians and nurses and device functionalities to achieve:

- Better health provision;
- Higher control of time spent in each task;
- Improved routing through tasks;
- Improved selection of patient's priority according to health status.

This module is currently under development. Therefore, we are currently studying different algorithms that can deal with several users and their respective roles and tasks, keeping in mind the above proposed objectives. In face of the problems of NP-Complete complexity we are currently facing, the solution will most likely consist of genetic algorithms. The complexity of the problems arises mostly because a single routine route usually requires more than one person. Given that Human resources tend to be limited and that each one has his own tasks, a significant work of optimization and coordination must be performed.

5 Application Scenarios

In this project there are two type scenarios: *single user* and *institution*.

The *single user* scenario is one in which the user has an iGenda platform and has the possibility to be connected either to a healthcare institution or directly to a physician. In this scenario, the platform is optimized to adjust to each user, to respond to the user's necessities.

The *institution* scenario corresponds to the description of the CMS module. The institution will have a set of objectives and practitioners, like physicians and nurses, will comply with the instructions given by the CMS platform. In this case, the iGenda platform will optimize the work cycles of the institution, maximizing availability of Human workforce and other resources, resulting in advantages for the institution and the quality of the healthcare provided.

These scenarios require different types of tests and different types of implementations. The implementation will follow the typical test-improve method. In that sense, several personae were created for test purposes, each one with given roles, in a typical simulation platform. The personae were created with different demands, according to the specific purposes of each test. The validation was made by comparing the results of the system to the expected ones.

5.1 Results

The results of the implementation are depicting the failures and successes of the development path initially devised. The following examples represent a group of problems faced during development, the solutions adopted and the respective results. These solutions were tested in a simulation platform, with some students playing the several roles defined, in order to assess the behaviour of the whole system and cater for new developments. Thus, for the sake of realism, during the tests, each of these students represented one of the personas created. The use of the defined personas keeps the tests linear and sets a base pattern for the incoming results. This way, it was constructed a batch of possible and acceptable results to each persona, which were used to compare with the results of the simulation.
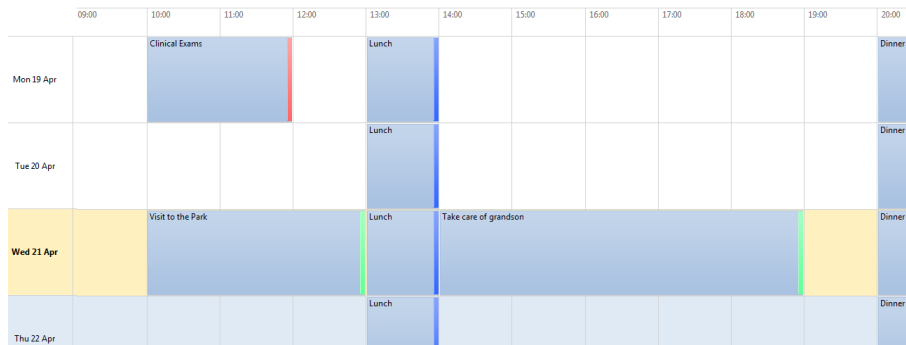

Fig. 14. Initial agenda with a few events.

Two personas were created to modulate the typical users, a *young worker* and *aged person*. The personas were created with several options within the configuration of each one, following the most common characteristics of each target of de-

mographics. For instance the *young worker* should have the FTM set up to the minimum possible execution. The results can be depicted as follows.

In the *young worker* persona the Free Time Manager was a challenge. To this persona, the free time should be totally controlled by him and the FTM should be a mere suggestive agent that should simply monitor the routines of the user and make suggestions in cases of free time.

In the *aged person* persona, results suggested that the family connection is not that important and the Free Time Manager was also a bit harsh in terms of scheduling. Moreover, the initial user interface was difficult to use by a person that was not a regular user of technology. The decisions of the Conflicts Manager in terms of rescheduling were also a bit offset and if another person was involved sometimes this person could not proceed to the event.
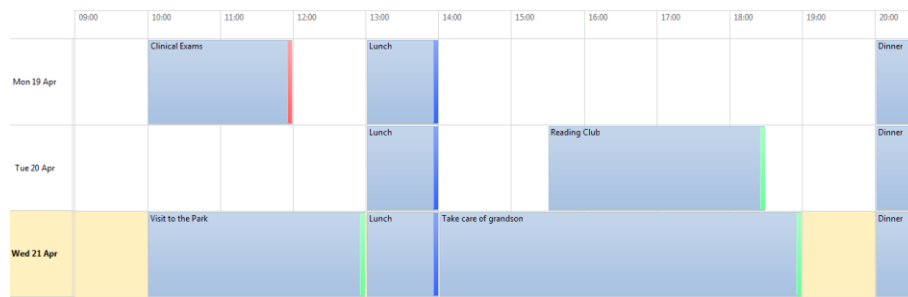

Fig. 15. Execution of the Free Time Manager with a low spreading.

The Free Time Manager also showed a problem of spreading, i.e., if there are several short activities, e.g. if a user has 4 different activities of 30 minutes each on a single hour, this becomes more stressful than relaxing to the user.
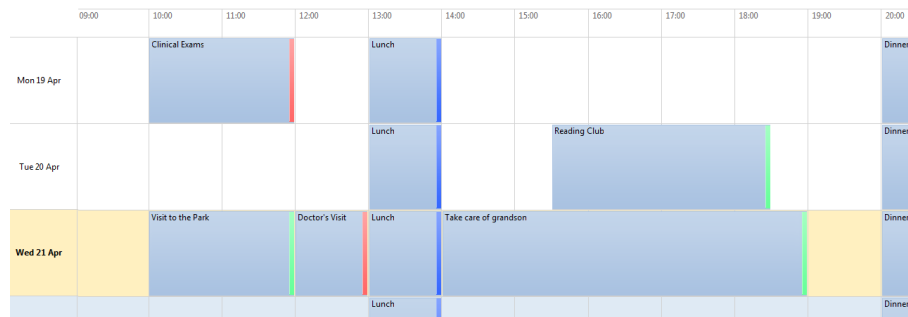

Fig. 16. A resizing of a non-important event with a scheduling of a priority 1 event.

Considering the Conflicts Manager, the reallocation of events did not always go as smoothly as expected. In multi-person activities, the rescheduling would not

always occur as expected if another person involved had a calendar filled with activities.

Moreover, the notification of errors was not always as expected. In terms of interfaces, the development philosophy was to keep them simple and minimalist, reducing the interaction needed (e.g. buttons, options). In that sense, the entire configuration must be done by a technician. The quest for usability and adaptive interfaces continues and usability guidelines will be considered from now on. To highlight the main functionalities that the interfaces provide, sequences of screenshots of the outcome of several tasks are presented in Figures Fig. 14 to Fig. 17.
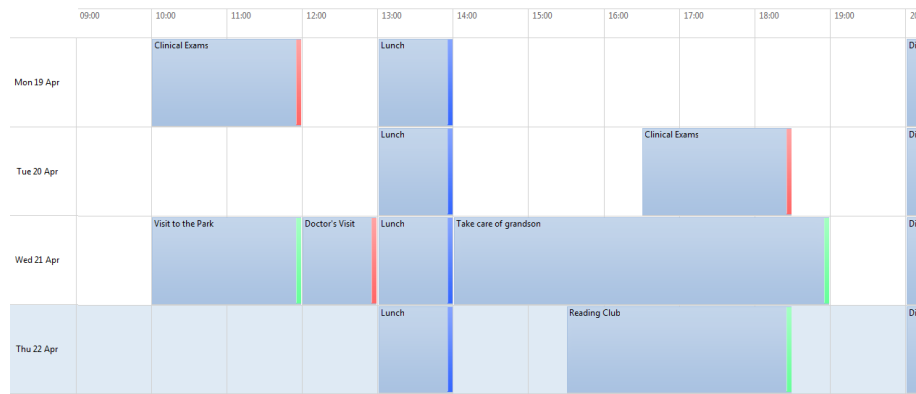

Fig. 17. Moving of a non-important activity and a scheduling of a priority 1 event.

6　Conclusions and Future Work

At this moment, this project follows several parallel lines of work. The main modules of the iGenda are fully developed and are already working in a beta state, allowing further higher-level services to be developed. Once the test phase is finished, an implementation in a real scenario is expected, using, at first, a selected controlled group of users. The rest of the modules will be gradually integrated as they enter a more mature stage of development, thus becoming available to its users. We expect CMS module to be the more challenging one to implement as it deals with medical real-life activities and, thus, risky environments.

Meanwhile, the strong acceptation of this project, by both the scientific and the medical community, encourages us to continue its development and proves that this is a valid path. Each new idea implemented opens new research doors and allows more functionalities, so possible future work is immense. Here, we include machine learning techniques to determine what combination or sequence of actions patients prefer. We are also considering the further development of the Sensing module in order to provide more information and possibly new ways of perceiving health-related problems and how to solve them.

It is our opinion that the little common things that we do every day are the ones that really make a difference in our well-being. This project is a life changing one as the characteristics made available to the users are simple in its essence but of a great value, specifically when one considers patients with memory impairments.

References

[1] UNFPA, *State of World Population 2009.*, 2009.

[2] Y. E. Geda et al., "Prevalence of neuropsy-chiatric symptoms in mild cognitive impairment and normal cognitive aging: population-based study," *Archives of General Psychiatry*, vol. 65, pp. 1193–1198, 2008.

[3] Neil Charness, "Aging and human performance," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 50, pp. 548–555, 2008.

[4] Ronald C. Petersen and Selamawit Negash, "Mild cognitive impairment: an overview," *CNS spectrums*, vol. 113, no. 1, pp. 45-53, January 2008.

[5] G. Tucker, "Age-associated memory loss: Prevalence and implications," *Journal Watch Psychiatry*, 1995.

[6] Donna R. Addis, Alana T. Wong, and Daniel L. Schacter, "Remembering the past and imagining the future: Common and distinct neural substrates during event construction and elaboration," *Neuropsychologia*, vol. 45, no. 7, pp. 1363-1377, 2007.

[7] Paulo Novais, Ricardo Costa, Davide Carneiro, and José Neves, "Inter-Organization Cooperation for Ambient Assisted Living," *Journal of Ambient Intelligence and Smart Environments*, vol. 2, no. 2, pp. 179–195, 2010.

[8] Ricardo Costa, Paulo Novais, Ângelo Costa, and José Neves, "Memory Support in Ambient Assisted Living," in *Leveraging Knowledge for Innovation in Collaborative Networks*, Luis Camarinha-Matos, Iraklis Paraskakis, and Hamideh Afsarmanesh, Eds.: Springer Boston, 2009, vol. 307, pp. 745-752, 10.1007/978-3-642-04568-4_75.

[9] J. Jiang, A. Geven, and S. Zhang, "Hermes: A fp7 funded project towards computer-aided memory management via intelligent computations," 2009.

[10] L. Williams et al., "Sensecam: A retrospective memory aid," 2006.

[11] Peter J Brown, *GC3: Memory for Life: Getting Things Back.*, 2004.

[12] R. Ashri, M. D'Inverno, and M. M. Luck, *Agent-Based Software Development.*: Artech House Publishers, 2004.

[13] Michael Wooldridge and Nicholas R. Jennings, "Intelligent Agents: Theory and Practice," *Knowledge Engineering Review*, vol. 10, pp. 115-152, 1995.

[14] Antonio Moreno, Aida Valls, and Alexandre Viejo, "Using JADE-LEAP to implement agents in mobile devices," 2003.

[15] Marion J. Ball and Jennifer Lillis, "E-health: transforming the physician/patient relationship," *International Journal of Medical Informatics*, vol. 61, pp. 1-10, 2001.

[16] Ângelo Costa, Paulo Novais, Ricardo Costa, José Machado, and José Neves, "A Memory Assistant for the Elderly," in *Intelligent Distributed Computing III*, George Papadopoulos and Costin Badica, Eds.: Springer Berlin / Heidelberg, 2009, vol. 237, pp. 209-214, 10.1007/978-3-642-03214-1_21.

[17] Ângelo Costa, Paulo Novais, Ricardo Costa, Juan Corchado, and José Neves, "Multi-agent Personal Memory Assistant," in *Trends in Practical Applications of Agents and*

Multiagent Systems, Yves Demazeau et al., Eds.: Springer Berlin / Heidelberg, 2010, vol. 71, pp. 97-104, 10.1007/978-3-642-12433-4_12.

[18] Juan A. Fraile, Dante I. Tapia, Sara Rodriguez, and Juan M. Corchado, "Agents in Home Care: A Case Study," , 2009, pp. 1-8.

[19] António Teixeira et al., "Speech as the Basic Interface for Assistive Technology," in *DSAI'09, DSAI Software Development for Enhancing Accessibility and Fighting Info-exclusion*, 2009.

[20] Davide Carneiro, Paulo Novais, Ricardo Costa, Pedro Gomes, and José Neves, "EMon: Embodied Monitorization," , 2009, pp. 133-142.

[21] K. Li and J. Decker, "Coordinated hospital patient scheduling," *ICMAS '98: Proceedings of the 3rd International Conference on Multi Agent Systems*, 1998.

[22] G. Syswerda, *Handbook of Genetic Algorithms*., 1991.

[23] Anind K. Dey, Achilles Kameas, and Carlos Ramos, "Thematic issue on contribution of Artificial Intelligence to Ambient Intelligence," *Journal of Ambient Intelligence and Smart Environments*, vol. 1, no. 3, 2009.