Accepted Manuscript

A model-driven approach for constructing ambient assisted-living multi-agent systems customized for Parkinson patients

Iván García-Magariño, Guillermo Palacios-Navarro

PII: S0164-1212(15)00208-3
DOI: 10.1016/j.jss.2015.09.014
Reference: JSS 9589

To appear in: *The Journal of Systems & Software*

Received date: 26 June 2015
Revised date: 9 September 2015
Accepted date: 11 September 2015

Highlights

- The approach develops multi-agent systems customized for Parkinson patients.
- It was compared with three alternatives by 24 developers and 13 patients.
- The presented model-driven approach reduces development time over alternatives.
- The obtained applications are more efficient in terms of response time.
- It improves the usability and other aspects according to patients and developers.

A model-driven approach for constructing ambient assisted-living multi-agent systems customized for Parkinson patients

Iván García-Magariño*,[a], Guillermo Palacios-Navarro[b]

[a]*Department of Computer Science and Engineering of Systems, University of Zaragoza, Escuela Universitaria Politécnica de Teruel, c/ Ciudad Escolar s/n, 44003 Teruel, Spain*
[b]*Department of Electronic Engineering and Communications, University of Zaragoza, Escuela Universitaria Politécnica de Teruel, c/ Ciudad Escolar s/n, 44003 Teruel, Spain*

Abstract

The Parkinson disease affects some people, especially in the last years of their lives. Ambient assisted living systems can support them, especially in the middle stages of the disease. However, these systems usually need to be customized for each Parkinson patient. In this context, the current work follows the model-driven engineering principles to achieve this customized development. It represents each patient with a model. This is transformed into an agent-based model, from which a skeleton of programming code is generated. A case study illustrates this approach. Moreover, 24 engineers expert in model-driven engineering, multi-agent systems and/or health experienced the current approach alongside the three most similar works, by implementing actual systems. Some of these systems were tested by Parkinson patients. The results showed that (1) the current approach reduced the development time, (2) the developed system satisfied a higher percentage of the requirements established for certain Parkinson patients, (3) the usability increased, (4) the performance of the systems improved taking response time into account, and (5) the developers considered that the underlying metamodel is more appropriate for the current goal.

Key words: Agent-oriented software engineering, metamodel, model-driven

*Corresponding author
Email addresses: ivangmg@unizar.es (Iván García-Magariño), guillermo.palacios@unizar.es (Guillermo Palacios-Navarro)

engineering, multi-agent system, Parkinson disease

1. Introduction

According to a recent study [11], Parkinson Disease (PD) is affecting the population with many variants due to the different mutations of the gene that encodes the F-BoX Only protein 7. In particular, 17 variants were detected in the southern of Spain. Although some factors can influence the risks of suffering this disease, such as smoking, there is a percentage of people with a certain gene that unavoidably suffer PD in the last years of their lives [18]. In these cases in which the most advanced medicine research cannot avoid the symptoms, patients need other kinds of assistance for getting along with their disease. The social environment including the family members usually becomes a pillar of support for overcoming the needs of patients [14]. However, in some cases there is not any member that can exclusively dedicate their life to take care of a patient. In these cases, the different familiar caregivers have to coordinate for taking care of the patient, and in some hours of the day the patient may not have any human company.

In this context, this work proposes customized Multi-agent Systems (MASs) as a solution for coordinating partial human caregivers and assisting patients in some needs. This work proposes MASs as these have proven to be effective for coordination and collaboration. This can be observed in the later works about MASs such as the Real-time Order-driven approach for collaborative production planning and scheduling [21]. In particular, the current approach develops the customized MASs with a Model-driven Engineering (MDE) approach. This approach includes a metamodel for defining a Modeling Language (ML). In this, each model determines all the features of a patient including (a) their social environment with home members and other caregivers, (b) their symptoms, (c) the skills in which the patient needs assistance, and (d) their economical circumstances. In addition, there are some Model Transformations (MTs) from the mentioned ML to initial MAS models. MASs can be developed from these models after refining these. In particular, the remaining development of each MAS is recommended to follow the Ingenias methodology, which was previously introduced in [22]. This methodology has been selected because it already follows a MDE approach including the generation of programming code. This methodology has also been chosen because the underlying metamodel of its tool support, i.e. the

Ingenias Development Kit (IDK), had already been defined with the ECore language. In this manner, this work can use the Atlas Transformation Language (ATL) to define MTs from the current metamodel to the Ingenias metamodel.

The current work introduces the development of a complete functional Ambient Assisted Living (AAL) MAS with an application for mobile devices (i.e. smartphones and tablets) as a case study. In addition, the current approach has been compared with the most relevant and similar ones, which are presented by (1) Calvillo et al. [3], (2) Lopez and Blobel [17], and (3) Raghupathi and Umar [23]. For this comparison, 24 engineers expert in MDE, Agent-oriented Software Engineering (AOSE) and/or health experienced the current one and the others. They developed applications customized for particular PD patients with each approach. Some of these applications were tested by real PD patients. In particular, the evaluation of the present work compares the development time, the satisfied requirements, the usability, the response time and the perceptions of experts.

The current approach was previously introduced in [7], but it has been extended in several ways. The metamodel has been enhanced by including new concepts such as the ones related with treatments, tests, caregiver symptoms, timetables, other diseases, other patient symptoms, dressing and going out. The MTs have been refined to obtain more useful output models. The paper now graphically presents some of these MTs. The development of the case study now shows a complete functional system. The paper now also includes the aforementioned experimental comparison evaluation.

The remaining of the paper is organized as follows. The next section briefly introduces the background comparing some related works to the current one. Section 3 presents the model-driven approach for developing and customizing MASs for PD patients. Section 4 presents the modeling of a particular PD patient and the development of a customized MAS for its assistance, as a case study. Section 5 compares the current approach with others by analyzing the experience of the group of experts and PD patients. Lastly, section 6 mentions the conclusions and future lines of research.

2. Related work

2.1. MDE and MDA

There are several works that use MDE approaches for developing ambient intelligent systems. In particular, the FamiWare [5] framework uses a MDE approach for developing ambient intelligent systems, with models determining the cardinality-based features. Their process propagates the changes made in the feature level to the different components of the FamiWare middleware. In particular, this framework can be used to evolve AAL intelligent systems. In addition, Reubi [24] is a method for acquiring requirements and then following a MDE approach for constructing ubiquitous systems. Their authors present this method by developing an AAL healthcare system.

In the literature, several metamodels have been defined for modeling patients in general. For example, Calvillo et al. [3] present a metamodel that is integrated in a healthcare system. In this system, each patient can determine who can access their information such as demographic data, health, well-being and social conditions. This metamodel defines the information regarding each patient considering three main groups of actors: people (e.g. nurses, relatives and friends), organizations and healthcare devices. In addition, Lopez and Blobel [17] have developed a framework for achieving semantical interoperability in health information systems. This framework uses the Model-driven Architecture (MDA) approach, defining the corresponding metamodels for the computation-independent models, platform-independent models and the platform-specific models. Moreover, Raghupathi and Umar [23] apply the MDA approach for developing healthcare systems. Their metamodels are mainly focused on defining the models of the health clinics. In particular, a platform-independent model is created for each clinic, and this is transformed into several platform-specific models.

Kevoree [4] is a framework for designing and deploying distributed adaptive systems following the Model@Runtime paradigm. It has a layer to manage different types of nodes such as sensors and mobile devices that are close to the users. It also considers that some nodes may only be connected sporadically, in order to manage the differences between the states of the sub-systems. In addition, EntiMid [19] is a middleware that manages certain nodes in house automation for assisting elderly people. This middleware satisfies a list of requirements desirable in distributed applications for the aging population.

Nevertheless, all these works do not generate programming code for impersonating real users and coordinating them. By contrast, the current work generates programming code for this coordination in which some agents impersonate real people according to their preferences and circumstances. In particular, MASs are an appropriate choice for impersonating people that coordinate between each other, like in the current approach.

2.2. MASs

There are several MASs that have been developed for AAL for elderly people or people suffering a disease. For instance, Kaluza et al. [16] present a MAS that assists elderly people that are living on their own at home, in order to prolong their independence. This system can detect an emergency situation in real time by means of several sensors. The system detects domestic accidents with several facts such as vertical acceleration or a frozen weird position for a long time. This MAS was tested in a nearly-realistic room with several movements. This MAS could be used for people who suffer PD.

Moreover, Nefti el al. [20] present a MAS for AAL of people suffering the dementia disease. In fact, this MAS keeps patients observed in an unobtrusive way, and warns them of possible risks. It alerts the local authority when a risk is ignored. Furthermore, Su [26] introduces a framework for e-health monitoring in wide areas such as metropolitan and national. This framework contains mobile agents conforming MASs. These MASs allow caregivers to monitor the patients with light-weight portable devices, without interfering their daily activities.

Furthermore, Benhajji et al. [1] present a MAS that coordinates all the hospitals resources and control the patients flow. Their aim is to improve the planning of healthcare resources for patients and to efficiently manage unpredictable disruptions.

All these works present MASs that assist healthcare in different ways. However, these works do not provide a proper interface that actually lets patients to guide their assistance. For instance, these works do not include interfaces with speech recognition for patients with hand shaking. These neither offer immediate technological treatments for psychological aspects. These neither allow patients to establish their own timetable to coordinate with their caregivers. To the best of authors' knowledge, the current work is the first one that takes all these aspects into account.

2.3. MDE for MASs

Gascueña et al. [9] use a model-driven approach for developing MASs, using the set of the Eclipse modeling tools. In particular, this work defines the Prometheus metamodel with the ECore language, and generates the corresponding graphical modeling tool by means of the Graphical Modeling Framework. In addition, Ghorbani et al. [10] present a model-driven approach for developing agent-based simulators. Their work formalizes knowledge of social sciences, and represents collaborative relations among individuals. In their approach, the user can define the decision-making process of the agents and their actions.

Moreover, a metamodel is proposed for determining the security requirements in MASs [2]. This work uses a MDE approach for transforming high-abstraction models into code-specific models, so MASs can be constructed considering the security requirements. Another metamodel is specifically designed for designing robotic MASs with the Gaia methodology [25]. For instance, this metamodel can determine the environment of robotic agents, sub-organization of agents, interactions and certain kinds of roles. MAS-ML 2.0 [13] is defined with a metamodel that integrates heterogeneous agent-oriented architectures. In this way, MASs can integrate proactive agents, reflex agents, goal-based agents and utility-based agents.

All these works mainly focus on the definition of MAS models for then developing the systems. However, all these works miss the definition of a model of the intended user as a previous step. This step would allow developers to customize the development of each MAS to a specific user, obtaining a final system adapted to their specific needs. Thus, the current work is the first one that develops MASs with a MDE approach including the modeling of the user in the model-driven chain.

3. A model-driven approach for customizing the development of AAL MASs for each PD patient

A model-driven approach is proposed for developing and customizing a MAS for each PD patient with their particular social, economical and symptomatic circumstances. The present approach guides the developers with the phases that are illustrated in Figure 1. Firstly, a developer defines a patient model, conforming the novel metamodel of the current approach. Then, the MTs of this approach automatically transform this model into a MAS model. After this, IDK can load this model and generate a programming code skeleton [22]. Finally, the developers extend this skeleton until obtaining the proper application.

The patient diseases may evolve. The developers have two options after changing the patient model. For minor changes in the patient model, such as the incorporation of a single symptom, the developers are advised to manually include the corresponding software module without going through all the phases, transformations and code generation. In case of major changes, the developers are advised to run again the MTs but using the ATL Refining

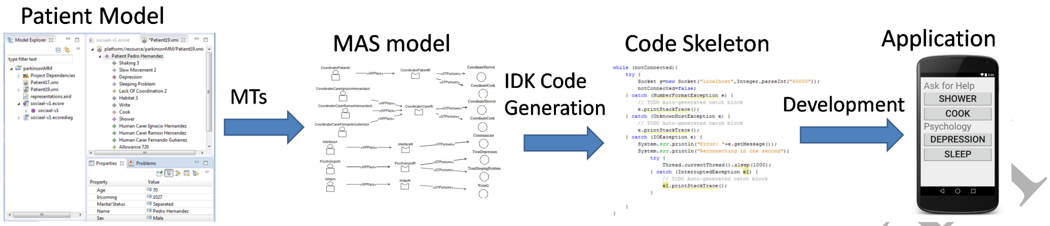


Figure 1: Overview of the current approach

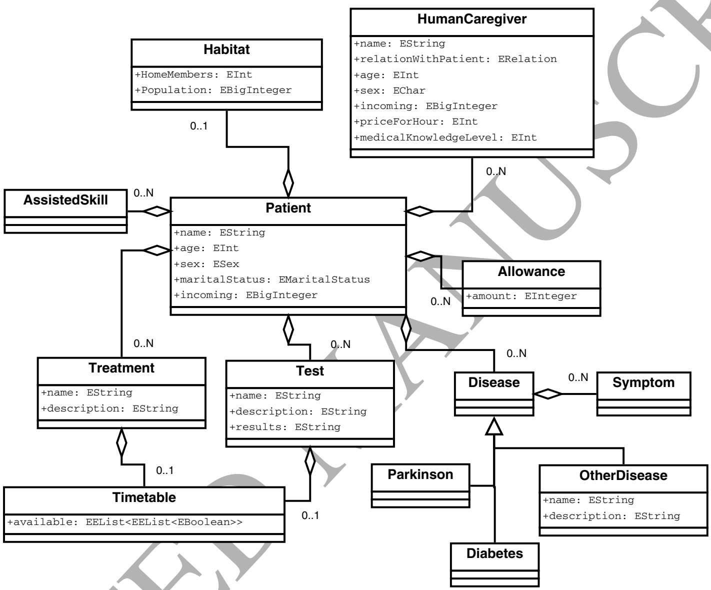


Figure 2: Excerpt of the metamodel for modeling PD patients

Mode, to only add or change the necessary elements in the target MAS model without overwriting the other elements. Then, IDK avoids overwriting the sensitive components when regenerating the programming code skeleton, as indicated in the work of Gomez-Sanz et al. [12].

Subsection 3.1 presents the metamodel for modeling each patient and their circumstances, whereas subsection 3.2 introduces the MTs for creating a MAS model from each patient model.

3.1. Metamodel for modeling PD patients

In this approach, the metamodel defines a ML that describes patients suffering PD. In particular, each model of this ML represents a patient with all the features concerning their illness, and their social and economical circumstances. Thus, the "patient" is the central concept of the model, and will be represented with the root element of the model. Figure 2 shows the excerpt of the metamodel that concerns the patient and all its surrounding concepts. As one can observe, the social environment of the patient is determined with the "habitat" and "human caregiver" concepts. The economical circumstances are represented with the "allowance" concept and the "incoming" attribute of the human caregiver and patient concepts. The features of the patient concerning their illness are represented with their symptoms and the skills for which they need assistance. All these aspects of a patient influence in the MAS that can assist them, and consequently are taken into account in the metamodel.

The patient can also receive regular treatments and tests. These treatments and tests can be named and described within the metamodel with the "treatment" and "test" concepts. The tests can store the results of their last occurrence. These concepts can be assigned to a specific timetable. The "timetable" concept will be introduced later when describing the human caregivers.

In some cases, the PD patients can also have other diseases such as diabetes or similar ones. For this reason, the patient is related with the "disease" concept, which is extended with concepts such as "Parkinson" and "diabetes". In particular, the "other disease" concept allows one to name and describe any disease, since having a complete hierarchy of diseases is out of the scope of the current work. Each disease is related with the symptoms that the patient is actually suffering from.

In the metamodel, the human caregivers are modeled as one can observe in Figure 3. Each caregiver has two different timetables indicating their availability. The first timetable determines when they are available in their daily routine for common situations. The second timetable indicates their availability for emergencies. The latter timetable should include much many hours than the former one. Both timetables are represented with two nested lists of booleans that represent a matrix considering the days of the week and the different hours of each day. The metamodel can also indicate the common symptoms of caregivers such as exhaustion, depression and economical pressure with the "caregiver symptom" concept and its hierarchy of concepts. Generally, these symptoms are only specified for very close family members. The generated application usually reduces the workloads of the caregivers suffering from the highest levels of exhaustion or depression if possible. The

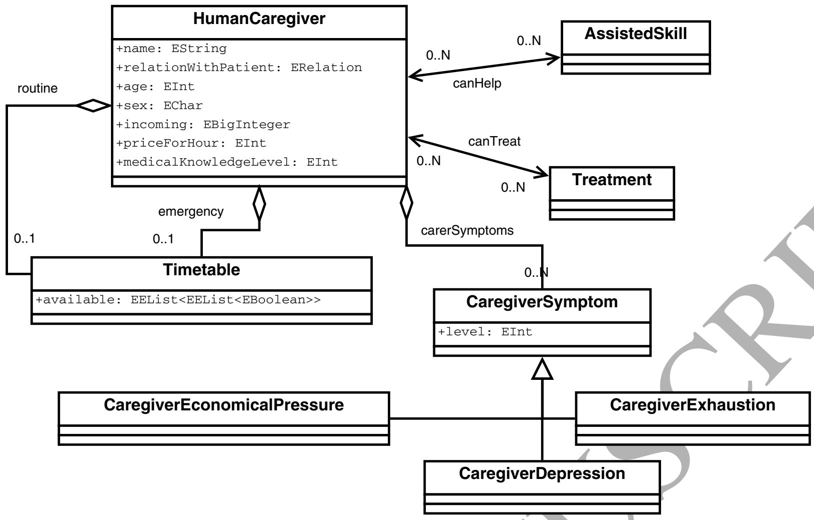


Figure 3: Excerpt of the metamodel for modeling human caregivers

levels of economical pressure of close family members are useful for estimating what AAL equipment they can afford. The intensity level of each symptom is indicated through an integer from one (low) to five (high) that is inherited by all the concepts of the hierarchy. Each human caregiver is related with (1) the skills of the patient that they can assist and (2) the treatments that they can provide. The human caregiver also indicates their relation with the patient, their incoming (in case of familiar or friends), their price per hour (in case for example of a nurse), and their level of medical knowledge with an integer from one (low) to five (high).

In the metamodel, the symptoms are classified into two different kinds of symptoms, which are physical symptoms and psychological symptoms, as one can observe in Figure 4. This classification is determined by extending the "symptom" metamodeling concept with these two kinds. Each of these kinds is extended with concrete symptoms, which are detailed in the same figure. There are also some generic concepts (i.e. "other physical symptom" and "other psychological symptom") that can detail other symptoms with their names and descriptions. In this way, the metamodel can represent patients that have symptoms of PD and other diseases. In the future, developers can add more concrete symptoms by including the corresponding metamodeling concepts if necessary, without altering the structure of this metamodel. It is worth indicating that the level of intensity of each symptom of a patient is

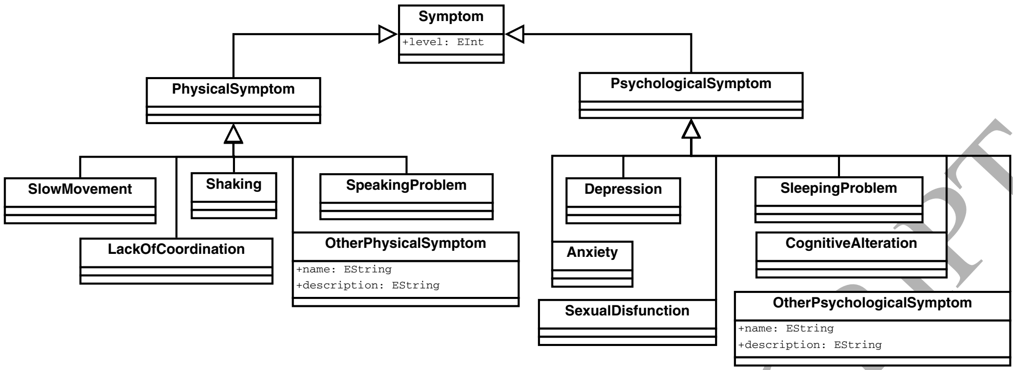


Figure 4: Excerpt of the metamodel for modeling PD symptoms

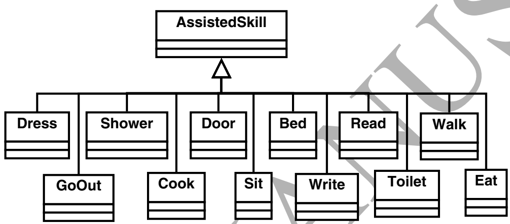


Figure 5: Excerpt of the metamodel for skills of PD patients that need to be assisted

determined with the "level" attribute from one to five. The MASs for PD patients depend on their symptoms, mainly for determining the interface of communication between each patient and its customized MAS.

The ML of the metamodel can also specify the particular skills in which a patient needs to be assisted. A PD patient may need assistance for dressing, going out, having a shower, cooking, opening a door, sitting down, going to bed, writing, reading, going to the toilet, walking or eating. As one can observe in Figure 5, each of these particular skills is represented with a metamodeling element that extends the "assisted skill" metamodeling element.

Each AAL MAS is customized according to the skills in which each patient needs to be assisted. For instance, the patient may only need assistance in skills that only take place few times per day or only in a specific part of the day (e.g. having a shower or going to bed). In this case, if the human caregivers are partially available, the MAS can coordinate them and the patient to achieve the necessary assistance. Otherwise, if the patient needs more assistance than the human caregivers can offer, then the MAS may need extra devices. In these cases, the necessary devices depend on the nature of the skills that are assisted.

A modeling editor tool has been automatically generated from the presented metamodel by means of the Eclipse Modeling Framework. This generated tool allows developers to define models that represent PD patients and their circumstances. In most cases, the developer mainly has to define a patient entity associated with a Parkinson entity, conforming to the metamodel excerpt of Figure 2. Then, the patient entity is usually associated with the human caregiver entities. These are related with the entities of their timetables and the entities of the skills that they can assist. This conforms to the metamodel part of Figure 3. Finally, the patient entity must be related with entities of the specific subclasses of the symptom concept (see Figure 4). The patient entity must also be related with entities of the specific subclasses of the assisted skill concept (see Figure 5). In some specific cases, developers may need to customize the models with some of the remaining entities. The effort of customizing models is medium, but it is compensated with the decrease of the global development time of applications. Section 5.1 will show that the development time with the current approach is shorter in average than with other similar approaches. In this comparison, the time for customizing models was computed as part of the development time.

The presented metamodel is an object-oriented model, and it takes advantage of the inheritance. For instance, the symptoms are expressed as a hierarchy by means of the inheritance (see the previously introduced Figure 4). The symptom concept has the information that is common to all symptoms (e.g. the intensity level), and this information is inherited in all the specific concepts of the hierarchy, including the specific physical and psychological symptoms. In this way, the specific symptoms are specializations of the generalized symptom concept that is root of the hierarchy. Similarly, the inheritance is applied for specializing the assisted skills (remind Figure 5), the diseases (in Figure 2) and the caregiver symptoms (in Figure 3). The customization of patient models is performed by instantiating the presented metamodel. Thus, this customization takes advantage of the inheritance, since it instantiates a metamodel that uses the generalization and the specialization through the inheritance.

It is worth noting that the current approach may have a steep learning curve, especially for developers that are not expert in MDE. In order to mitigate this downside, the current approach includes documentation and tutorials for guiding developers in their learning process.

The presented metamodel is compatible with defining several diagram

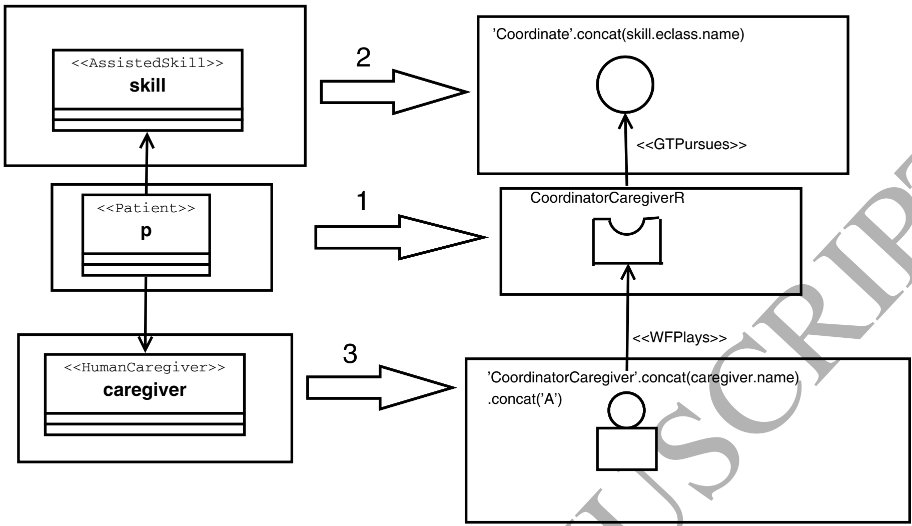


Figure 6: The MT for caregivers

types for its modeling language. This can be performed with the Diagram-type Editor Tool following the approach presented in our previous work [8].

3.2. MTs for the development of MASs for PD patients

Several aspects of each PD patient condition the architecture and behavior of an AAL MAS. Hence, in this approach, the information formalized in the proposed metamodel is taken into account for customizing a MAS for each particular patient. However, the association between the specific metamodeling elements of the proposed metamodel and the elements of a particular AOSE metamodel is not straightforward.

This work includes some MTs from the proposed metamodel for PD patients to the metamodel of the Ingenias AOSE methodology that is defined with ECore. The MTs are defined with ATL [15], since this language is appropriate for transforming models that are instances of ECore metamodels.

To begin with, there is a group of MTs that concern the coordination of the patient and human caregivers. The first MT of this group transforms each human caregiver element into a "coordinator caregiver agent" copying most of their attribute values, and creating the corresponding role and goals. This MT has three rules, which are illustrated in Figure 6. The source model prototypes are in the left side, while the target model prototypes are in the right side. The arrows represent the rules, and these are enumerated to in-

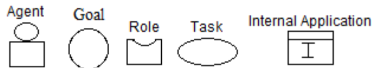


Figure 7: Notation of the Ingenias ML for the basic agent-oriented concepts

dicate their order of execution. Our previous thesis document [6] explicitly describes the conversion of this MT graphical notation to ATL. The first rule creates the "coordinator caregiver role". The second rule creates a goal for coordinating the assistance of each skill that only needs to be assisted a few times per day. The third rule creates a coordinator caregiver agent for each human caregiver. Each coordination goal is to schedule a timetable in which each patient need is covered with the minimum effort from behalf of the human caregivers. Each caregiver agent interacts with its human caregiver, generally in a mobile device. This agent warns the human caregiver when necessary. It can also update the restrictions of a human caregiver, such as their availability timetables. The second MT transforms the patient element into a "coordinator patient" agent that plays a role with the same goals as the coordinator caregiver agents. In brief, the coordinator caregiver agents and the coordinator patient agent interact with each other in order to guarantee that the patient is assisted when needing a human caregiver. The communication among caregiver agents and the coordinator patient agent is established by means of interactions composed of interaction units (i.e. messages). The coordinator patient agent delivers a timetable with the routine patient needs to all the caregiver agents. Each caregiver agent replies with their time availability. Finally, the coordinator patient agent fills the timetable with the caregiver agents, minimizing the effort from behalf of the corresponding human caregivers. Then, the coordinator patient agent broadcasts this filled timetable to all the caregiver agents. The caregiver agents show this timetable to the corresponding human caregivers.

Figure 7 indicates the notation of the Ingenias ML for the basic agent-oriented concepts that are used in the current article, so that readers can understand the graphical notation of MTs and the diagrams generated by these. In general, in the MTs of the current approach, the names of the created agents, roles, tasks and internal applications have respectively the suffixes "-A", "-R", "-T" and "-IA", in order to avoid conflicts of names and make the design of the MAS clear.

There is another set of MTs that regard the creation of an adequate "interface" agent with a role and a goal, according to the physical symptoms.

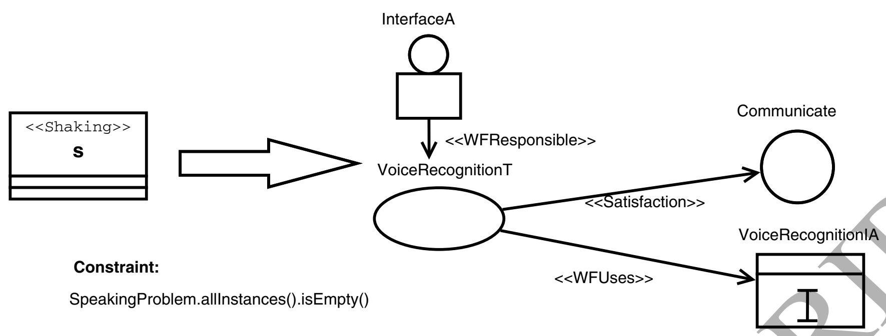


Figure 8: The MT for creating an interface agent with voice recognition when necessary

The interface agent is the responsible for a suitable and fluid communication between the patient and the MAS. Specifically, the interface agent receives the petitions from the patient, and makes these requests to some of the other agents. For instance, if the patient needs a human caregiver, then the interface agent indicates so to the coordinator patient agent, which can search for a human caregiver through the coordinator caregiver agents. The interface agent also provides some responses to the patient if convenient. In particular, a MT creates an interface agent with the ability of recognizing the human voice of the patient, if the patient has the symptom of shaking and does not have the symptom of speaking problem. This MT is illustrated with Figure 8, and has a constraint for checking the absence of speaking problems in the patient. Conversely, if the patient has a speaking problem, another MT creates an interface agent with the keyboard communication. If the patient has both symptoms of shaking and speaking problem, the MTs compare their intensity levels in order to create the interface agent with keyboard communication, voice recognition or both kinds of communication.

Some MTs are aimed at creating a "psychologist" agent that imports certain modules for treating the particular psychological symptoms, such as depression, anxiety and sleeping problems. Another group of MTs adds certain modeling elements to the MAS for assisting the skills that may not be able to be assisted by human caregivers because for instance these are necessary a high number of times per day. Some skills like reading and writing can be directly attended by the generated system with only the corresponding agents and software modules. However, the assistance in certain skills such as dressing, going out, eating, having a shower, walking, opening doors, cooking, sitting down and going to bed may require specific additional hardware components. The incorporation of these may depend on what the patient and their caregivers can afford according to their incomings.

4. Case study of modeling a PD patient for the MDE development of a customized MAS

The current approach is exemplified with a case study. A PD patient is modeled with the modeling editor tool generated from the presented metamodel. This PD patient is called Pedro Hernandez and his home has three members including himself. The other two members are two sons, called respectively Ignacio and Ramon. Both of them have jobs and act as human caregivers. The patient has also a nephew called Fernando Gutierrez, who lives in the nearby and also is a human caregiver. The patient has mainly three physical symptoms, which are shaking, slow movement and lack of coordination. The level of the first symptom is three, while the level of the others is two. His psychological symptoms are depression and sleeping problem. Finally, he needs assistance for writing, cooking and having a shower. Although this case study is based on a real case, this article has used pseudonyms for its presentation in order to keep the identity of the patient and human caregivers confidentially.

The MTs of this approach were applied to transform the patient model into an initial design model in the Ingenias ML for the development of a MAS for assisting his life. An excerpt of the generated agent diagram is presented in Figure 9. Firstly, one can observe the agents for coordinating the patient with his human caregivers. There are two roles for these agents, which are the "coordinator patient" role and the "coordinator caregiver" role. A MT creates an agent playing the former role since the system supports only one patient. A rule of another MT (i.e. rule 3 of previously introduced Figure 6) matched three times for the three input human caregivers. This rule outputted three different agents that play the latter role. The names of these agents were the result of concatenating the "Coordination Caregiver" string with the specific name of each human caregiver and the "A" suffix. In this way, these three agents can be easily distinguished from each other. Both roles pursue the goals of coordinating the assistance of the patient skills that only need to be assisted a few times per day, which are (1) to have a shower and (2) to cook. Notice that the cooking of lunch and dinner can be done together if necessary. In addition, there is an interface agent that guarantees a fluid communication with the patient. The psychologist agent with its

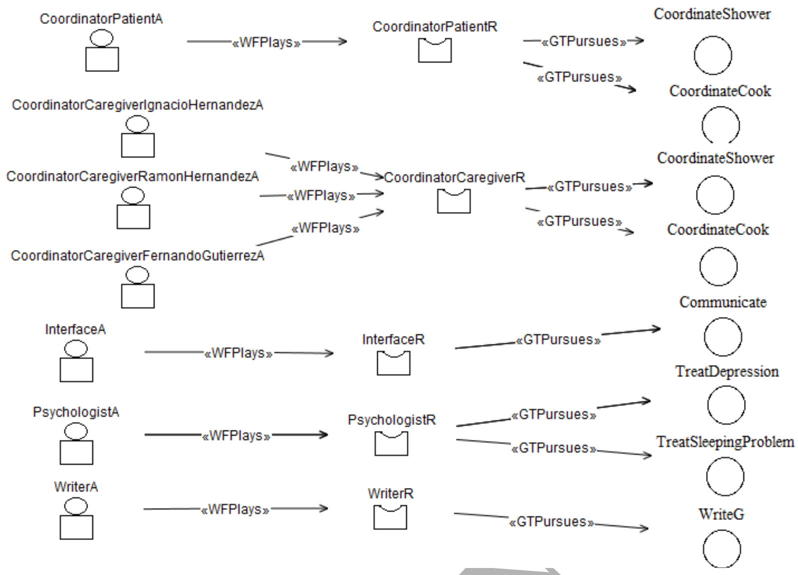


Figure 9: Excerpt of the agents diagram of the generated MAS design

role pursues to treat the two different psychological symptoms, which are (a) depression and (b) sleeping problem. Finally, a writer agent is generated with its role and goal for writing in a browser or a document whenever the patient requests so.

Figure 10 presents the main interface of the resulting application for the PD patient in a smartphone emulator. This interface uses a big letter font in buttons to make its usage simpler for PD patients. The patient user can ask for help in their skills that need assistance according to the patient model, i.e. having a shower and cooking respectively with the “shower” and “cook” buttons. In case that the user presses any of these buttons, the AAL MAS coordinates which caregiver should assist the patient, and the MAS alerts this caregiver. This coordination considers (1) the caregivers timetables, (2) their relations with the patient, and (3) the previous assignments to distribute the caregivers workloads. The patient can also request assistance for a psychological symptom of the PD. The application provides assistance for the symptoms determined in the patient model, i.e. depression and sleeping problems respectively with the “depression” and “sleep” buttons. In case of depression, the psychologist agent asks the user the reason of their depression. The user speaks aloud the reason, and the writer agent transcribes

Figure 10: Application for PD patients according to a patient model

the response. For now, the application searches for keywords such as "loneliness", "alone", "tired", "bored", "incapable", "unable" or "useless" in the user response. Then, the application plays some previously recorded advice(s) with human voice for the corresponding keyword(s) found. Moreover, for now, the sleeping problem is treated by reproducing some relaxing music randomly selected from a list of tranquilizing songs during a limited amount of time. The default duration is 25 minutes, but it can be changed according to the patient.

Developers were advised by a psychology professional for developing the treatments of this application for depression and sleeping problems. Two psychology professionals different from the previous one evaluated this part of the application. They agreed that this application can be helpful to alleviate these psychological symptoms in some cases. However, they also indicated that if the patient does not feel any progress, he should contact a real human psychologist.

One may think that if a person is completely unable to have a shower or to cook, they may be unable to use a mobile application too. The user of this case study was not completely unable to perform these daily tasks, but he preferred to be assisted to avoid the high risk of falling in the shower or getting burned when cooking. Thus, he chose the mobile application, which he was able to properly use without any known risk. However, in more advanced stages of PD, patients may not even be able to use a mobile application. Thus, the current approach is planned to be extended with another user interface in the future.

5. Evaluation

The experimentation has been designed and implemented following the guidelines of Wohlin et al. [28]. The empirical study of the current approach combines exploratory research and explanatory research. The exploratory research includes a case study, which was presented in the previous section. The explanatory research mainly focuses on identifying the effects of the current approach in comparison to similar approaches. The main research questions are aimed at determining (a) how much time is necessary for developing applications for PD patients, (b) in what measure the developed applications meet the needs of PD patients, (c) how much usable these applications are for them, (d) how much efficient the developed applications are in terms of response time, and (e) how much useful the different metamodel properties are for the developers. This explanatory research has been performed with a fixed design, which received input from quantitative data. Some of these research questions were analyzed with experiments measuring some objective aspects, such as the development time and the response time. Other research questions needed to survey patients or developers, but their responses were quantified and most of these were statistically analyzed.

The most similar approaches are presented in the works of (1) Calvillo et al. [3] for obtaining a healthcare system in which each patient can determine who can access their information, (2) Lopez and Blobel [17] with a model-driven framework for achieving semantical interoperability in health information systems, and (3) Raghupathi and Umar [23] for defining the models of the health clinics with a MDA approach. These works have been further introduced in section 2.

The compared works explicitly provided the metamodels. However, these works just introduced some transformations and in some cases mechanisms for code generation, without explicitly providing any of these. Hence, the authors of the current work interpreted their descriptions, and implemented the transformations and code generation as similar to their descriptions as possible. In addition, the metamodels were adapted to include some concepts such as (a) some necessary Parkinson symptoms, (b) some assisted skills, and (c) some other elements for representing mobile applications. Furthermore, the authors of the current work attached some reusable software modules to these approaches. In this manner, the participants of these experiments were able to automate the development of systems with the compared approaches.

The participants experienced the compared approaches and the current one, without knowing which one was under evaluation. For all the approaches, they received metamodels, transformations, mechanisms for code generation and certain reusable software modules. For all the approaches, they defined models, transformed these, generated some programming code, and implemented functional systems.

In order to alleviate the learning effect that affects subsequent implementations of the same specifications, the participant engineers were divided into four groups. Each group followed a different order for implementing the applications with the different approaches. Assuming that numbers one to three denote the aforementioned approaches and that letter "c" denotes the current approach, these groups followed respectively the orders (1, 2, 3, c), (2, 3, c, 1), (3, c, 1, 2) and (c, 1, 2, 3).

The current section presents the explanatory research. Concretely, subsection 5.1 describes the participants of this experiment including their previous expertise, and presents the comparison of development time. Subsection 5.2 compares the applications developed with each approach considering the requirements satisfied, the usability and the response time. Subsection 5.3 introduces the survey about the properties of the metamodel, and analyzes the responses of the participants after their experience.

5.1. Comparison of development time

In this experiment, 28 engineers expert in MDE, AOSE and/or health and working in different companies were invited to participate. All of them had in common to be students of the subject *Advanced Design of Software Architecture* in an official post-graduate master about *Software Architectures* in the modality of international distance education, in the Open University of Madrid. Regarding the participation, 24 experts actually performed the experiment out of the 28 experts, conforming a 85.71% of participation. Theses 24 experts experienced all the approaches, except one expert who experienced three out of the four approaches and another one who experienced two of these.

Table 1 details the previous professional experience of the experts that actually participated in the experiment. This table indicates the number of months of experience of each expert alongside the field(s) of their expertise. In this table and from this point forward, *App* denotes the development of applications for mobile devices (e.g. smartphones and tablets). The real names of the experts have been omitted here for the sake of the anonymity. As one can observe, all the experts have experience in MDE, AOSE and/or

Expert	Duration of experience (months)	Field(s) of expertise
Expert 1	38	MDE and Health
Expert 2	42	MDE (especially in MTs)
Expert 3	44	MDE and AOSE
Expert 4	62	Health and AOSE
Expert 5	25	MDE and Health
Expert 6	72	MDE and Business
Expert 7	30	AOSE
Expert 8	75	Health and App
Expert 9	37	Health and AOSE
Expert 10	41	MDE, Health and App
Expert 11	52	Health and App
Expert 12	57	MDE, Health and App
Expert 13	60	MDE, Health and App
Expert 14	29	AOSE and App
Expert 15	70	MDE
Expert 16	58	MDE (especially in metamodeling)
Expert 17	50	Health and AOSE
Expert 18	61	MDE and Health
Expert 19	71	MDE (especially in MTs)
Expert 20	31	AOSE and App
Expert 21	37	MDE and App
Expert 22	42	MDE (especially in metamodeling)
Expert 23	33	AOSE and App
Expert 24	55	MDE (especially in MT) and Health
Average	48.83	

Table 1: Previous professional experience of the experts and their fields of expertise

health, which are core in the current approach. In most cases, they have expertise in at least two fields considering these fields and App, which can be useful for providing applications to patients. The average amount of professional experience of the experts is 48.83 months (i.e. 4 years and 0.83 months). Hence, these experts have been considered appropriate for participating in this experiment due to their amount of experience and their fields of expertise.

The experts of this evaluation were from two different countries. Specifically, out of the 24 participants, 13 participants were from Spain and 11 participants were from Dominican Republic. In average, they were 33.25 years old. In addition, there were 22 men and two women.

There was a training phase for all the participants. In this phase, the participants were lectured about all the compared approaches including the current one. The participants developed simple application examples with all the approaches as part of the training phase. In addition, all the participants received some detailed information about PD including their symptoms, their stages, their social implications, their economical implications, the effects on their caregivers and so on. The participants were also encouraged to search more information about PD.

Afterwards, each of them selected a specific PD patient (real or imagined) and applied each of the compared approaches to assist this patient. In fact, each of these experts was asked to develop an application with the same specifications in all these approaches. After the experience, each expert provided a report indicating how they had applied each approach in the development of each application.

The development time was measured by each developer by noting when starting and finishing the work in this experience each day. Then, each developer calculated the number of development hours by summing the time spent in all days. Table 2 includes these development time values for all the participants for the development with each approach. In particular, the average development time was 32.79 hours with the Calvillo et al.'s approach, 30.18 hours with the approach of Lopez and Blovel, 29.96 hours with the Raghupathi and Umar's approach, and 23.83 hours with the current approach. This table also presents the reduction of development time of the current approach over each approach with percentages. In particular, the current approach reduces the 27.32% of the development time in comparison with the Calvillo et al.'s approach, 21.03% over the approach of Lopez and Blovel and 20.44% over the approach of Raghupathi and Umar.

Table 2 also presents the development time for obtaining the elements that were reused in each approach. In particular, this time includes the one for defining the metamodels and the model transformations. It also includes the time for developing the reusable software modules that can be included in the different approaches. It also considers the time for developing the generic agents and their connections. For each approach, this table contains the average total development time as the sum of (a) the average time of particular developments and (b) the development time of these reusable elements. This table also measures the reduction of time of the current approach over each one considering these total times. One can observe that the current approach reduces at least 22% of the total development time over all the analyzed approaches.

The reason of this decrease of development time may be the combination of MDE techniques with a well-supported AOSE methodology. On the one hand, the MDE techniques allowed developers to obtain MAS specification models quickly from the models customized for PD patients. On the other hand, from the MAS specification models, the Ingenias AOSE methodology and its tool support (i.e. the IDK) allowed developers to automatically generate MAS programming code skeletons that saved time in the software developments.

Expert	Development time (hours)			
	Calvillo et al.	**Lopez and Blovel**	**Raghu-pathi and Umar**	**Current approach**
Expert 1	40	42	37	31
Expert 2	38	37	39	30
Expert 3	46	37	35	29
Expert 4	25	22	24	19
Expert 5	45	32	34	25
Expert 6	44	39	41	32
Expert 7	20	20	19	14
Expert 8	25	25	21	15
Expert 9	35	33	31	25
Expert 10	20	20	20	18
Expert 11	36	29	33	25
Expert 12	31	31	29	22
Expert 13	30	32	28	23
Expert 14	21	25	19	14
Expert 15	43		40	33
Expert 16	30	34	29	25
Expert 17	38	36	37	30
Expert 18	45	40	35	29
Expert 19	37	34	32	28
Expert 20	25	24	26	18
Expert 21	26	19	30	24
Expert 22	35	32	31	22
Expert 23	20	21	19	15
Expert 24	32			26
Average	32.79	30.18	29.96	23.83
Time reduction of the current approach over each approach	27.32 %	21.03%	20.44%	
Development time of reusable components	89	98	92	71
Total time (i.e. sum of average time and reusable components time)	121.79	128.18	121.96	94.83
Time reduction of the current approach over each approach considering the total times	22.13%	26.02%	22.24%	

Table 2: Development time of all participants for constructing applications with each approach

Id.	Requirement
1	Coordination with caregivers for dressing in the morning and night
2	Coordination with caregivers for having a shower in the morning or night
3	Coordination with caregivers for cooking the meals
4	Coordination with caregivers for eating the meals
5	Coordination with caregivers for going out
6	Automatic speech recognition for a patient that cannot easily write on the device
7	Automatic reading for a patient that cannot easily read on the device
8	Treatment of depression
9	Treatment of sleeping problem
10	Treatment of anxiety

Table 3: Requirements with their identifiers

It is worth mentioning that the availability of these participant experts was probably due to the fact that this experience was immerse in a post-graduate official master that was valuable for them. The experience was initially planned to be performed in a range up to 120 days, but some participants needed more time to complete the experience. In particular, 16 participants accomplished the experience in the range up to 120 days, while six participants needed a range up to 244 days and two participants needed up to 304 days.

In conclusion, the current approach reduces the development time of applications for PD patients in at least 20% of the time in comparison to all the evaluated approaches. If the comparison also considers the time for developing reusable elements, the reduction is at least 22% over all the approaches.

5.2. Comparison of the developed applications

This section evaluates and compares the applications that were developed with the different approaches as described in the previous section. Some of the applications were able to be tested by real PD patients. This testing phase was only performed by the experts that actually had a relative or a friend that suffered PD, or were able to contact someone with this disease. In particular, the applications of 13 experts were actually tested by real PD patients. Each application was tested by the PD patient for who it was customized.

Each expert initially asked a patient about their needs, in order to determine the requirements of the applications for this patient. After the development and the testing phase, each patient was asked whether each requirement was properly satisfied by each application. Table 3 presents a list of requirements with their identifiers. Some of these requirements refer to specific functionalities, such as the coordination of certain daily tasks and the treatment of certain psychological symptoms. Others requirements refer to

Developer	Requirements for each patient	Calvillo et al.	Lopez and Blovel	Raghupathi and Umar	Presented Metamodel
Expert 1	1; 5	1; 5	1;5	1; 5	1;5
Expert 3	5; 6; 7	5; 6	5; 6; 7	5	5; 7
Expert 4	4; 8	4; 8	4; 8	4; 8	4; 8
Expert 6	3; 4; 7	3; 4	3; 4; 7	3; 4; 7	3; 4; 7
Expert 8	1; 6; 9	1; 6	1; 6	1; 6	1; 6
Expert 9	2; 3; 4	2; 3; 4	2; 3; 4	2; 3; 4	2; 3; 4
Expert 11	1; 8; 9; 10	1; 8	1; 8	1	1; 8; 9; 10
Expert 14	6; 7; 8	6; 7	6; 7	6; 7; 8	6; 7; 8
Expert 17	5; 7; 9	5; 7	5; 7	5; 7	5; 7
Expert 19	2; 3; 4	2; 3; 4	2; 3; 4	2; 3; 4	2; 3; 4
Expert 20	5; 6; 8	5; 6	5; 6	5; 6; 8	5; 6; 8
Expert 21	1; 7; 9; 10	1; 7	1; 7	1; 7	1; 7; 9
Expert 23	1;7	1; 7	1;7	1; 7	1;7
Average percentages of satisfied requirements		76.92%	82.05%	80.13%	95.51%
Averages of usability assessments		4.31	4.46	4.62	5.46

Table 4: Requirements satisfied by each application according to the experience of PD patients, and their average usability assessments with a seven-point Likert scale.

accessibility properties such as automatic speech recognition and automatic reading. Table 4 determines the requirements that were initially established for each patient, by indicating their identifiers. This table also indicates the requirements that were actually satisfied according to the experience of the patients, for the applications developed with each approach. As one can observe, the current approach has the highest average percentage of requirements satisfied according to the patients' experience in comparison to the other analyzed approaches.

Moreover, the PD patients were asked about the usability of the applications with the question "Do you find this application usable?" This question was answered with a seven-point Likert scale. Table 4 also presents the averages of the corresponding responses for each approach. One can observe that the PD patients considered that the applications developed with the current approach are more usable than the others. The users were also requested to provide commentaries about the usability of all the applications. Ten out of 13 users provided these commentaries. In the applications of the current approach, they mainly assessed positively the following aspects in comparison to the other approaches: (1) the large font size of the interface buttons, (2) the speech recognition for avoiding writing with hand shaking, and (3) the ease for finding the necessary functionalities in the interface.

As another property of the applications quality, this work measures the average response time per operation. In each application, all the operations were performed at least 20 times each one measuring the response time, and the average response time was calculated for each application. Table 5

Feature	Calvillo et al.	Lopez and Blovel	Raghupathi and Umar	Presented Metamodel
Average response time (ms)	1173.50	985.63	881.33	770.46
Average marks about proper use of models (out of ten)	9.29	9.42	9.38	9.67

Table 5: Average response times of applications and marks about their proper use of models

presents the averages of these response times for all the applications developed with each approach expressed with milliseconds. This was measured for the applications of the 24 experts, since this measurement did not require real PD patients for being performed. One can observe that the applications with the current approach have the lowest response time in average (i.e. 770 ms). It has a reduction of at least 12.58% of the response time per operation over all the other approaches in average.

The coordination mechanisms between the patient and the caregivers differ from the current approach to the other ones. The coordination usually involves high time-consuming operations since it requires communications between devices. In the current approach, the software about the coordination is automatically generated. Its high performance may rely on the fact that the patient agent keeps a local copy of the two kinds of timetables (i.e. for routine activities and for the emergency activities) of all their caregivers. Thus, when a patient requests assistance for some activity out of their daily routine, their agent can consult these timetables locally. In this manner, it knows which caregivers are normally available and which ones are available for emergencies in the specific hour and day of the week. Therefore, in most cases the patient agent only needs to communicate with one caregiver agent for getting help. It could be possible that the caregiver agent did not accept the request and the patient agent needed to contact other agents, but this hardly happens. Since the other approaches do not generate the coordination mechanism, the participants implemented different coordination mechanisms in the applications with other approaches. The coordination mechanisms may be the reason why the current approach improves the average response time over the alternatives.

Furthermore, the instructor read the reports about all the applications (i.e. of the 24 experts), and inspected their models conforming to the metamodels and their programming code. He evaluated the grade in which experts used the models correctly for each approach with marks out of ten. Table

		Specification kind and response times	Specification kind and proper use of models
Calvillo et al.	Pearson Correlation	0.113	-0.296
	Sig. (2-tailed)	0.600	0.161
Lopez and Blovel	Pearson Correlation	-0.012	-0.185
	Sig. (2-tailed)	0.957	0.387
Raghupathi and Umar	Pearson Correlation	-0.087	-0.298
	Sig. (2-tailed)	0.685	0.157
Current approach	Pearson Correlation	0.296	0.059
	Sig. (2-tailed)	0.161	0.784

Table 6: Analysis of correlations (a) between the specification kind and the response times, and (b) between the specification kind and the proper use of models.

5 shows the average marks for each approach. Developers properly used the models in the applications of all the approaches with average marks above nine for all the approaches.

This work performs a correlation analysis for determining whether the aforementioned features were influenced by the kind of specifications. This work considers two specification kinds, which are (1) the specifications based on real patients and (2) the ones based on imagined patients. In particular, this analysis considers the correlations (a) between the specification kind and the response times, and (b) between the specification kind and the proper use of models. Since there is a dichotomous variable in each correlation, this work applies the Point-Biserial Correlation analysis [27]. The dichotomous variable is the specification kind, in which the one and zero values represent respectively the mentioned kinds. The Point-Biserial Correlation analysis has been performed as a particular case of the Pearson's correlation analysis, and Table 6 presents the results. The correlations have been analyzed for all the approaches, including the current one. Considering a significance level of 0.05, one can observe that there is not any correlation between the specification kind and the response times of applications in any of the approaches. Likewise, there is not any significant correlation between the specification kind and the proper use of models in any of the approaches.

5.3. Evaluation trough a survey

The participants evaluated the compared approaches through a survey after the developments mentioned in section 5.1. The participants had the previous expertise introduced in that section (i.e. in Table 1). In addition, all the participants were informed of the particularities of PD. Furthermore, they had acquired experience in the compared approaches supported with their reports and measured with the time dedicated to each approach (previously introduced in Table 2). Specifically, the participants spent 116.76

Id.	Question
Q1	Is this metamodel useful for developing computer applications that assist PD patients?
Q2	Is this metamodel appropriate for developing computer applications that assist patients in general?
Q3	Does this metamodel define all the skills that are assisted for PD patients?
Q4	Is this metamodel better than other existing metamodels for modeling PD patients?
Q5	Does this metamodel allow one to model all the necessary information for customizing the computer applications that assist PD patients?
Q6	Does this metamodel cover the needs for modeling PD patients?
Q7	Does this metamodel properly define the human caregivers of PD patients?
Q8	Can the computer applications developed with this metamodel cover the needs of PD patients?
Q9	Can this metamodel define the social-economical aspects of PD Patients properly?
Q10	Can this metamodel define all the symptoms of PD patients?
Q11	Make any comment about this metamodel that you find suitable. Indicate whether you think that this metamodel can improve and how you would do it if so. In the previous questions that obtained low marks and you want, mention what you missed in the metamodel.

Table 7: Questions of the survey

hours in average to experience all the compared approaches. Therefore, the perceptions of these experts have been considered useful for completing the evaluation of the current approach. In order to avoid indirect conditioning, all the approaches were presented in a similar way to the experts without mentioning which one was under evaluation. The authors of the approaches were neither mentioned to them for the same reason.

The questions of the survey are presented in Table 7. Among other aspects, the goal of this survey is to determine whether the metamodel of each approach is suitable for modeling PD patients in order to develop AAL systems for these. In particular, some of the questions refer to the utility of the metamodel for developing AAL applications for patients, either suffering PD in Q1 question or suffering any disease in Q2 question. The survey also asks about the completeness of the metamodel for representing PD patients in general in Q4 question, and specifically considering either their skills in Q3 question, their needs in Q6 question or their symptoms in Q10 question. Some questions refer to the environment of PD patients, such as their caregivers in Q7 and their social-economical aspects in Q9. Finally, some questions relate to the obtained computer applications, considering their customization in Q5 and their support for the needs of PD patients in Q8.

All the questions from Q1 to Q10 are replied with a seven-point Likert scale, and Table 8 shows the answer alternatives for this scale. The Q11 question is asked to be replied by writing some comments without restrictions, to obtain all the remaining possible feedback from experts. All these questions are formulated for the metamodel of each approach in order to compare the results of the presented metamodel with the most similar ones in the literature.

Answer	Value
Strongly disagree	1
Disagree	2
Disagree somewhat	3
Neutral	4
Agree somewhat	5
Agree	6
Strongly agree	7

Table 8: Seven-point Likert scale for replying questions Q1-Q10

	Calvillo et al.		Lopez and Blobel		Raghupathi and Umar		Presented metamodel		Friedman test		
Question	Mean	Median	Mean	Median	Mean	Median	Mean	Median	Asymp. Sig.	Decision	χ^2(df)
Q1	4.091	4.500	4.950	5.000	4.636	5.000	6.208	6.500	0.000	Reject the null hypothesis	19.052(3)
Q2	4.364	5.000	5.250	6.000	5.136	6.000	5.500	6.000	0.181	Retain the null hypothesis	4.879(3)
Q3	2.455	2.000	3.300	2.500	2.546	2.000	4.750	5.000	0.000	Reject the null hypothesis	25.485(3)
Q4	2.546	2.000	3.300	3.000	3.227	3.000	4.583	4.000	0.000	Reject the null hypothesis	21.565(3)
Q5	3.136	2.500	3.900	4.000	3.409	3.500	5.042	5.500	0.000	Reject the null hypothesis	25.395(3)
Q6	2.727	2.000	3.500	3.000	3.227	3.000	5.208	5.000	0.000	Reject the null hypothesis	25.582(3)
Q7	2.546	2.000	3.050	3.000	2.182	2.000	5.042	5.500	0.000	Reject the null hypothesis	35.516(3)
Q8	3.546	3.500	4.050	4.500	3.682	3.500	5.500	6.000	0.000	Reject the null hypothesis	25.897(3)
Q9	2.091	2.000	3.050	2.000	2.909	2.000	4.625	5.000	0.000	Reject the null hypothesis	20.598(3)
Q10	2.955	3.000	3.350	3.000	3.955	4.000	5.417	6.000	0.000	Reject the null hypothesis	24.579(3)

Table 9: Means and Medians of questions Q1-Q10 for the four metamodels, and the results of the Friedman test with a significance level of 0.05

Table 9 shows the means and medians of the results of questions Q1-Q10 for the evaluated metamodels. As one can observe, the presented metamodel obtains higher means than all the other evaluated metamodels for all the questions. In addition, the presented metamodel has a greater median than all the other metamodels for all the questions except for question Q2.

In order to determine whether the aforementioned differences of means and medians are statistically significant, this work has applied the Friedman test. The results of this test are also presented in Table 9. One can observe that all the questions present significant differences except Q2 question. In this test, when a question presents significant differences, it is represented with the phrase "Reject the null hypothesis" in the decision column. The asymptotic significance (Asymp. Sig.) column represents the probability that an equal or greater difference occurs randomly. When the probability is below 0.05, the differences are considered to be statistically significant. The lower Asymp. Sig. value is, the more significant the difference is. In

	Calvillo et al. vs the current one			**Lopez and Blobel vs the current one**			**Raghupathi and Umar vs the current one**		
Question	**Asymp. Sig. (2-tailed)**	**Decision**	**Diff. of Means**	**Asymp. Sig. (2-tailed)**	**Decision**	**Diff. of Means**	**Asymp. Sig. (2-tailed)**	**Decision**	**Diff. of Means**
Q1	0.000	Reject the null hypothesis	2.117	0.001	Reject the null hypothesis	1.258	0.001	Reject the null hypothesis	1.572
Q2	0.051	Retain the null hypothesis	1.136	0.638	Retain the null hypothesis	0.250	0.573	Retain the null hypothesis	0.364
Q3	0.000	Reject the null hypothesis	2.296	0.026	Reject the null hypothesis	1.450	0.000	Reject the null hypothesis	2.205
Q4	0.000	Reject the null hypothesis	2.038	0.002	Reject the null hypothesis	1.283	0.009	Reject the null hypothesis	1.356
Q5	0.000	Reject the null hypothesis	1.905	0.007	Reject the null hypothesis	1.142	0.004	Reject the null hypothesis	1.633
Q6	0.000	Reject the null hypothesis	2.481	0.001	Reject the null hypothesis	1.708	0.002	Reject the null hypothesis	1.981
Q7	0.000	Reject the null hypothesis	2.496	0.000	Reject the null hypothesis	1.992	0.000	Reject the null hypothesis	2.860
Q8	0.001	Reject the null hypothesis	1.955	0.001	Reject the null hypothesis	1.450	0.000	Reject the null hypothesis	1.818
Q9	0.000	Reject the null hypothesis	2.534	0.005	Reject the null hypothesis	1.575	0.007	Reject the null hypothesis	1.716
Q10	0.000	Reject the null hypothesis	2.462	0.001	Reject the null hypothesis	2.067	0.003	Reject the null hypothesis	1.462

Table 10: Results of the Wilcoxon Signed Ranks test alongside the differences of means. The significance level is 0.05.

particular, the differences are very significant as Asymp. Sig. is 0.000 for all questions except for Q2 question. The chi-square value (also denoted as χ^2) indicates how different the values are. The greater chi-square is, the larger the difference is. The difference does not only depends of chi-square, but also of the degrees of freedom (denoted as *df* in the table). The degrees of freedom are three for all the questions. Hence, chi-square determines how much different the responses are for each question. One can observe that the difference is considerably greater in Q7 question ($\chi^2 = 35.516$) than in the other questions ($\chi^2 \leq 26.000$). Thus, the greatest improvement of the current approach is related to the definition of human caregivers for PD patients according to the chi-square value of the Friedman test.

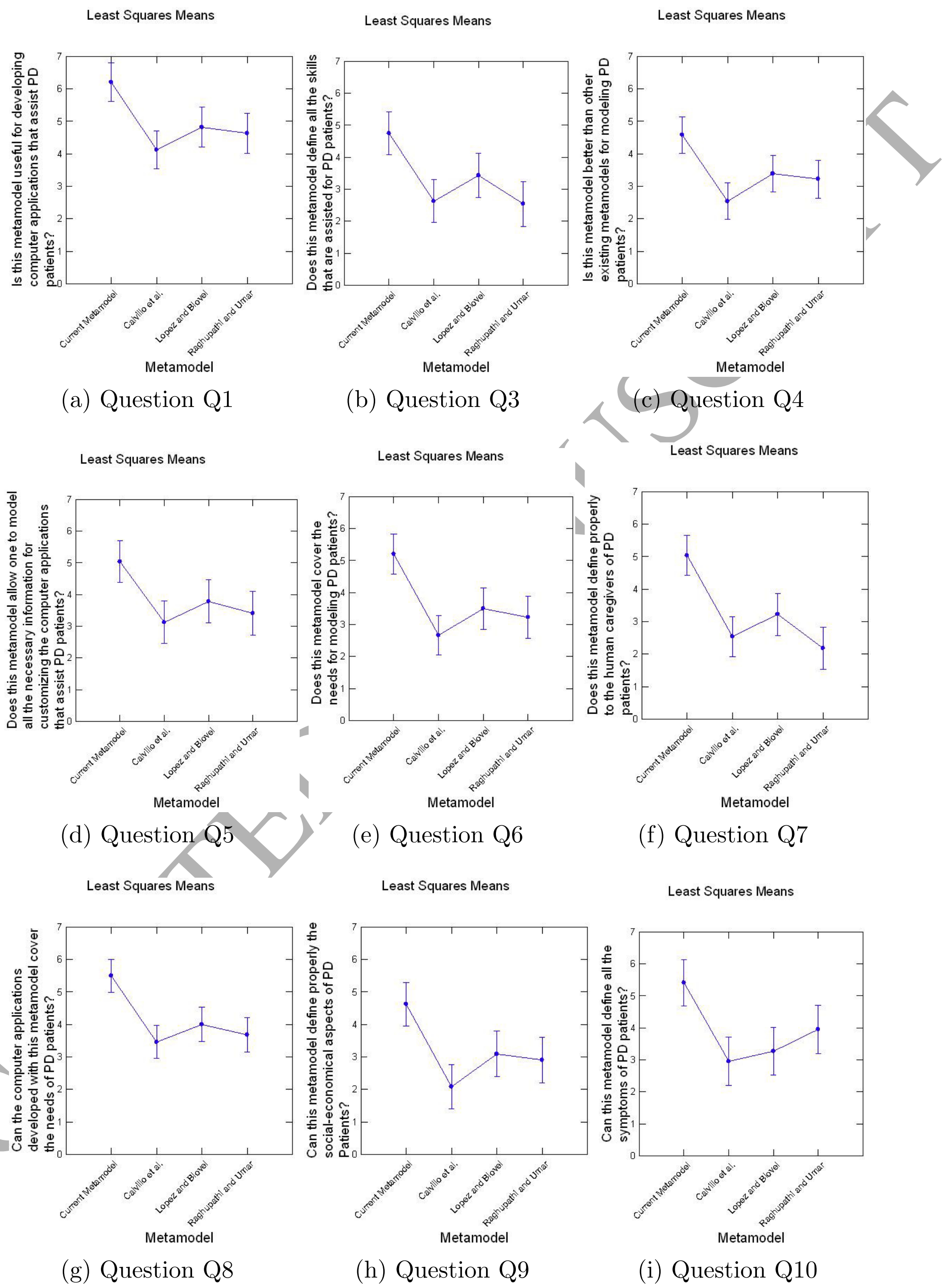
Moreover, the current work applies the Wilcoxon Signed Ranks test, for determining the significance of the difference between the presented metamodel and each of the other metamodels. Table 10 presents the results of this test alongside the differences of means. One can observe that the current metamodel improves each evaluated metamodel with statistical significance in all questions except Q2, according to this test. These results coincide with the previous test. The Asymp. Sig. column has a similar meaning as in the previous test, but it only refers to pairs of metamodels (the current one and each of the others). Discarding Q2, the significance of differences is greater with Calvillo et al. (in which 0.001 is the maximum value of Asymp. Sig.) than with Raghupathi and Umar (in which 0.009 is the maximum value of Asymp. Sig.) and Lopez and Blobel (in which 0.026 is the maximum value of Asymp. Sig.).

The difference of means of Table 10 measures the gap between each of the similar metamodels and the current one for each question. Figure 11 graphically presents these differences alongside the standard deviations for each of the questions that obtained a statistical significant difference. These differences represent scores in the seven-point Likert scale. Considering the means and the standard deviations of the graphs of this figure and the data of this table, the greatest improvements of the current approach over the existing ones are related to questions Q3, Q6, Q7 and Q9. In other words, the most valuable advances of the current approach over the literature concern (a) the modeling of the skills assisted for PD patients, (b) the support for their needs, (c) the definition of their human caregivers, and (d) the consideration of their social-economical circumstances as relevant information in the development of AAL systems.

It is worth mentioning that question Q2 asks about appropriateness of the current approach for developing applications for patients in general, instead of specifically PD patients. As the current approach is mainly focused on PD patients, it is reasonable that the current approach does not improve the others in this issue.

Considering all the questions with statistically significant improvement, the current metamodeling approach improves the others approaches in all the evaluated aspects specifically related to PD patients. It is more useful for developing computer applications that assist them. It better defines their skills that need to be assisted. It further determines the information for customizing the applications that assist them. It further covers the needs for modeling them. It defines more properly their human caregivers. The computer applications developed with the current approach can better cover their needs. It defines their social economical aspects more accurately. It is considered to better determine all their common symptoms.

Regarding Q11 question, experts provided useful and varied feedback that is planned to guide our future work for improving the presented metamodel. The feedback indicated some symptoms that can appear in PD patients, such as incontinence and a psychotic outbreak. The experts also detected the absence of some skills that PD patients may need assistance, such as having a bath. They also advised to include more information about human



(a) Question Q1 (b) Question Q3 (c) Question Q4

(d) Question Q5 (e) Question Q6 (f) Question Q7

(g) Question Q8 (h) Question Q9 (i) Question Q10

Figure 11: Graphical comparisons of means and standard deviations

caregivers such as their numbers of dependent children for considering these in the distribution of workloads.

In conclusion, the evaluation of the current metamodel shows that it further determines the features of PD patients than other existing similar metamodels. The current metamodel can also further assist the development of AAL applications for PD patients than others. This evaluation has been performed with 24 experts with professional experience in MDE, AOSE and/or health for more than four years in average. This evaluation was performed after each participant had spent 116 hours in average applying the compared approaches.

6. Conclusions and future work

A MDE approach has been proposed for constructing AAL MASs for assisting PD patients in their lives. The first step is the definition of a model of a PD patient with their needs and circumstances, by means of the presented metamodel. Then, the proposed MTs are applied to obtain an initial MAS design model customized for the particularities of the patient. At last, the Ingenias methodology guides the process for refining the initial model design and generating a functional MAS. This approach is exemplified with a case study in order to show its usefulness and present its practical application. This approach has been evaluated by 24 experts from two different countries, who developed applications with the current approach and other three similar ones. PD patients tested the applications of 13 of these experts. The results showed that the current approach improves (1) the development time, (2) the percentage of requirements satisfied for PD patients according to them, (3) the usability of applications, (4) the response time of applications, and (5) the underlying metamodel for the current purpose according to the experts.

The current approach is planned to be enhanced in several ways. For now, the proposed MTs only generate modeling elements for agents, tasks, goals and interaction units. In the future, new MTs will be added to generate other modeling elements such as testing units and mental states. In addition, the metamodel is planned to be improved by considering the constructive feedback of the experts in the evaluation. The metamodel will include new elements for representing more features and circumstances of PD patients and their caregivers. Moreover, more MASs will be developed with this approach for more real PD patients with different circumstances. In this manner, the current approach will be adapted for a wider range of cases. Furthermore, the presented metamodel will be mapped to the most common health standards, such as the ones proposed by the Health Level Seven and the Integrating the Healthcare Enterprise organizations. In this way, the current approach will improve its interoperability with the existing medical systems. Finally, the current work will analyze a wide range of user interface kinds for disable and elder people for selecting one. Then, the current approach will be extended to develop applications for the patients in advanced stages of PD with an appropriate user interface.

Acknowledgments

This work has been done in the context of the project "Social Ambient Assisting Living - Methods", supported by Spanish Ministry for Economy and Competitiveness, with grant TIN2011-28335-C02-01. We also acknowledge the "Fondo Social Europeo" and the "Departamento de Industria e Innovación del Gobierno de Aragón" for their support.

References

[1] Benhajji, N., Roy, D., Anciaux, D., 2015. MAS-ML 2.0: Supporting the modelling of multi-agent systems with different agent architectures. IFAC-PapersOnLine 48 (3), 710–714.

[2] Beydoun, G., Low, G., Mouratidis, H., Henderson-Sellers, B., 2009. A security-aware metamodel for multi-agent systems (MAS). Information and software technology 51 (5), 832–845.

[3] Calvillo, J., Román, I., Roa, L., 2013. Empowering citizens with access control mechanisms to their personal health resources. International Journal of Medical Informatics 82 (1), 58–72.

[4] Fouquet, F., Mar. 2013. Kevoree: Model@Runtime for continuous development of heterogeneous distributed adaptive systems. Ph.D. thesis, Université Rennes 1.
URL `https://tel.archives-ouvertes.fr/tel-00831018 (last accessed 09/09/2015)`

[5] Gamez, N., Fuentes, L., 2013. Architectural evolution of FamiWare using cardinality-based feature models. Information and Software Technology 55 (3), 563–580.

[6] García-Magariño, I., Jun. 2010. A framework for the definition and transformation of models in multi-agent systems. Ph.D. thesis, Universidad Complutense de Madrid.
URL `http://eprints.ucm.es/10639/ (last accessed 09/09/2015)`

[7] García-Magariño, I., 2013. Defining and Transforming Models of Parkinson Patients in the Development of Assisted-Living Multi-agent Systems with INGENIAS. In: Highlights on Practical Applications of Agents and Multi-Agent Systems. Vol. 365 of Communications in Computer and Information Science. Springer, pp. 460–471.

[8] García-Magariño, I., Palacios-Navarro, G., 2015. A technique for metamodeling diagram types with tool support. Arabian Journal for Science and Engineering, DOI: 10.1007/s13369–015–1607–z.

[9] Gascueña, J., Navarro, E., Fernández-Caballero, A., 2012. Model-driven engineering techniques for the development of multi-agent systems. Engineering Applications of Artificial Intelligence 25 (1), 159–173.

[10] Ghorbani, A., Dijkema, G., Bots, P., Alderwereld, H., Dignum, V., 2014. Model-driven agent-based simulation: Procedural semantics of a maia model. Simulation Modelling Practice and Theory 49, 27–40.

[11] Gómez-Garre, P., Jesús, S., Carrillo, F., Cáceres-Redondo, M. T., Huertas-Fernández, I., Bernal-Bernal, I., Bonilla-Toribio, M., Vargas-González, L., Carballo, M., Mir, P., 2014. Systematic mutational analysis of FBXO7 in a Parkinson's disease population from southern Spain. Neurobiology of Aging 35 (3), 727.e5–727.e7.

[12] Gomez-Sanz, J. J., Fernández, C. R., Pavón, J., 2009. Quick Prototyping and Simulation with the INGENIAS Agent Framework. In: Proceedings of the Second Multi-Agent Logics, Languages, and Organisations Federated Workshops. Vol. 494. p. 46.

[13] Gonçalves, E. J. T., Cortés, M. I., Campos, G. A. L., Lopes, Y. S., Freire, E. S., da Silva, V. T., de Oliveira, K. S. F., de Oliveira, M. A., 2015. Mas-ml 2.0: Supporting the modelling of multi-agent systems with different agent architectures. Journal of Systems and Software 108, 77–109.

[14] Jenkinson, C., Dummett, S., Kelly, L., Peters, M., Dawson, J., Morley, D., Fitzpatrick, R., 2012. The development and validation of a quality of life measure for the carers of people with Parkinson's disease (the PDQ-Carer). Parkinsonism & Related Disorders 18 (5), 483–487.

[15] Jouault, F., Allilaire, F., Bézivin, J., Kurtev, I., Valduriez, P., 2006. Atl: a qvt-like transformation language. In: Companion to the 21st ACM SIGPLAN symposium on Object-oriented programming systems, languages, and applications. ACM, pp. 719–720.

[16] Kaluža, B., Mirchevska, V., Dovgan, E., Luštrek, M., Gams, M., 2010. An agent-based approach to care in independent living. In: Ambient Intelligence. Vol. 6439 of Lecture Notes in Computer Science. Springer, pp. 177–186.

[17] Lopez, D., Blobel, B., 2009. A development framework for semantically interoperable health information systems. International journal of medical informatics 78 (2), 83–103.

[18] Miyake, Y., Tsuboi, Y., Koyanagi, M., Fujimoto, T., Shirasawa, S., Kiyohara, C., Tanaka, K., Fukushima, W., Sasaki, S., Yamada, T., et al., 2010. LRRK2 Gly2385Arg polymorphism, cigarette smoking, and risk of sporadic Parkinson's disease: A case-control study in Japan. Journal of the neurological sciences 297 (1-2), 15–18.

[19] Nain, G., Barais, O., Fleurquin, R., Jézéquel, J.-M., 2009. EntiMid : un middleware aux services de la maison. In: 3ème Conférence Francophone sur les Architectures Logicielles (CAL'O9). Nancy, France, France, pp. 59–72.

[20] Nefti, S., Manzoor, U., Manzoor, S., 2010. Cognitive agent based intelligent warning system to monitor patients suffering from dementia using ambient assisted living. In: Information Society (i-Society), 2010 International Conference on. IEEE, pp. 92–97.

[21] Ni, Y., Wang, Y., 2015. Development of an agent-based collaborative production system based on real-time order-driven approach. Arabian Journal for Science and Engineering 40 (4), 1239–1253.

[22] Pavón, J., Gómez-Sanz, J., 2003. Agent oriented software engineering with INGENIAS. In: Multi-Agent Systems and Applications III. Vol. 2691 of Lecture Notes in Computer Science. Springer, pp. 394–403.

[23] Raghupathi, W., Umar, A., 2008. Exploring a model-driven architecture (MDA) approach to health care information systems development. International journal of medical informatics 77 (5), 305–314.

[24] Ruiz-López, T., Noguera, M., Rodríguez, M. J., Garrido, J. L., Chung, L., 2013. REUBI: A requirements engineering method for ubiquitous systems. Science of Computer Programming 78 (10), 1895–1911.

[25] Silva, D., Braga, R., Reis, L., Oliveira, E., 2012. Designing a meta-model for a generic robotic agent system using Gaia methodology. Information Sciences 195, 190–210.

[26] Su, C., 2008. Mobile multi-agent based, distributed information platform (MADIP) for wide-area e-health monitoring. Computers in industry 59 (1), 55–68.

[27] Wherry, R. J., 1984. Inferred Correlations and Reliability Measures. In: Contributions to correlational analysis. Orlando: Academic Press, Elsevier, pp. 45–62.

[28] Wohlin, C., Runeson, P., Höst, M., Ohlsson, M., Regnell, B., Wesslén, A., 2000. Experimentation in software engineering: an introduction. Kluwer Academic Publishers.

Biography

Iván García-Magariño was born in Madrid, Spain, in 1982. He obtained the Ph.D. in Computer Science Engineering in the Complutense University of Madrid, in 2009. He had a FPI researcher scholarship from 2006 to 2010. He worked as lectured in Madrid Open University from 2010 to 2014. He is currently lecturer at the University of Zaragoza from 2014. Among journals, book chapters, conferences and workshops, he has over sixty publications. His most relevant publications belong to international journals with a high impact, such as "Engineering Applications of Artificial Intelligence", "Expert Systems with Applications", "Knowledge-based Systems" and "Information and Software Technology".

Guillermo Palacios-Navarro was born in Zaragoza (Spain) in 1972. He received a M.Sc. degree in Telecommunication Engineering and a Ph.D. degree in Electronic Engineering from the University of Zaragoza, Spain, in 1996 and 2009, respectively. Since 1998, he works in the Department of Electronic Engineering and Communications, University of Zaragoza, where he is currently an Associate Professor. His current research interests include virtual reality, augmented reality and virtual rehabilitation techniques applied to the recovery of neurodegenerative disorders, human computer interaction, image processing and biomedical engineering.