Embedded Systems for People with Special Needs: Insights from a Real Case

Elena Gómez-Martínez and Álvaro Fernández-Díaz
Babel Group, DLSIIS, Escuela Técnica Superior de Ingeniería Informática
Universidad Politécnica de Madrid, Spain
Email: {egomez, avalor}@babel.ls.fi.upm.es

***Abstract*—In this paper we report on some insights derived from our experience in the development of an embedded system. A real-world embedded system implementing a smart home is used as an example. The target audience of this system includes people with special needs, such as impaired or elderly people. A description of the user testing phase is also provided. A series of difficulties related to the accessibility to the system for people with special needs were identified. These difficulties suggest the necessity to enforce the training on accessibility techniques for software engineers.**

I. Introduction

Software engineers acquire several skills on technical aspects, such as programming, computational complexity and operating systems, among others. Concerning embedded and real-time systems, they mainly learn the importance of building safe, secure and robust software systems. Nevertheless, some embedded systems need to cover other aspects, such as performance, maintainability and/or dependability. In the case of embedded systems with Human-Computer Interaction (HCI), for instance one deployed at a smart home, accessibility and usability are mandatory. The user experience is one of the key requirements, particulary if the embedded system will be used by people with special needs. However, the education and training on these disciplines, that is, accessibility and usability, is usually during the software engineering studies.

In industrial societies, people massively use electronic devices in everyday life: mobile phones, TV sets or washing machines are just some of several possible examples. Within these societies it exists a considerable number of people with functional diversity (including those with aging-related conditions). There is also a growing gap between their abilities and their access to digital information technologies: the digital divide. The user interfaces of the aforementioned electronic devices are not generally designed considering neither their special needs nor the so-called *Design for all* principles [1]. In spite of this fact, the majority of the European countries own more mobile subscriptions than inhabitants, as Eurostat office [2] reports. Internal studies carried out by ONCE[1] foundation suggest similar trends for people with disabilities. According to the United Nations [3], around 15 per cent of the total world's population is affected by some disability. To address this situation, many governments and organizations carry out e-inclusion initiatives that promote activities aimed at achieving an inclusive information society.

[1]Organización Nacional de Ciegos Españoles (National Organization of blind people in Spain)

This paper describes our experiences obtained while testing an interoperable architecture with adaptative interfaces for people with special needs. This general purpose architecture was deployed as an embedded system in a smart home [4], [5], as well as within other environments, covering a wide range of real world scenarios. These scenarios include leisure services (location and purchase of tickets for events), urban networking [6], social networks [7], eGoverment [8] and banking services (ATMs) [9]. The interoperable architecture is assessed in [10] and detailed in [11].

The paper is organized as follows. Firstly, Section II outlines the aforementioned interoperable architecture and its use context. Then, Section III describes the user testing phase. Section IV discusses the lessons learnt from the point of view of the engineering students' education and training. Finally, Section V includes the conclusions.

II. Case Study: A Smart Home

Universal Access continues being a critical quality target for Information and Communications Technology (ICT), as [12] stated, especially in industrial societies where there is a growing number of people with functional diversity, including those with aging-related conditions. Indeed, ICTs may require particular skills and abilities to interact with platforms, the plethora of wireless communication systems and smart devices such as kiosks or ATMs. The inexistence of these skills and abilities extends in some cases the traditional concept of disabled people towards people with functional diversity or special needs. The growing gap between their abilities and access to ICT is called the *digital divide*. Interoperable software architectures that support universal designed user interfaces and Assistive Software are two approaches to bridge this gap, e.g., [13]–[15].

The INREDIS project (INterfaces for RElations between Environment and people with DISabilities) [16] aimed to develop environments that enable the creation of communication and interaction channels between people with some kind of special need and their context, where the targets are a set of auto-discoverable devices.

The project was structured into nine main activities or workpackages (WPs) that covered the main aspects to achieve a leap forward in the design of accessible and interoperable technology. Among the most relevant activities were:

- Detection of the needs in the use of technologies among people with functional diversity. Around a

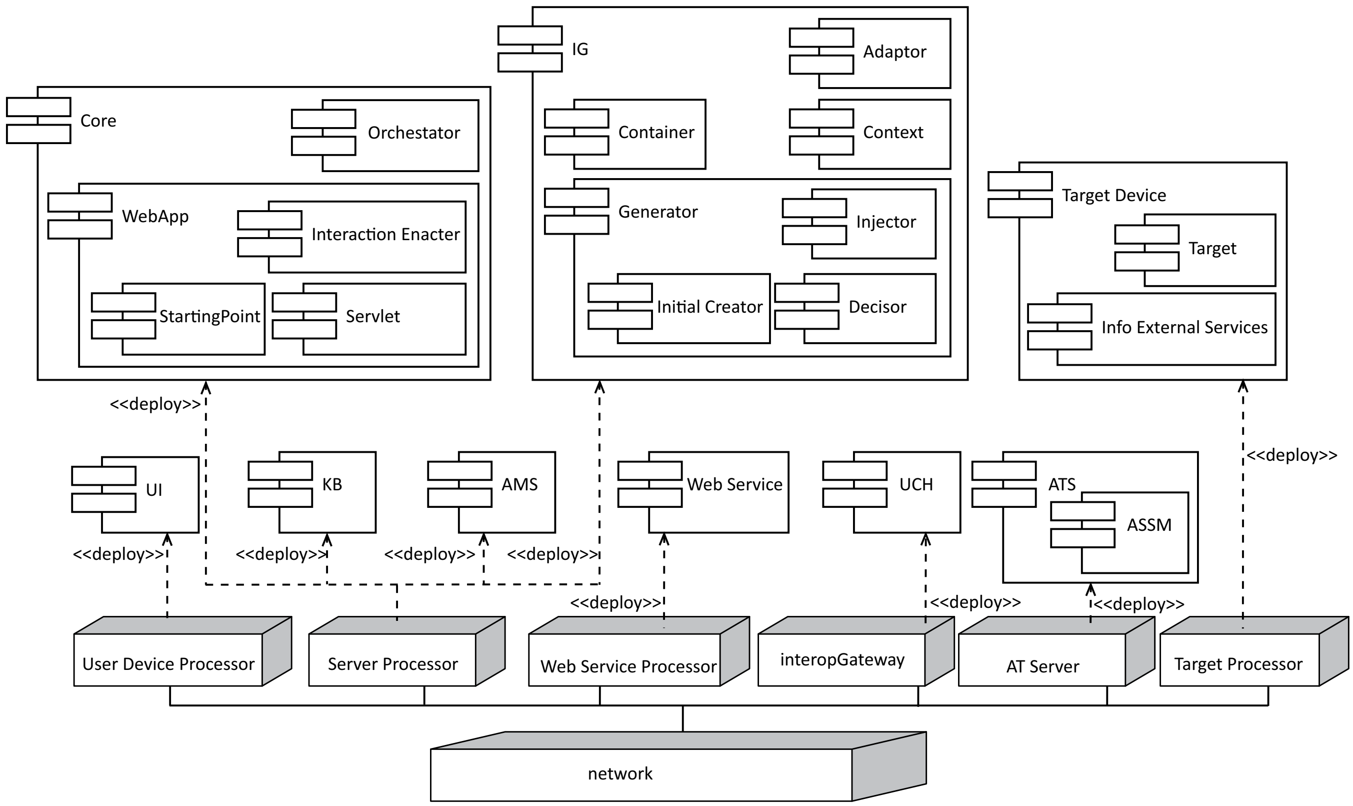

Figure 1. UML deployment diagram of the Interoperable Architecture [10].

thousand of interviews of impaired and elderly people were carried out. The collected information was systematized to extract users' requirements and preferences based on their disability.

- Study of different human-machine interaction technologies, such as haptic devices, smart fabrics or sensors networks, among others, and their potential of adaptation to people with special needs.
- Integration of users with special needs in the technological environment developed during the project, including lifecycle software development, by means of a user-centered design, and an extensive user testing phase.

In the context of this project, an interoperable architecture, capable of adapting different types of interfaces to users' needs and preferences, was designed and developed.

The INREDIS architecture was conceived as a universal solution capable to provide disabled and elderly people with accessible and personalized interfaces according to their preferences and needs. Consequently, the architecture was designed for a general purpose context of use. Nevertheless, some running prototypes were built for different environments, covering a wide range of real world scenarios, among them: leisure services (location and purchase tickets for events), smart home [4], urban networking [6], social networks [7], eGoverment [8] and banking services (ATMs) [9]. Although the architecture was envisaged for impaired users, "any" user, i.e. user without disabilities, might exploit and obtain benefits when using the system (e.g., by using their mobiles as universal remote controllers in the smart environment). The architecture is partially described in [10], [17] and detailed in [11]. In this paper, we only focus on this embedded architecture deployed in a home environment [4].

A smart home is a set of intelligent appliances that provide a better home life experience to its occupants without overpowering them with complex technologies or nonintuitive user interfaces [18]. Therefore, it is an embedded system in which it is necessary to take into account the user interaction with the different devices both directly and through a user interface.

The interoperable architecture is an event-driven and service-oriented architecture that further develops the idea of Universal Control Hub (UCH) [19]. The rationale behind such approach is that a person with an adapted controller device (e.g. a mobile phone) should be able to interact and control different devices by means of an interoperability architecture. Using an appropriate controller, the user control over the system is much easier to achieve as the accessibility problem of the whole environment is reduced to just solving the accessibility issues with the user controller. The architecture, using different interoperability protocols and frameworks (such as URC [20], [21], OSGi and Web Services standards stack) facilitates the interaction in accordance with the specific nature of each *target device/service* in a unified manner.

The UML Deployment Diagram (DD) in Figure 1 depicts the most important components of the architecture.

The main processes performed by the architecture are pictured by the UML Interaction Overview Diagram (IOD) in Figure 2.

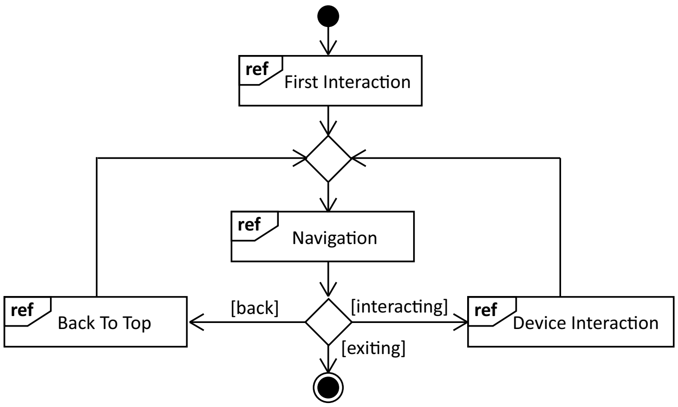

Figure 2. UML Interaction Overview Diagram of the main processes of the architecture [10].

- **First Interaction.** It consists in the creation of the initial interface that acts as the access medium to the environment for the user. In the generation of such interface the system must take into consideration the relevant set of devices and services for the user (the perimeter) and their state (without forgetting the special needs of the user). This process involves an interface generation subprocess, in order to build an accessible XHTML interface, and the determination of the set of assistive software instances that permit the user to interact which such interface.
- **Navigation.** Once the user has selected the device or service to interact with, the navigation process starts. Devices and services are defined by complex multi-staged interface descriptions that users can navigate. Through navigation, we simplify the information offered at a time to the user and we allow complex conversations with the device.
- **Device interaction.** When user navigation ends, or when the user performs certain actions in the device interface, interactions with the end device occur. The architecture supports interactions with devices either as a UCH Target or as a Web service transparently.
- **Back to top.** The user can, at any moment, reset its interaction with the device, going back to the first interface that the device offers. An updated initial interface of the device must be rendered again.

As detailed in [10], the main processes can be summarized with the following example: A user stays at home. S/he wants to turn the TV at home on. First, the user logs in with his/her name and password using his/her mobile phone (*device controller*). A screenshot with the available devices and services, grouped by environment, is displayed (**First Interaction**), e.g. it reads "Smart Home", "Products and Services" and "Health Care", among others. These devices and services depend on the user's location. The user navigates through the screenshots until s/he identifies the device or service that s/he wants to control (**Navigation**); for instance, in the "Smart Home" display, s/he selects "TV set" (*target device*) and "Turn on/Turn off" options. S/he turns on the TV (**Device Interaction**) and waits for the notification of the new status. Finally, s/he comes back to the first screenshot in order to interact with other device or service (**Back to top**). Obviously, all the screenshots must be accessible and adapted to the specific needs and preferences of this user. Figure 3 shows the architecture at high level.

III. USER TESTING

The architecture was validated for running prototypes, thus some pivotal pieces (modules) were implemented and tested. Tests considered diverse user disabilities, preferences and profiles. The main challenge was to measure the satisfaction of the user experience with diverse interaction modes of services and devices for people with special needs. This level of satisfaction included accessibility and usability aspects as well as other non-functional objectives, such as performance or security. The user testing phase was performed in a smart home prototype. Sainz et al. described in [4] the methodology carried out during this stage, as well as the chief outcomes from the viewpoint of the user interaction and experience.

In this phase, the number of concurrent users never exceeded 5 people, due to the logistical difficulties of real experimentation. At first sight, it may seem that the experimental controlled tests are a thin basis due to the small number of users involved. The burden of experimentation in this environment spans several issues (e.g., logistics, user selection or ethical). In the following we detail a few of these issues:

- The user's tests were carried out in a rented special facility (smart home) in the Alicante city (Spain). The space was small and the number of devices limited. For each disabled person the attendance of a team of two people was needed. For twelve users, a team larger than twenty four people in the small room was working. The persons of the teams were hired from different Spanish cities due to the heterogeneity of the technicians, some of them traveling every day to this city.
- Users were selected according to different profiles of disabilities (deaf, cognitive, partial blind, blind, congenital blind, etc.) and among elderly people. We had to combine them to get meaningful groups for the tests. Notice that the tests were not only focused on performance but mainly on usability. It was not easy to form these groups in a small city, and it was not easy to manage the user tests to get results for different purposes.
- Each test lasted for at least two hours. For people with some disability it is a great challenge to keep attention a relatively long time. On the other hand, people with intellectual disabilities may overrate some of their capabilities, so the tests had to be done at least in two stages. For deaf people it was necessary to hire sign language interpreters. For elderly people it was difficult to use tablets, smartphones, etc. Therefore carrying out tests for 5 concurrent users in special groups, analysing different results, was very challenging in this context.
- The high budget requited to support all the team and infrastructure was not negligible.

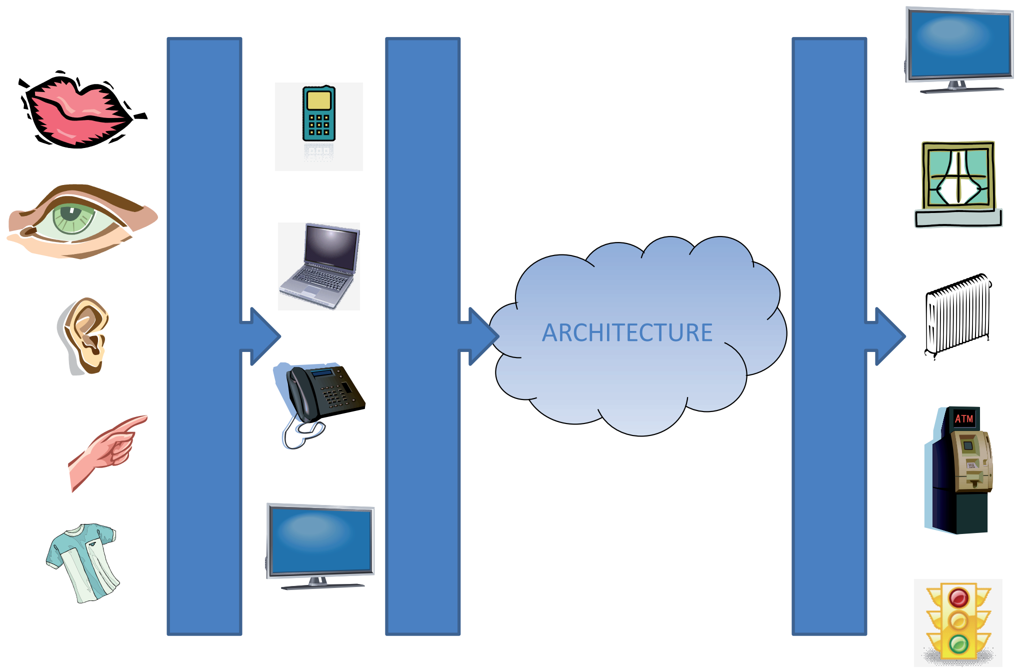

Figure 3. Description of the architecture at high level.

- Finally, some articles of the Spanish Civil Code Law with respect to the group of people with disabilities, did not facilitate the tests.

The different issues described above show the great difficulties that had to be overcome by the team of testers. Therefore, the realisation of more advanced tests would be very difficult to carry out in reality. In a nutshell, the costs of the team carrying out the experiments, as well as the rent of the facilities (i.e. the smart home), prevented the execution of more complex experimentations involving more individuals.

During the user testing phase, the development team, including programmers and software engineers, noticed that some users, namely people with cognitive disabilities, faced some difficulties to understand the device interactions. In addition, they realized the high impact that the interface interaction had for the user experience, chiefly for users with special needs, as well as their lack of sufficient knowledge in this discipline.

Hence, experimentation problems and limitations of real implementations advocated the use of models, such as Generalized Stochastic Petri nets (GSPN) [22], specially in the initial phases of the system life-cycle. Models can represent the system in a variety of hypothetical situations and can perform different kind of analysis at a lower cost, as established in [10], [11], [23].

IV. Usability and Accessibility Teaching

Throughout the system design phase, the engineers applied a plethora of techniques aimed at increasing the dependability of the system. These engineers had a considerably strong expertise in fundamental topics for embedded systems such as safety, maintainability and performance, as well as in methodologies to assess the fulfillment of the corresponding functional and non-functional properties [24]. This fact resulted in the achievement of a high Quality of Service, as remarked in [25]. Besides, as the target audience included people with special needs, the system engineers worked in cooperation with a team of psychologists and sociologists. Such multidisciplinary team of specialists is not uncommon nowadays. However, we noticed that the engineers were completely oblivious to the singularities of the aforementioned target audience (with respect to accessibility), which hindered their cooperation with other specialists. Note that the first steps towards the acquisition of their expertise by the system engineers undoubtedly emanate from their education. Therefore, the difficulty to cooperate with different specialists in the process of software engineering may be traced back to the lack in their education of contents related to accessibility for people with special needs. These contents, even if very basic, would have greatly facilitated the development of the different tasks.

With regards to the engineering of embedded systems, the different study plans typically introduce mechanisms that enable the interaction of a software system with other entities. The interaction between different software entities, also known as interoperability, is crucial in nowadays component-based systems and the widespread Service-Oriented Architectures [26]. Several technologies facilitate the interoperability of systems. For instance, consider the well-known eXtensible Markup Language (XML) [27]. XML provides a (standardised) means to encode information in a human-readable and machine-readable way. This technology facilitates the exchange of information between different systems, hence making their interoperability easier. This and similar interoperability-related technologies belong to the programmes of nowadays software engineering studies.

The interaction between software entities and humans, also known as Human Computer Interaction (HCI), also belongs to such programmes. The topics of HCI typically covered include general-purpose recommendations that enhance the usability of software systems. As described in [28]: *"Usability is a quality attribute that assesses how easy user interfaces are to use"*. The term *"usability"* also refers to methods for improving ease-of-use during the design process. The usability of the system described in Section II has been addressed by the application of the Nielsen Usability Principles [28]. These principles establish a series of guidelines for a wide variety of aspects of software systems. As a matter of example, consider the case of the system's response time, understood, from the user's perspective, as the number of seconds required to respond to a user request.The recommendations in [28] determine then that:

- 0.1 seconds represents a maximum threshold such that, if not exceeded, the system's reaction is perceived as instantaneous by the user.
- 1.0 second is, roughly, the time span that a user's flow of thought may be expected to stay uninterrupted. However, the delay introduced by a response time of this magnitude would not remain unnoticed.
- 10 seconds represents, roughly, an upper bound in order to keep the user's attention on the dialogue. In order to maintain the user's focus of attention, an estimate about the response time should be given to them. Therefore, the system must be instrumented so as to incorporate the capabilities necessary to provide such feedback.

Note then that the usability techniques enacted did not suffice to accomplish the desired user experience. These techniques address the usability of systems for the general public and disregards users with special needs. The ability of a system to be used by as many people as possible is known as accessibility. As mentioned before, in order to enable the use of the system by people with special needs, as well as to evaluate the reactions elicited from them, the involvement of psychologists and sociologists was required. Therefore, from our experience, some basic knowledge about accessibility for people with special needs should be given to software engineers in order to facilitate their cooperation with the aforementions specialists. Therefore we consider as being of interest the development of teaching methodologies for accessibility. A possible approach may be inspired by successful methodologies aimed at teaching usability to engineers like, e.g., the Scenario-Based Usability Engineering described in [29]. The development of such techniques lies beyond the scope of this work and is then left as future line of work.

V. Conclusions

In this paper, we have gained insight in the importance of accessibility and usability of embedded systems with human interaction. We utilise a real use-case to motivate the conclusions reported in this section. This use-case describes a smart home embedded system whose target audience is comprised by people with special needs. We have noticed that a significant proportion of the software engineers involved in the development process had received training on usability techniques. However, few of them had received enough, if any, training on accessibility for people with special needs. This lack of training impeded their collaboration with other necessary specialists (psychologists and sociologists in this case) in the different phases of the system development, hence encumbering such process. We believe that incorporating some basic knowledge on accessibility for people with special needs to the study programmes for software engineers would facilitate the development of embedded systems for this target audience.

As future lines of work we consider the development of teaching methodologies for the accessibility of people with special needs. We intend to test these teachings on our software engineering students. Besides, we also consider the incorporation of other knowledge that we find relevant for the development of embedded systems, such as the UML profiles for real-time (MARTE [30]), safety [31] and dependability [32].

Acknowledgment

The research described in this paper arises from the Spanish research project INREDIS (INterfaces for RElations between Environment and people with DISabilities). INREDIS was funded by the CDTI (Industrial Technology Development Centre) under the CENIT (National Strategic Technical Research Consortia) Programme, within the framework of the Spanish government's INGENIO 2010 initiative. The opinions expressed in this paper are those of the authors and are not necessarily those of the INREDIS project partners or of the CDTI.

The research leading to these results has received funding from the ARTEMIS Joint Undertaking, under grant agreement n^{o} 295373 (project nSafeCer) and from National funding.

References

[1] A. F. Newell, "Accessible Computing–Past Trends and Future Suggestions: Commentary on "Computers and People with Disabilities"," *ACM Trans. Access. Comput.*, vol. 1, no. 2, pp. 9:1–9:7, 2008.

[2] Eurostat, "Statistical Office of European Union," 2013. [Online]. Available: http://epp.eurostat.ec.europa.eu

[3] United Nations, "Factsheet on Persons with Disabilities," 2014. [Online]. Available: http://www.un.org/disabilities/default.asp?id=18

[4] F. Sainz, J. Casacuberta, M. Díaz, and J. Madrid, "Evaluation of an Accessible Home Control and Telecare System," in *Proc. 13rd Human-Computer Interaction (INTERACT'11)*, ser. Lecture Notes in Computer Science. Springer, 2011, vol. 6949, pp. 527–530.

[5] M. Alvargonzález, F. Sainz, J. A. Gutiérrez, J. Casacuberta, F. Flórez-Revuelta, M. Díez-Cabo, and J. Madrid, "Personalización automática de interfaces de usuario para el control domótico accesible," in *Actas del Congreso Internacional De Diseño, Redes de Investigación y Tecnología para Todos (DRT4ALL)*, June 2011.

[6] R. Giménez, M. Pous, and F. Rico-Novella, "Securing an Interoperability Architecture for Home and Urban Networking: Implementation of the Security Aspects in the INREDIS Interoperability Architecture," in *Proc. 26th Int. Conf. on Advanced Information Networking and Applications Workshops (WAINA'12)*. IEEE Computer Society, 2012, pp. 714–719.

[7] A. Murua, I. González, and E. Gómez-Martínez, "Cloud-based Assistive Technology Services," in *Proc. 3rd Workshop on Software Services: Semantic-based Software Services (WoSS) at the Federated Conf. on Computer Science and Information Systems (FedCSIS'11)*, 2011, pp. 985–989.

[8] M. Alvargonzález, E. Etayo, J. A. Gutiérrez, and J. Madrid, "Arquitectura orientada a servicios para proporcionar accesibilidad," in *Actas del V Jornadas Científico-Ténicas en Servicios Web y SOA (JSWEB'10)*. Editorial Garceta, September 2010.

[9] M. Pous, C. Serra-Vallmitjana, R. Giménez, M. Torrent-Moreno, and D. Boix, "Enhancing accessibility: Mobile to ATM case study," in *Proc. IEEE Consumer Communications and Networking Conf. (CCNC'12)*. IEEE Computer Society, 2012, pp. 404–408.

[10] E. Gómez-Martínez, R. González-Cabero, and J. Merseguer, "Performance Assessment of an Architecture with Adaptative Interfaces for People with Special Needs," *Empir Softw Eng*, 2013, accepted for publication.

[11] E. Gómez-Martínez, "Software Performance Assessment at Architectural Level: a Methodology and its Application," Ph.D. dissertation, University of Zaragoza, Spain, February 2014.

[12] C. Stephanidis, "Adaptive Techniques for Universal Access," *User Modeling and User-Adapted Interaction*, vol. 11, pp. 159–179, March 2001.

[13] G. Margetis, M. Antona, S. Ntoa, and C. Stephanidis, "Towards Accessibility in Ambient Intelligence Environments," *Ambient Intelligence*, vol. 7683, pp. 328–337, 2012.

[14] M. F. Cabrera-Umpiérrez, A. Rodríguez Castro, J. Azpiroz, J. B. Montalvá Colomer, M. T. Arredondo, and J. Cano-Moreno, "Developing Accessible Mobile Phone Applications: The case of a Contact Manager and Real Time Text Applications," in *Universal Access in Human-Computer Interaction. Context Diversity*, ser. Lecture Notes in Computer Science. Springer, 2011, vol. 6767, pp. 12–18.

[15] G. Zimmermann and G. Vanderheiden, "Accessible design and testing in the application development process: considerations for an integrated approach," *Universal Access in the Information Society*, vol. 7, no. 1-2, pp. 117–128, 2008.

[16] INREDIS Consortium, "INterfaces for RElations between Environment and people with DISabilities," 2010. [Online]. Available: http://www.inredis.es/

[17] R. González-Cabero, "A Semantic Matching Process for Detecting and Reducing Accessibility Gaps in an Ambient Intelligence Scenario," in *Proc. 4th Int. Symposium of Ubiquitous Computing and Ambient Intelligence (UCAmI'10)*. IBERGACETA Publicaciones, 2010, pp. 315–324.

[18] S. H. Park, S. H. Won, J. B. Lee, and S. W. Kim, "Smart home – digitally engineered domestic life," *Personal Ubiquitous Comput.*, vol. 7, no. 3-4, pp. 189–196, Jul. 2003.

[19] G. Zimmermann and G. Vanderheiden, "The Universal Control Hub: An Open Platform for Remote User Interfaces in the Digital Home," in *Proc. 12th Int. Conf. Human-Computer Interaction. Interaction Platforms and Techniques (HCI'07)*, ser. Lecture Notes in Computer Science, vol. 4551. Springer, 2007, pp. 1040–1049.

[20] ISO/IEC, *24752-1:2008-Information technology–User interfaces–Universal remote console–Part 1: Framework*, Std. ISO/IEC 24 752-1, 2008.

[21] URC Consortium, "Universal Remote Console," Specification available at: http://myurc.org, 2005.

[22] M. Ajmone Marsan, G. Balbo, G. Conte, S. Donatelli, and G. Franceschinis, *Modelling with Generalized Stochastic Petri Nets*, 1st ed., ser. Wiley Series in Parallel Computing. J. Wiley, 1995.

[23] E. Gómez-Martínez and J. Merseguer, "Performance Modeling and Analysis of the Universal Control Hub," in *Proc. the 7th European Performance Engineering Workshop (EPEW'10)*, ser. Lecture Notes in Computer Science, vol. 6342. Springer, 2010, pp. 160–174.

[24] H. Espinoza, H. Dubois, S. Gérard, J. M. Pasaje, D. C. Petriu, and M. Woodside, "Annotating UML Models with Non-functional Properties for Quantitative Analysis," in *Procs. of the 2005 Int. Conf. on Satellite Events at the MoDELS*, ser. Lecture Notes in Computer Science, vol. 3844. Springer, 2005, pp. 79–90.

[25] *Int. ACM Sigsoft Conf. on the Quality of Software Architectures*. SIGSOFT, 2005-2013.

[26] T. Erl, *SOA Principles of Service Design*. Prentice Hall PTR, 2007.

[27] T. Bray, J. Paoli, C. M. Sperberg-McQueen, E. Maler, and F. Yergeau, "Extensible markup language (XML)," *World Wide Web Consortium Recommendation REC-xml-19980210*, 1998.

[28] J. Nielsen, *Usability Engineering*. Morgan Kaufmann, 1993.

[29] M. B. Rosson, J. M. Carroll, and C. M. Rodi, "Case studies for teaching usability engineering," *SIGCSE Bull.*, vol. 36, no. 1, pp. 36–40, Mar. 2004. [Online]. Available: http://doi.acm.org/10.1145/1028174.971315

[30] OMG, "A UML profile for Modeling and Analysis of Real Time Embedded Systems (MARTE)," 2011, version 1.1. [Online]. Available: http://www.omgmarte.org/

[31] OMG, "UML Profile for Modeling Quality of Service and Fault Tolerance Characteristics and Mechanisms (QoS & FT)," 2008, version 1.1. [Online]. Available: http://www.omg.org/spec/QFTP/

[32] S. Bernardi, J. Merseguer, and D. C. Petriu, "Dependability modeling and analysis of software systems specified with UML," *ACM Comput. Surv.*, vol. 45, no. 1, p. 2, 2012.