Universally Accessible Task-Based User Interfaces

Ane Murua, Xabier Valencia, Eduardo Carrasco
Vicomtech
San Sebastian, Spain
{amurua, xvalencia, ecarrasco, info}@vicomtech.org

Gottfried Zimmermann
Access Technologies Group
Pfullingen, Germany
gzimmermann@acm.org

Bruno Rosa, Jürgen Bund
Meticube
Coimbra, Portugal
info@meticube.com

Jan Alexandersson
DFKI GmbH
Saarbrücken, Germany
Jan.Alexandersson@dfki.de

Unai Díaz
Ingema
San Sebastian, Spain
unai.diaz@ingema.es

Abstract— **This paper introduces a new AAL architecture intended to simplify and enhance the end user interaction with the technology. The proposed concept makes state-of-the-art task model technology available and accessible to all types of users. The concept relies on the integration of both ANSI/CEA-2018 Task Model Description (CE TASK 1.0) and ISO/IEC 24752 Universal Remote Console Framework standards. Additionally, a proof-of-concept implementation has been carried out which assist people in performing blood pressure measurements. Finally, a validation involving 8 elderly persons suffering from Alzheimer's disease has been carried out, obtaining encouraging results.**

Keywords- Independent Living; Usability;Interoperability Task-Based User Interfaces; Universal Control Hub

I. INTRODUCTION

The global proliferation of increasing longevity in modern society has fixed a focus on assisting the elderly and those living with disability through computing technology [1].

On December 2009 the European Day of People with Disabilities 2009 conference took place at Brussels, focusing on the right of persons with disabilities to live independently and on making that right a reality. The conference took place at a time when the European Union and the Member States are in the process of ratifying and implementing the UN Convention on the Rights of Persons with Disabilities [2]. Special emphasis was placed on Article 19, which refers to "Living independently and being included in the community" and declares that "States Parties to this Convention recognize the equal right of all persons with disabilities to live in the community, with choices equal to others, and shall take effective and appropriate measures to facilitate full enjoyment by persons with disabilities of this right and their full inclusion and participation in the community" [3].

In this sense, efforts are being carried out in order to de-institutionalize people, helping them in the transition from institution to community care and guaranteeing them a satisfactory Independent Living. In this scenario, Ambient Assisted Living (AAL) becomes a key stakeholder.

According to the definition given by AALIANCE [4], AAL refers to intelligent systems of assistance for a better, healthier and safer life in the preferred living environment and covers concepts, products and services that interlink and improve new technologies and the social environment. It aims at enhancing the quality of life (the physical, mental and social well-being) of everyone (with a focus on elder persons) in all the stages of their life.

People getting benefit from AAL could pertain to many different collectives, being the elderly and/or the disabled the ones intended to get the greater profit of those systems.

When designing new AAL applications, researchers and developers often focus in technological innovations, forgetting that their target users have limited or no technological skills. In this sense, User Interfaces (UIs) are a key part of any application intended to be usable and accessible. Without a proper UI there is nothing to do.

It does not only depend on the UIs, it also has to do with properly helping and guiding the users in order to simplify the interaction. The users should be provided with a simplified interaction UI, to be able to carry out complex tasks in a simple manner and without getting lost.

The complexity of a task could increase for different reasons: a lot of steps to be performed by a person; multiple devices involved at the performance of the task; or both of them.

This research was partially funded by the EU 6th Framework Program under grant FP6-033502 (i2home) [5]. The opinions expressed herein are those of the authors and not necessarily those of the funding agencies.)

978-1-4244-6376-3/10/$26.00 ©2010 IEEE

The ideal scenario would be where the user is not aware of those issues. It means hiding low level tasks, avoiding bothering users with non-sense decisions, or guiding the user into those decisions.

For example, when the user wants to carry the task "watch a movie" what he wants is just the result: to watch a movie. Other actions such as switching the TV on, switching the movie player on, adjusting the volume, and so on, are just part of the process, but not really relevant for the user.

In other words, what matters to users is “What to do” rather than “How to do it” [6]. So, in this sense, we believe that Task Modeling will help to achieve user friendliness in the field of AAL applications.

II. State of the Art

A. *Universally Accessible UIs*

The Universal Remote Console (URC) framework, specified as a family of ISO standards [7], defines a “Protocol to Facilitate Operation of Information and Electronic Products through Remote and Alternative Interfaces and Intelligent Agents”. See Fig.1. Its purpose is to define a user interface layer on top of any existing interoperability framework for device discovery, control and eventing.

The Universal Control Hub (UCH) is a gateway based architecture for implementing the Universal Remote Console (URC) framework in the digital home. It reaps the benefits of the URC framework without requiring URC-compliant targets and controllers. The UCH architecture was originally proposed for UPnP based home networks, but is also applicable to any other networking platform and any combination of them.

The UCH architecture adds the ability for 3rd-party user interfaces and possible use of intelligent agents for task-based and/or natural-language user interfaces [8]. This option makes the UCH a suitable tool for the AAL field, where pluggable UIs become a key feature.

As an extension to the URC framework, we are proposing to add a standard on Task Model Descriptions. Hereby the task model tree would be based on the UI Socket, i.e. its leaves would be elements of one or multiple UI Sockets. Thus the Task Model Description would be independent from any networking platform specifics, and 3rd parties could contribute to them.

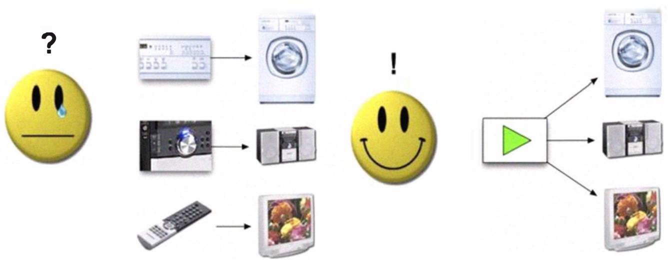

Figure 1. The role of the URC on user-machine interactions

B. *Task Modeling*

The best-known techniques for task modeling are based on hierarchical task representation, where tasks are decomposed in subtasks, have an order of execution and include preconditions, effects and subgoals that compose the execution of the task [9].

There is a great variety of standards and techniques which intend to describe tasks formally. Some of the efforts coming from the world of remote controlling of the devices, such as consumer electronics (UPnP) or laboratory equipment (LECIS), could be so powerful when controlling single devices, but fail when the task involves multiple steps on a diversity of devices.

Currently, ConcurrTaskTree notation (CTT) seems to be a very helpful tool in the field of User Interface design. The strength of this notation relies on its expressive power, and on its easy-to-use graphical syntax. On one hand, those features are interesting for interface designers but, on the other hand, they are too expensive in terms of computational processing. It is not a problem at the UI design time, but it becomes a handicap when task needs to be processed at the time it occurs, especially if it takes place at common home computer or embedded into a CE device.

CTT is part of the TERESA XML set of user interface description languages. Some interesting work has been carried out combining TERESA XML and Web Services-based home interoperability architectures [10]. This approach demonstrates the automatic model-based UI generation for ubiquous home-appliance controlling, but it is still missing some linkage with user guidance into a task supported by a task model.

ANSI/CEA-2018 is a standard, developed by the Working Group 12 of the CEA committee on home networks (CEA R7 WG12) which makes it possible for devices from different manufacturers to interoperate, supporting high-level integrated services that involve multiple steps on multiple devices and/or carried by humans. It is an XML-based language for the description of task models, which distills the key features of task models in a way that enables practical runtime use in CE and similar low-cost applications [11].

The facts mentioned above made us choose ANSI/CEA-2018 as the core of the Task Model runtime engine that has been integrated at the Universal Control Hub.

III. Proposed Architecture

The architecture mixes the URC and ANSI/CEA-2018 Task Model Description standards, in order to create an innovative architecture. This architecture enables the task modeling that can interact with the user and also with electronic devices, interoperability and among other features, the creation of adaptive UIs.

Now we will explain the UCH architecture before integration of the ANSI/CEA-2018 Task Model Description, and the results of the changes in it when task models are incorporated.

The main objective of the Universal Remote Console framework (URC) and Universal Control Hub (UCH) as an implementation of the URC is to separate device's functionality from device.

There are two kinds of devices in the URC ecosystem targets (air conditioning, blood pressure meter…) and clients (PC, PDA, TV….). Some devices like TV can be targets or clients depending of the action that is been performed. If we use a PC to change channels, the TV acts as a target. But if we use it to control the air conditioning system it will act as a controller.

The UCH is a highly modular and scalable architecture that allows the support for new controllers and targets without any implementation changes.

The URC standard defines an abstract layer called, UI Socket layer. This layer exposes to the client layer, the functionalities of the device that have been described in a XML file. This allows the development of user interfaces for any type of user and since this architecture is standards based, the development of compatible user interfaces is open to any third party.

Current devices aren't URC standard compliant so an adaptation is needed for them. The responsible of doing that adaptation is the Target Adapter layer. This layer allows the communication between UCH and targets. In other words it translates the protocol of the device to the URC standard protocol. Clients also need an adaptation that is provided by the Client layer.

In a later step of the i2home project [5], the ANSI/CEA-2018 Task Model Description was added to the UCH as can be seen in Fig. 2. This new module that support task modeling was called Activity Management System.

The developed task engine deals specifically with task models compliant with the ANSI/CEA-2018 standard. The task engine implementation is responsible for parsing the task model .XML file and to create its runtime representation. It is also responsible for obtaining tasks and task models referred in the original task model and to keep the runtime representation of the task model environment up to date with the latest changes and with the relevant changes on the surrounding environment.

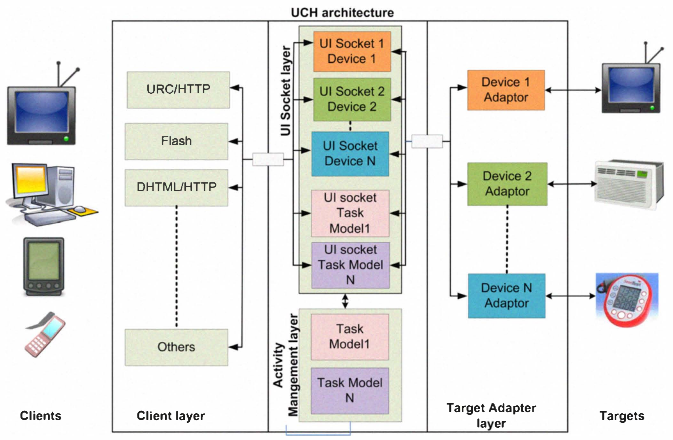

Figure 2. UCH architecture plus ANSI/CEA-2018 task model description

The Activity Management System architecture is modular. It has a manager (Activity Management Manager) that is responsible for each individual Activity Management Module and handles the interaction between the sockets and the AM modules and vice-versa.

Each Activity Management Module is responsible for providing the runtime environment necessary to execute a CEA-2018 task model. To accomplish this, each individual AM module has its own task engine instance and its own Ecmascript engine.

The Activity Management Modules can also expose a socket to allow interaction with the user. Each socket should be made specific to the task model that is intended to be executed in the AM module. This way the socket will have all the socket elements necessary to display and request all the necessary information to/from the user. It is also possible to have task models that don't require interaction with the user, these won't have any socket associated to them.

A task model was created with the purpose of interacting with the user. This task model contains specific pre-defined tasks that allow that interaction. Some of them are: "Message" to display a message, "InputRequest" to get user input data, "SelectOneRequest" to get one of the available options and "ConfirmRequest" to receive a confirmation (yes/no).

In order to allow the interaction of an Activity Management module with the users, the task model that is being executed in the Activity Management module will have its own socket description. This socket description must contain mapping elements to map socket elements present in this socket to tasks, steps and input/output slots present in the task model.

There are two types of mappings:

• Step mapping: this mapping is used to map all the existent input/output values in a step in the task model. For example, we have an input slot from a step referent to a task that represents the channel that the user wants to change on TV. To allow the user to insert this value we add a variable to the socket description that will contain it. By using this mapping when the user inserts the channel value in his user interface, the Activity Management assigns that value to the proper step input slot. Step mapping is also used to present data to the user, not only for getting the user response. An example is, in the above example, to display the message to ask the user to select the TV channel, we can use mapping between a slot that contains the message to be presented and a variable socket element which will display the message to the user in its user interface.

• Task mapping: this mapping is used to map command elements to tasks (or steps), so that when a user invokes the command in its user interface the AM knows what task (or step) to execute (if possible).

This integration allows the task model to interact with all the existing targets (blood pressure meter, TV…), clients and task models seamlessly. For example the task "watch a movie" could involve the following steps, adjust the temperature of the room, dim the light, switch on TV, and play movie. That would involve light, air conditioning and TV targets.

978-1-4244-6376-3/10/$26.00 ©2010 IEEE

Since task model can be exposed to the UI Socket layer, it can be used like a target, allowing the reusability of the task model, customizing the UI for the target user group

IV. Proof of Concept: Blood Pressure Measurement

A. Proposed Scenario

Manuela, an elderly person with a diagnosis of mild-to-moderate Alzheimer's disease, lives with her daughter Maria, because sometimes had problems with some daily tasks.

She needs to check her blood pressure on a daily basis. She is provided with an easy-to-use home blood pressure meter device CareTec SweetHeart [12], but often she forgets to do it. Furthermore, she is even forgetting how to use it. The system will assist her to successfully accomplish this daily task.

Maria would like to know Manuela's blood pressure measure, so she selects her preferred UI and introduces the date and hour of the measurement. Also she selects how receive the results (e-mail, SMS or both). See Fig. 3

Manuela loves watching TV and while she is watching it, the TV broadcast is paused, an introducing music sounds and an avatar appears on the TV saying to Manuela that it's time to measure the blood pressure.

The avatar guides Manuela step by step explaining how to use the blood pressure meter with voice and a demonstrative video that highlights key moments. Later the system acquires the measurement and acknowledges Maria or the caregiver with a SMS, e-mail or both about the measurement results.

Manuela didn't had enough time to complete the blood pressure measurement correctly so the system advices Maria that there has been an error in the measurement. After a while a second attempt is repeated.

This time Manuela has completed the measurement successfully. While she is taking off the blood pressure cuff a SMS and e-mail is been send to Maria with the blood pressure and heart rate measures.

This has been the proposed scenario but, due to the UCH approach, the same task model could be applied to other people, each of them using their needed or preferred UI and controller (TV, PDA, PC…).

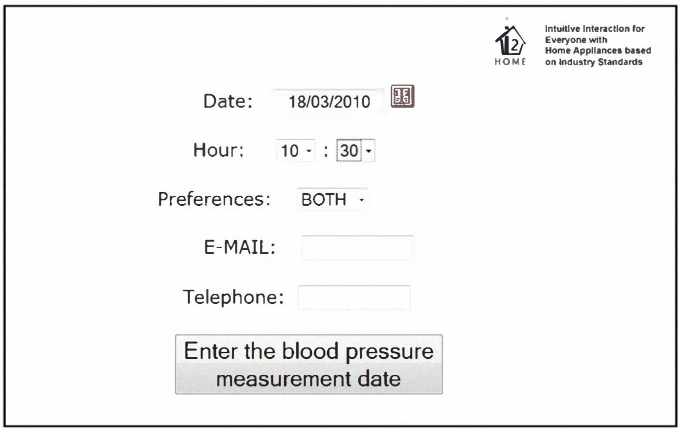

Figure 3. Caregiver's UI

B. Implementation of the Scenario

For the proposed scenario we integrated different devices: a TV, a computer and a blood pressure meter. The computer had Windows Media Center (WMC) that can be found in Windows Vista Home Premium and Ultimate versions [13] or in most of the Windows 7 versions [14]

There are various URC implementations in different program languages such as C# (Meticube's UCH), java (UCHj) and C++ (UCHe) The last two implementations are open source and can be downloaded from [15] and [16]. For this scenario C# implementation has been used. Meticube's UCH, at this moment is the only one that implements the ANSI/CEA-2018 task model.

The UCH was running under the WMC as a background application and its targets were the blood pressure meter, the SMS-Email sender, the reminder and the WMC TV. These targets were integrated previously and, due to the flexibility of the system, these targets have been reused without any changes.

UCH's clients were Maria's UI and the avatar on TV. Fig. 4. The avatar on TV has been selected as the UI for Manuela because it is a natural-human computer interface that fits well to people with cognitive diseases like Alzheimer [17].

The UCH and the reminder and SMS-Email sender targets are available in UCH Extension Kit provided by Meticube. The WMC TV and Avatar Extension Kits are provided by Vicomtech. All of them are free of charge for non commercial purpose.

The task model is the responsible of the interaction between targets and client. In Fig. 5 can be seen, in general terms, the proposed task model diagram.

First the task model makes a connection with the reminder target. The reminder is a virtual calendar that raises agenda items previously introduced on it. So an event is raised when the measurement appointment arrives. The task model gets this event and continues with the following tasks.

The next task pauses the TV broadcast and sends messages that are processed by the avatar on TV client. In this specific scenario the avatar acts as output interface due to the user and scenario characteristics, but it could also act as an input-output interface depending on the scenario.

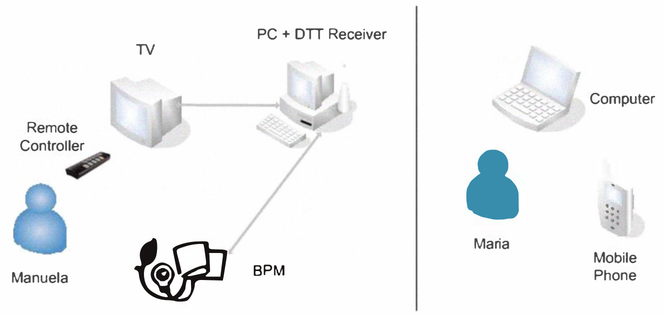

Figure 4. Devices involved at the proposed scenario

978-1-4244-6376-3/10/$26.00 ©2010 IEEE

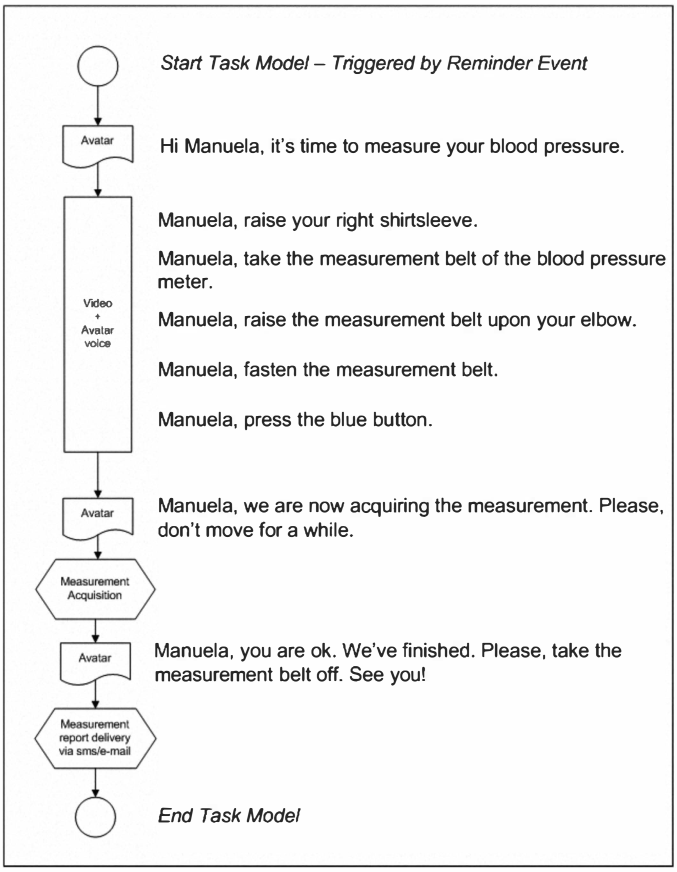

Figure 5. Proposed scenario's task model

After these instructions, the task model acquires the measures and sends a message to the caregiver with the results. At the same time, a message is sent to the user and finally the avatar is hidden and the TV displays normal TV broadcasting again.

Apart from the main subtasks listed above, there are other subtasks that take care of checking the validity of the blood pressure measurement. In case some error is detected in the measurement acquisition, the whole process will be repeated later on. And if an error is detected again at the second try, the caregiver will be reported through SMS or e-mail that the blood pressure measurement acquisition was not possible.

C. Validation

The main objective of the validation was to check that the task model was a good enough tool to improve the quality of life of people with Alzheimer disease but not only for them also of their caregivers, especially family caregivers.

The validation was carried out by Ingema [18] in the Matia Foundation's Bermingham Day Care Centre, San Sebastian, Spain. All the users suffered from Alzheimer's disease ranging from mild to moderate.

Some photos of validation's final setup from the user's point of view are shown in Fig. 6. Regarding caregivers, they don't need any special setup, just their pc or their cell-phone.

Figure 6. Validation's final setup

Before the tests, a short training was done individually with every tester. This was needed because of the difficulties to put on the blood pressure meter's belt. There were two main problems: the tube was too short, and some of them also had some troubles tightening up the cuff with only one hand, due to their reduced mobility.

After the training, the user test began, as mentioned below, watching the TV broadcast... Interviewers were present in the room, observing users' behavior during the entire process (without interfering with them) and annotated their observations in usability reports.

Even though it was not possible to conduct a wider experiment with more users (due to the characteristics of the target group), it was possible to identify relevant results.

Some users reacted speaking to avatar in a natural way and all of them accepted it speaking to them. For example a common reaction to the greeting or farewell of the avatar was to say hello or goodbye to it.

All of them copied what they saw in the explanation video. In this sense, the task model behind the video and the entire system seems to encourage and help them to complete the task. For example a woman with a reduce mobility that couldn't move her arm, said "I wish I could raise the arm like the woman on the video".

Thanks to the task model and the avatar on TV (as a personalized UI for them) they can be able to do complex tasks such as the blood pressure measurement. This would increase user's autonomy and start decreasing its need of a 24/7 caregiver supervision.

V. Ongoing Eforts

The proposed architecture is not only implemented in the aforementioned i2home project. Rather, various i2home project partners as well as further R&TD institutions and companies are applying the UCH and presented Task Model runtime engine in other applied research and product development projects of theirs.

Examples are the European projects “VITAL” (FP6), “BrainAble” (FP7), “BEDMOND” and “HELASCoL” (both AAL Programme) and i2Life (Portuguese QREN Programme). Furthermore the UCH and Task Model Engine are being applied in a product development project by Deutsche Telekom / T-Systems. Additionally, Meticube is conducting an internal project that will complement the UCH and Task Model Engine by a set of rich, interactive, graphical tools, called the UCH workbench. This project is also co-financed by the Portuguese QREN Programme (Ministery of Economy and Innovation).

At the time of compiling this paper, projects of various kinds, with a total budget of approximately 60 million Euros, involving more than 100 institutions, are being carried out in more than 20 countries in 3 continents.

VI. Conclusion

A new architecture for providing universally accessible task-based user interfaces has been described. The Universal Remote Console framework (ISO/IEC 24752) represents an open framework, facilitating any controller controlling any device or service through User Interface Sockets and pluggable user interfaces. In tandem with an emerging Task Model Description standard (CEA-2018), this approach allows for coherent, seamless, task-oriented and scalable user interfaces for AAL environments.

Based on this architecture, a fully functional prototype has been developed and tested with 8 users. Those users belong from a Care Center and all of them are elderly people suffering from Alzheimer’s Disease. Results of the testing allow us to assert that our architecture is valid as a solution of the aforementioned issues:

• We provide a framework that makes possible the use of proper user interfaces for AAL applications which let users interact with the system successfully.

• And moreover, we provide an architecture that allows the creation of task models that guide users at the runtime of the task and lets them to carry out complex tasks easily.

One of the biggest challenges of the AAL is that users must perceive the system as something so simple that allows them performing complex tasks in a natural way and without any help from the outside. We have demonstrated that the integration of two open and standardized platforms (i.e. UCH and CE TASK 1.0) allows people to have successful interaction with AAL and, furthermore, could empower them in their way to Independent Living.

References

[1] R. Hegarty, T. Lunney, K. Curran and M. Mulvenna, “Ambient interface design (AID) for aging, independence and disability,” 3rd European Conference on Ambient Intelligence (AmI09). Adjunct Proceedings, pp. 108-112, November 2009.

[2] C. Gooding, “Creating conditions for independent living,” EDPD 2009 Conference Report, January 2010.

[3] Convention on the Rights of Persons with Disabilities http://www.un.org/disabilities/convention/conventionfull.shtml

[4] AAliance Roadmap http://www.aaliance.eu/public/documents/aaliance-roadmap/aaliance-aal-roadmap.pdf

[5] I2home project http://www.i2home.org

[6] G. Zimmermann “Open user interface standards - Towards coherent, task-oriented and scalable user interfaces in home environments,” IET, 2007.

[7] ISO/IEC 24752. Information technology - User interfaces - Universal remote console - 5 parts. International Organization for Standardization (ISO), 2008.

[8] G. Zimmermann, G. Vanderheiden, C. Rich, “Universal control hub & task-based user interfaces,” URC Consortium, January 2006.

[9] C. Rich, “Building task-based user interfaces with ANSI/CEA-2018,” Computer, vol. 42, no. 8, pp. 20-27, August 2009.

[10] G. Mori, F. Paternò, L. D. Spano, “Exploting Web Services and Model-Based User Interfaces for Multi-device Access to Home Applications” , T.C.N. Graham and P. Palanque (Eds): DSVIS 2008, LNCS 5136, pp 181-193, 2008.

[11] V. Eyharabide, A. Amandi, “Automatic task model generation for interface agent development,” Inteligencia Artificial V. 9, Nº 26, 2005.

[12] CareTec SweetHeart http://www.ic-online.co.uk/shop/ct-sweetheart_118568.htm

[13] Windows Vista Media Center http://www.microsoft.com/windows/windows-vista/features/media-center.aspx

[14] Windows 7 Media Center http://windows.microsoft.com/en-US/windows7/products/features/windows-media-center

[15] http://myurc.org/tools/UCHj/

[16] http://myurc.org/tools/UCHe/

[17] E. Carrasco et al. “Natural interaction between avatars and persons with Alzheimer’s disease” K. Miesenberger et al. (Eds.): ICCHP 2008, LNCS 5105, pp. 38-45, 2008.

[18] Ingema http://www.ingema.es/ingles/home.php