Journal of Applied Computing Research, 4(1):1-12
January-June 2014
 - doi: 10.4013/jacr.2014.41.01

A situation-aware ubiquitous approach for assessing therapeutic goals in hospital environment

João Lopes
Universidade Federal do Rio Grande do Sul. Av. Bento Gonçalves, 950, 91501-970, Porto Alegre, RS, Brasil
jblopes@inf.ufrgs.br

Alexandre Souza
Lifemed S.A. Rua Giuseppe Mattea, 350A, 96050-080, Pelotas, RS, Brasil
alexandre.souza@lifemed.com.br

Denis Moura Junior
Hospital Israelita Albert Einstein. Av. Albert Einstein, 627, 05662-000, São Paulo, SP, Brasil
denis.moura@einstein.br

Rodrigo Souza
Universidade Federal do Rio Grande do Sul. Av. Bento Gonçalves, 950, 91501-970, Porto Alegre, RS, Brasil
rssouza@inf.ufrgs.br

Ana Pernas, Adenauer Yamin
Universidade Federal de Pelotas. Rua Gomes Carneiro, 1, 96010-610, Pelotas, RS, Brasil
marilza@inf.ufpel.edu.br, adenauer@inf.ufpel.edu.br

Cláudio Geyer
Universidade Federal do Rio Grande do Sul. Av. Bento Gonçalves, 950, 91501-970, Porto Alegre, RS, Brasil
geyer@inf.ufrgs.br

Abstract. The medication therapy process in the hospital environment is complex and error prone. We consider that the use of technology can automate the verification of this process, reducing the number of errors during the treatment. Thus, this work proposes a software architecture aimed to provide the patient's current situation awareness, called EXEHDA-TG. As a central contribution, this work provides the monitoring of the patient's vital signs, allowing the physician to confirm that the desired effect with the administration of medications is being reached. In the developed architecture, this effect desired by physicians in vital parameters is called Therapeutic Goal (TG). The evaluation of EXEHDA-TG was taken through three usage scenarios. The results have been satisfactory in the analysis of the evolution of vital signs.

Keywords: ubiquitous computing, situation awareness, therapeutic goal.

Introduction

Mark Weiser's classic article (Weiser, 1991) describes the basic assumptions of Ubiquitous Computing related to user access to computing environment, anywhere, any time, with any device, non-intrusively, keeping user focus on their activities. In this perspective, the computational system must interact in autonomic way, no matter where the user is, constituting a highly distributed, heterogeneous, dynamic and mobile environment (Costa *et al*., 2008; Caceres and Friday, 2012).

In this sense, one of the main research problems in the area of UbiComp is context awareness, which refers to the ability of applications to perceive changes in the characteristics of the ubiquitous environment, which are of its interest, and respond to these changes through an adaptation process (Kakousis *et al*., 2010).

The literature review indicates several challenges in the support of context awareness for ubiquitous applications, including: (i) context acquisition from distributed and heterogeneous sources, (ii) context processing and actuation on the environment, and (iii) context dissemination to interested users in a distributed and timely way (Bettini *et al*., 2010; Bellavista *et al*., 2012; Knappmeyer *et al*., 2013).

Considering these challenges, the support for obtaining the patients' situation is based on the perception of vital signs, which is provided by sensed information. These sensors interact with patients, providing support for the ubiquitous systems services that react according to the context. Due to the complexity of the sensed data, inferences about the context should be made based on highest level information, abstracted from individual sensed data. This high-level abstraction of the sensed data, which defines a specific state of an entity, is called situation. In this sense, situation awareness is a promising way to extend the traditional models of monitoring and alerting, supporting the development of more robust applications, which meet the needs of the community in a hospital environment (Ye *et al*., 2012).

This article proposes the design of a software architecture for a ubiquitous environment aimed at providing situation awareness of patients. The EXEHDA-TG monitors the evolution of vital signs, allowing the physician to confirm that the desired effect with the administration of prescription medications is being achieved. This behavior, desired by physicians in vital parameters, is called Therapeutic Goal (TG) in the proposed architecture.

As a central contribution, the architecture automates the acquisition and enables the assessment of vital signs based on the knowledge of physicians, which specify in the EXEHDA-TG the therapeutic goal to be achieved. The EXEHDA-TG enables sending alerts to health professionals in a ubiquitous perspective, providing mobile access to medical information.

The propose EXEHDA-TG has the premise of its integration with EXEHDA middleware (Execution Environment for Highly Distributed Applications). EXEHDA provides a software architecture that aims at creating and managing a ubiquitous environment, and promote the implementation of ubiquitous computing applications (Lopes *et al*., 2014).

The paper is organized as follows. The second section presents an overview of EXEHDA middleware services. The third section describes the architecture and functionalities of EXEHDA-TG. The forth section presents a case study. The fifth section discusses the related work. Finally, the last section presents the concluding remarks.

EXEHDA middleware overview

EXEHDA middleware was conceived to support the execution of ubiquitous applications. The main properties of EXEHDA applications are: distributed, mobile, adaptive and reactive to the context. EXEHDA is composed of several integrated services that are conceptually organized in subsystems: data and code ubiquitous access, uncoupled spatial and temporal communication, large-scale distribution, context recognition and adaptation. The subsystem integration is shown in Figure 1 (Augustin *et al*., 2008).

Regarding communications, EXEHDA currently provides, through Dispatcher, WORB, and CCManager services, three types of communication primitives, each one addressing a distinct abstraction level.

The Dispatcher Service corresponds to the lowest abstraction level, providing message-based communications. Message delivering is done through per-application channels, which may be configured to ensure several levels of protection for the data being transmitted. Protection levels range from data integrity, using digital signatures, to privacy through encryption mechanisms. Additionally, the Dispatcher Service uses a checkpointing/recovery mechanism for the channels, which is activated when a planned disconnection is in course. This feature may or may not be activated by the upper communication layers depending on its particular demands.

In order to make the development of distributed services easier, EXEHDA also provides an intermediary solution for communications, based on Remote Method Invocations, through the WORB Service. The programming model is similar to Java RMI, but optimized to ubiquitous requirements. More specifically,

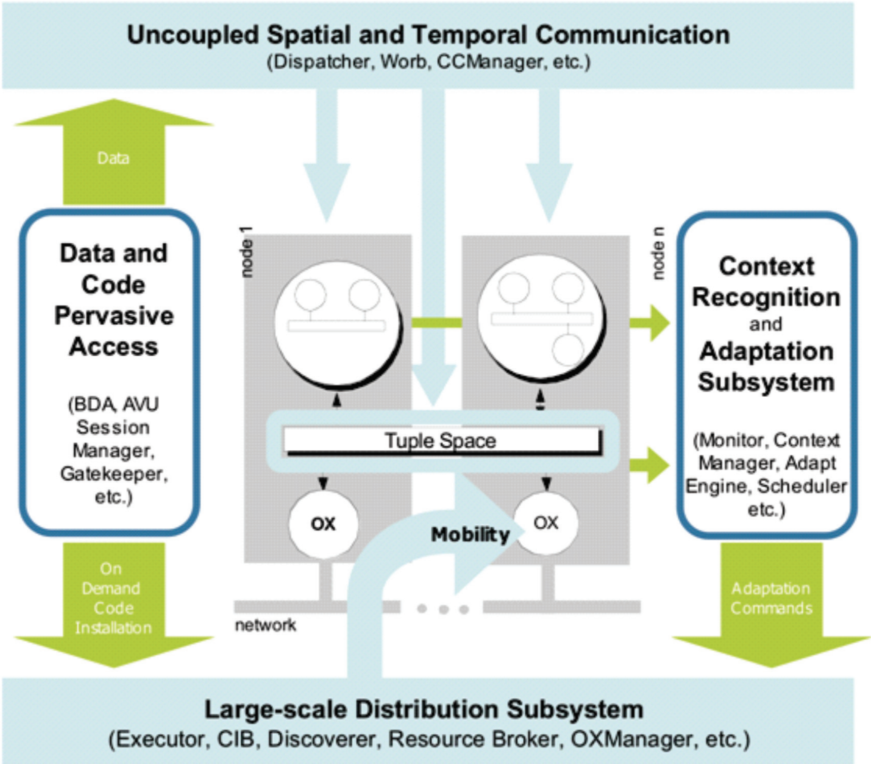

Figure 1. EXEHDA subsystems.

WORB remote method invocations, differently from Java RMI, do not require that the device keep connected during the entire execution of the method on the remote node. Instead, WORB was built on the functionality provided by the Dispatcher Service, including a per-invocation ID. The invocation ID remains valid during the disconnection, allowing the WORB to re-sync with the remote node after reconnection and obtain the returned values from the invocation.

At a higher level, the CCManager Service provides tuple-space based communications. It is built on the WORB Service, which also handles planned disconnections, providing applications to an anonymous and asynchronous communication support. This model is provided in the CCManager Service and is better suited to scenarios in which application components might migrate among nodes, since it does not require both sides to coexist for the communication to take place.

From the middleware point of view, environment resources fit in one of two categories: processing node or specialized resources. The former corresponds to the nodes, which effectively execute and whose access is managed by the middleware. The latter corresponds to specialized devices, e.g. printers, scanners, etc., whose access is not done through one of the middleware services, but through the use of some specific libraries. Although not managed by EXEHDA, the specialized devices are also cataloged in the CIB Service in order to allow applications to locate and use them.

The Discoverer Service is in charge of finding specialized resources in the environment based on an abstract definition of the resource. Typically, this service interacts with the CIB Service from its own cell, aiming at satisfying the resource discovery request in the scope of the local cell. When the local resources fail to fulfill the request, the Discoverer Service interacts with the Resource Broker service of the neighbors' cells. The strategy adopted in this extra-cell search is characteristic of the particular Discoverer Service instance in use. These services employ a language to describe resources and its interfaces are standardized. Since the middleware does not manage specialized resources, the results of a Discoverer Service search do not imply resource allocation or even resource reservation.

The Monitor Service implements a monitoring scheme based on sensors, which employs

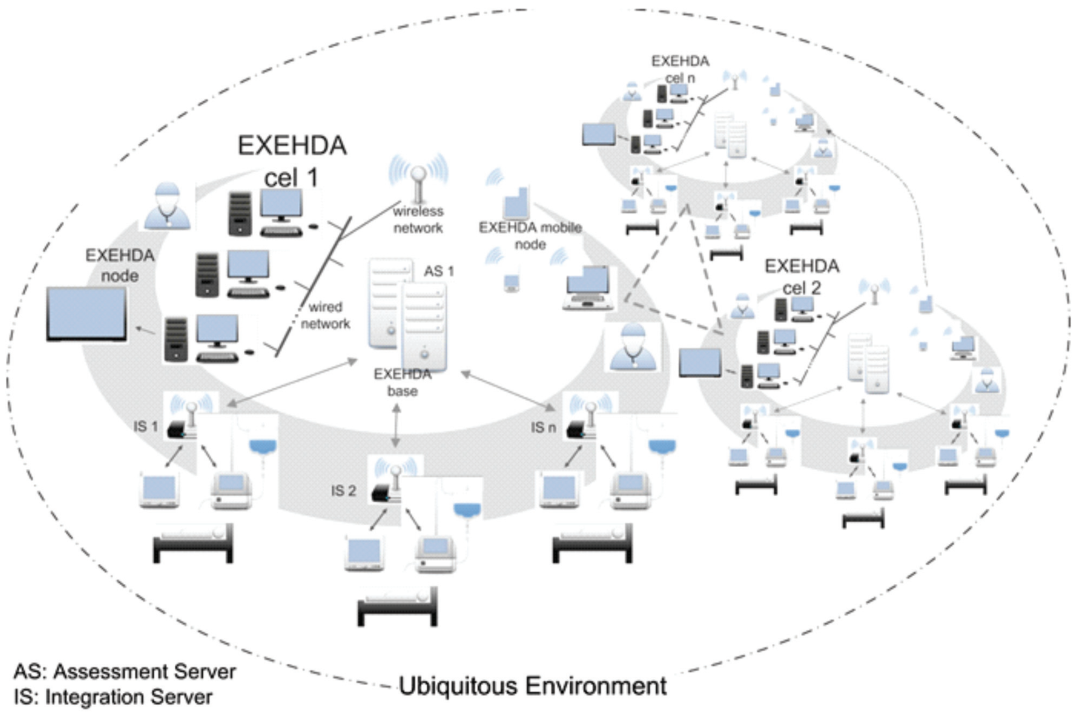

Figure 2. Ubiquitous Environment.

indexes to describe specific aspects of the environment. These sensors can be customized through parameters. The whole set of sensors installed on a node is part of the node description information registered in the CIB Service. The data generated by each sensor is gathered by the Monitor Service, which typically runs on the same node where sensors are installed. The gathered data is published by the Monitor Service to a Collector Service, which typically runs on the base-node.

Both data gathering by sensors and Collector Service publication by Monitor occurs in discrete multiples of a per-node configured quantum. The quantum parameter allows the resource owner to control, externally to the middleware, the degree of intrusion of the monitoring mechanism in the host. After a quantum of time expires, the Monitor Service executes a pooling operation over the active sensors in the node. Then, it applies the publishing criteria specified for the sensor data, determining, or not, the generation of a publishing event for that sensor. Thus, the events generated after a quantum expiration are grouped into a single message, reducing the amount of data that the Monitor has to transmit to the Collector. The Collector Service aggregates information from several monitors in the cell and forwards them to the registered consumers.

EXEHDA-TG architecture and functionalities

The EXEHDA-TG encompasses two types of servers: the Integration Server, responsible for the interaction with the environment through sensors and actuators, and the Assessment Server, responsible for storing and processing of contextual information related to patients, collected from the ubiquitous environment provided by EXEHDA.

The ubiquitous environment managed by EXEHDA has a cellular organization, where each cell can use multiple Integration Servers. Each Integration Server provides data to one Assessment Server, and can collect data from several hospital beds. A cell can serve one or more hospitals and a large hospital may have more than one cell. Figure 2 shows the ubiquitous environment managed by EXEHDA.

The EXEHDA-TG aims at providing a solution that: (i) collects contextual information in an automated way within the hospital environment; (ii) processes these contextual information collected, and (iii) provides actuations on the environment and allows alert sending to mobile devices.

The description of EXEHDA-TG, presented in the next sections, is organized based on it servers, corresponding features and the necessary associations between them and with the other middleware services.

Integration Server

The Integration Server is designed to provide the necessary functionality for interfacing with medical devices, enabling the acquisition of data through sensors and actuation. Figure 3 shows the architecture of this server.

The Acquisition Module is responsible for the acquisition and processing of sensed data of infusion pumps and vital signs monitors. This module enables the interface to capture data from sensors of electromedical equipment, considering the communication protocols required for each type of device. The sensors' reading is carried out according to the schedule defined by therapeutic goals. The data collected is treated in stages. In these steps data normalization and the definition of what data must be effectively published in the Assessment Server are performed.

The Publication Module is responsible for coordinating the main data flow between Integration Servers and Assessment Server, promoting the publication of all collected data and ensuring a Local Persistence in the periods when the communication between these servers is temporarily unavailable.

The Actuation Module has the function of triggering devices of electromedical equipment responsible for presenting visual and sound alerts to health professionals, such as liquid crystal displays, buzzers, speakers, LEDs, etc. Therefore, in the EXEHDA-TG messages on the screens of electromedical equipment are considered a specific type of actuation.

Assessment Server

The Assessment Server is responsible for receiving context data of patients and electromedical equipment, storing them and running situational evaluations, as well as requesting actuations and alerts when necessary.

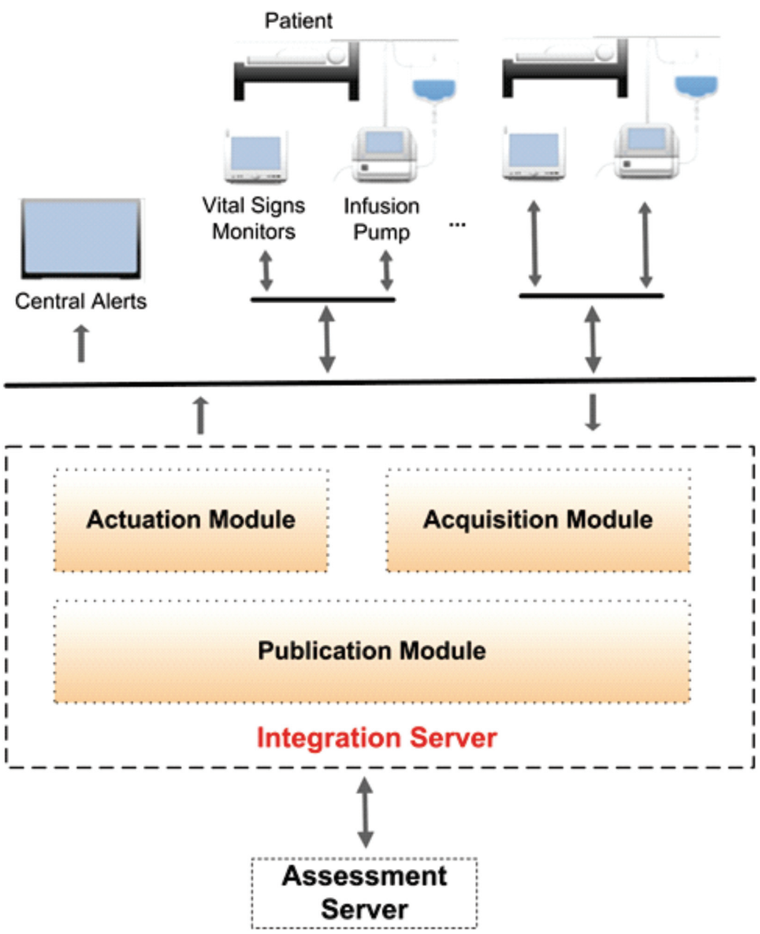

Figure 3. Integration Server Architecture.

Figure 4 shows an overview of this server, presenting its relationship with the Integration Servers, administrators' applications, ubiquitous applications for health professionals, other Assessment Servers, and administrative services of the hospital.

The modules which comprising the Assessment Server are responsible for situation awareness, since the acquisition of contextual data until the activation of actuators and sending alerts. The description of these modules is shown below.

The Configuration Manager, through administrators' applications, provides functionality to perform the setting and management of Assessment Server, considering the characteristics of the sensors and actuators of electromedical equipment. This module stores in a repository: the programming of infusion pumps; definitions of therapeutic goals; and acquisition parameters for electromedical devices.

The Communications Manager is composed of an ESB (Enterprise Service Bus) and a Mobile Access Module, which have the features required to meet the users' distributed applications. The ubiquitous applications of health professionals and other Assessment Servers use the Communications Manager to request vital signs data, status and parameters of electromedical equipment, and to trigger actuators. These ubiquitous applications of health professionals are running on different nodes that comprise the ubiquitous computing environment provided by the EXEHDA middleware.

Aiming at providing mobile access to EXEHDA-TG, a module was designed (see Figure 5), which is organized in two blocks:

(i) Display contextual information: runs on the mobile device and provides reports, both graphical and textual, with the context information. The communication of this block with Assessment Server occurs in two steps: inspection of contextual information being handled; and data request to specific context information in a time interval. The features are organized into three modules: (i) Report Module: handles the display of contextual information on the mobile device; (ii) Access Module: consists on a server with access

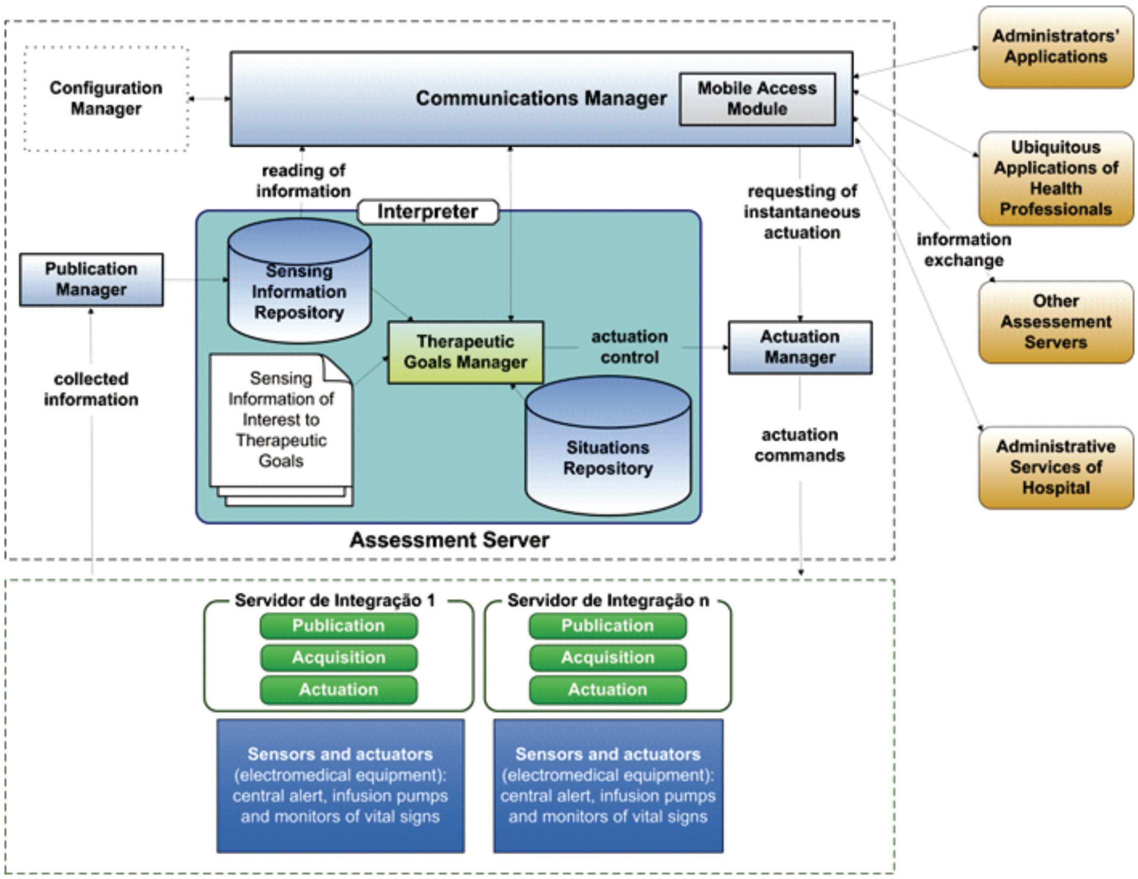

Figure 4. Assessment Server Architecture.

to the Sensing Information Repository of EXEHDA-TG; and (iii) Request Module: aims at requesting context information to the Access Module, considering the Report Module requests.

(ii) Treatment of Proactive Alerts: aims at notifying the user about the occurrence of interesting events through proactive notifications. The features are organized into two modules: (i) Alerts Module: runs on the mobile device, providing alerts to the user, using for it the native notification mechanism of the mobile platform. This option is characterized by providing the user with an integrated management of the nature of the alerts employed by their mobile device; (ii) Distribution Module: runs into the same equipment of the Assessment Server, under uninterrupted operation. This module operates keeping the alerts produced by different contextual rules, handled by the Interpreter. Access from mobile devices occurs through the Communication Module, which provides access to different features of the Assessment Server.

Returning to Figure 4, the Publishing Manager is responsible for supporting the collection of vital signs, status and parameters of infusion pumps and monitors of vital signs, captured through the Integration Server. These features are implemented through the ESB of Assessment Server.

The Actuation Manager is responsible for driving the actuators of electromedical equipment. This component has the function of triggering into the ubiquitous environment actions that change the state of the electromedical equipment, presenting visual and audible alerts to health professionals, whenever requested by the other modules of the Assessment Server.

The Interpreter's main functions are performing management tasks and identification of contextual information, using data specified in the Sensing Information of Interest to Therapeutic Goals component. The identification task is processed on the data collected by the sensors and aims at producing context information to the highest level. The components of this module are shown below.

The Sensing Information Repository stores the contextual data collected by the sensors of electromedical equipment. The prototyped structure reflects the organization of the EXEHDA middleware architecture, covering the relationships among applications, components, sensors, environments and contexts of interest. In this repository are stored patients' vital signs and data sensing of infusion pumps and monitors, obtained by Acquisition Module

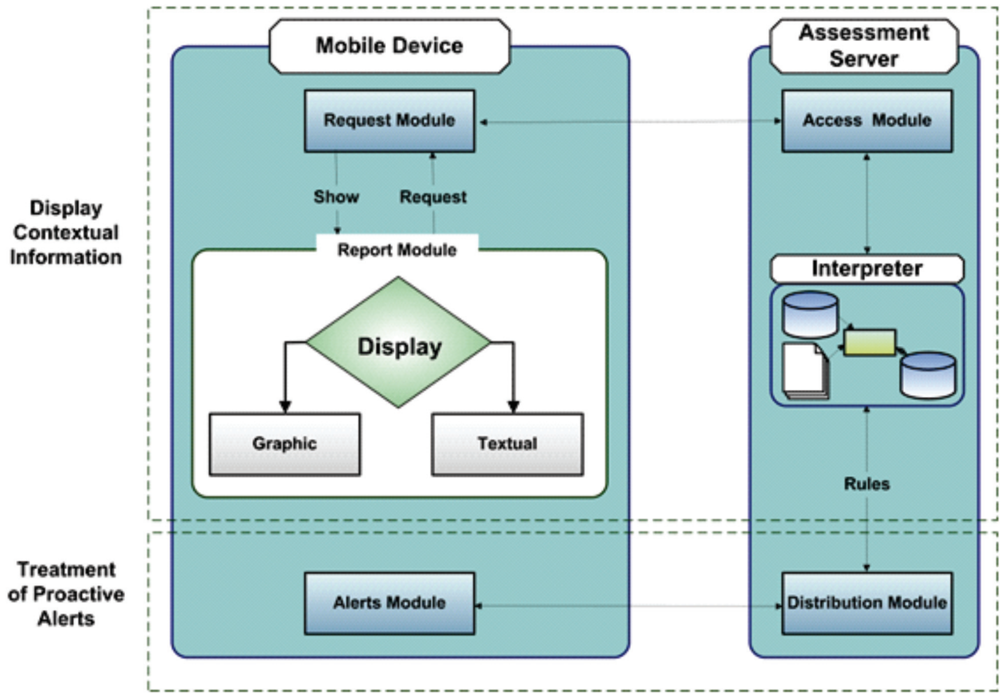

Figure 5. Mobile Access Module.

of Integration Server. Thus, we keep the historical contexts that can be assessed by the Therapeutic Goal Manager. Collected context information is registered in this repository, along with its timestamp collection, timestamp registration and Integration Server source. Information in this repository can be accessed by the ubiquitous applications of health professionals through the Communications Manager.

The Situations Repository stores the results of the evaluations performed by the Therapeutics Goal Manager about contextual information coming from Sensing Information Repository. This repository maintains a history of when and which therapeutic goals were achieved. Information in this repository can be accessed by the ubiquitous applications of health professionals through the Communications Manager.

The Therapeutic Goal Manager is a mechanism based on inference rules, which produces deducted contexts according to therapeutic goals defined by health professionals. The EXEHDA-TG processes, through the Therapeutic Goals Manager, execute rules of type Event-Condition-Action for identifying situations, using techniques based on Temporal and Spatial Logic (Allen and Ferguson, 1994; Ye *et al.*, 2012). The parameters of these rules are set by health professionals, taking into account the range of values and the goal for the vital signs of patients due to the administration of a medicament. When data collected is published in the Sensing Information Repository, the Therapeutic Goals Manager queries the Sensing Information of Interest to Therapeutic Goals component to identify which rules should be evaluated. The results of these evaluations are then registered in the Situations Repository.

Case study

This section presents the technologies employed in the design of EXEHDA-TG, as well as the usage scenarios that were used to evaluate its features. The case study includes sensing and collecting of contextual information, evaluating and notifying of context data to other middleware services.

The infusion pump and the vital signs monitor used in this case study have data communication by Wi-Fi. Consequently, it is possible to parameterize the data of a new infusion remotely, having the reading of vital signs col-

Figure 6. Graphical Interface.

lected, checking the status of sensors/alarms and the infusion progress.

Computing platforms based on Linux were used to provide the acquisition of data through sensors and to trigger actuators in the ubiquitous environment. Integration Server was used the Raspberry Pi Model B, whose main features are: small size, low power consumption, and low cost compared to conventional desktop.

The Assessment Server is responsible for receiving context values collected from patients and electromedical equipment, storing them and performing situational assessments, requesting actuations and generating alerts whenever necessary. Therefore, this case study used a hardware E3400-2.6GHz with Intel dual-core processor, 4GB RAM memory, and Ubuntu Server Operating System.

In this case study two applications were developed with the objective of monitoring and evaluating therapeutic goals: the first is intended for use with Internet browsers; and the second is targeted to the Android platform.

Through the application for Internet browsers it is possible: (i) to use the contextual data collected by the sensors of electromedical equipment and stored in the Sensing Information Repository of Assessment Server, viewing them through graphs and tables; (ii) to display textual reports with data collected from the previous week; (iii) to monitor the evolution of more than one vital sign on the same graph, allowing to compare them; and (iv) to perform statistical analyzes of the data collected.

The first application functionality for Internet browsers is the selection of patients and clinical data of interest to health professionals. Figure 6 shows a graph of the evolution of vital signs. Through it, it is possible to follow the variation of vital signs depending on the medication administered to the patient. By pointing the mouse on the chart, the value, date and time of collection are detailed. It is also possible to view more than one vital sign simultaneously on the same graph, and thus the comparison between them becomes possible.

Through the Web interface of EXEHDA-TG vital signs of a patient collected the previous week are displayed. This information is displayed in columns, one for each day of the week, where the user can also see the average, maximum and minimum values of each day. In Statistical options (see Figure 7) is presented

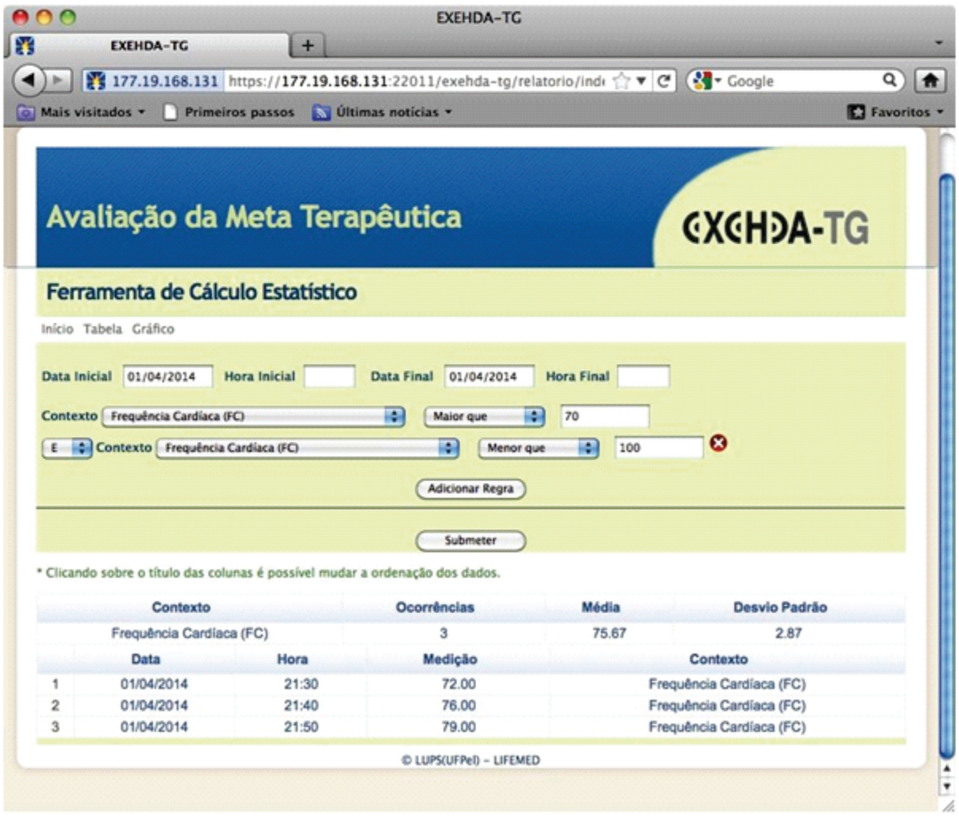

Figure 7. Statistical information.

a feature of the Web application that enables crossing contextual data involving multiple vital signs from different rules. This feature allows the addition, removal and editing of rules and parameters.

The prototyping of Mobile Access Module consists of two parts that interoperate: a mobile application part and the other part, which is processed in the Assessment Server. The mobile part is focused on smartphones and/or tablets that use the Android operating system. In the initial screen of the application it is possible to select a patient, the clinical data of interest and the form (graphical or textual) in which it is desired to be viewed. In the graphical information screen of the mobile application the graph with the evolution of vital signs is presented, where the user can select the desired period of time (an hour, a day, a week or a year). By clicking on the vital sign value curve the date and time of collection are presented. The vertical axis of the graph is adjusted automatically, taking into account the minimum and maximum values to be plotted.

The native notification mechanism of the Android platform was used to provide alerts to the user (see Figure 8). This option provides the user with an integrated management of the alerts on their mobile device.

Related work

The discussion of related work included the following projects: MTM-CC (Koutkias *et al*., 2010), HeartCycle (Chouvarda *et al*., 2011), SIMAp (Leite, 2011) e PA-DSS (Gunawardane *et al*., 2009).

This discussion was based on aspects arising from the assumptions used to design the EXEHDA-TG (see Table 1): (a) distributed or centralized architecture; (b) storage and query of contextual data; (c) support to rules processing; (d) execution of rules based on the knowledge of medical experts; (e) verification of response to medication treatment; (f) interface with electromedical equipment; (g) follow me approach for sending alerts.

Likewise the EXEHDA-TG, the projects studied have distributed architecture, which includes the processing of contextual data in different positions of the architecture. The storage of context data is a feature present in all projects. However, only EXEHDA-TG provides facilities for querying the data collected.

The SIMAp and PA-DSS projects do not consider the evolution of vital signs as a function of medication treatment per patient in a personalized way. The EXEHDA-TG differentiates itself by generating alerts when vital signs are not evolving as expected by the physician at each clinical case.

In MTM-CC project, vital signs are collected through a network of biomedical wireless sensors. In turn, EXEHDA-TG, similar to projects HeartCycle, SIMAp and PA-DSS, uses monitors to collect patients' vital signs. However, EXEHDA-TG differs from these works because it also has support for interfacing with infusion pumps, allowing evaluating and storing the alarms generated by them.

In MTM-CC project, alerts are sent to a central server, while in EXEHDA-TG these alerts are sent to an alert central in the hospital as well as for the mobile devices of health professionals. With this feature, EXEHDA-TG differentiates itself by supporting the follow me approach, which allows the user to start applications and access data from anywhere.

Table 1. Comparison of related work.

Aspects	Projects				
	MTM-CC	**HeartCycle**	**SIMAp**	**PA-DSS**	**EXEHDA-TG**
A	Distributed	Distributed	Distributed	Distributed	Distributed
B	No	Yes	No	No	Yes
C	Yes	No	Yes	Yes	Yes
D	Yes	-	Yes	Yes	Yes
E	Yes	Yes	No	No	Yes
F	No	Yes	Yes	Yes	Yes
G	No	No	Yes	Yes	Yes

Figure 8. Alert on Android Application.

Concluding remarks

The design of an architecture for monitoring the vital signs considering medication proves to be a timely tool to assess response to treatment, and for sending alerts to health professionals when the values of vital signs indicate an unexpected response to the medication.

Thus, by receiving these alerts, the physician may re-evaluate the prescription of medications or prescribe a new one. This alert to the medical staff, stating that the therapeutic goal was not reached, constitutes one of the central contributions of EXEHDA-TG to streamline practices related to the treatment of patients with potential contribution to reducing the length of stay.

Among others, the following aspects should be considered in future work: (i) perform clinical validation in the hospital environment, analyzing the EXEHDA-TG performance in an everyday situation; (ii) explore the use of contextual processing rules using the highest level of inference mechanisms, expanding the options of inference on the collected data; and (iii) continue the integration of EXEHDA-TG with different services and functionalities of EXEHDA middleware.

Acknowledgments

This work has been partially supported by PROPESP/IFSul (PE101213/114) and by FAPERGS (001/2013-PqG).

References

AUGUSTIN I.; YAMIN A.; SILVA L. 2008. Building a smart environment at large-scale with a pervasive grid middleware. *In:* I. AUGUSTIN; A. YAMIN; L. SILVA, *Grid Computing Research Progress*. 1ª ed., New York, Nova Science, p. 323-344.

ALLEN, J. F.; FERGUSON, G. 1994. Actions and Events in Interval Temporal Logic. *Journal of Logic and Computation*, **4**(5):531-579. http://dx.doi.org/10.1093/logcom/4.5.531

BELLAVISTA P.; CORRADI A.; FANELLI M.; FOSCHINI L. 2012. A Survey of Context Data Distribution for Mobile Ubiquitous Systems. *ACM Computing Surveys*, **44**(4):24:1-24:45. http://dx.doi.org/10.1145/2333112.2333119

BETTINI C.; BRDICZKA O.; HENRICKSEN K.; INDULSKA J.; NICKLAS D.; RANGANATHAN A.; RIBONI D. 2010. A Survey of Context Modelling and Reasoning Techniques. *Pervasive and Mobile Computing*, **6**(2):161-180. http://dx.doi.org/10.1016/j.pmcj.2009.06.002

CACERES, R.; FRIDAY A. 2012. Ubicomp Systems at 20: Progress, Opportunities, and Challenges. *IEEE Pervasive Computing*, **11**(1):14-21. http://dx.doi.org/10.1109/MPRV.2011.85

COSTA, C.; YAMIN, A.; GEYER, C. 2008. Toward a General Software Infrastructure for Ubiquitous Computing. *IEEE Pervasive Computing*, **7**(1):64-73. http://dx.doi.org/10.1109/MPRV.2008.21

CHOUVARDA, I.; GKONTRA, P.; KOKONOZI, A.; SEMERTZIDIS, P.; CAFFAREL, J.; MAGLAVERAS, N. 2011. Novel approaches for medication compliance and effectiveness analysis and support in cardiovascular disease patients. *In:*

Engineering in Medicine and Biology Society, Boston, 2011. *Proceedings…* IEEE, p. 888-891.

http://dx.doi.org/10.1109/IEMBS.2011.6090198

GUNAWARDANE, T.S.F.W.; KOGGALAGE, R.; RODRIGO, R.; RAJAPAKSE, S. 2009. Patient alert and decision support system. *In:* Advances in Computational Tools for Engineering Applications, Zouk Mosbeh, 2009. *Proceedings…* ACTEA, p. 656-660.

http://dx.doi.org/10.1109/ACTEA.2009.5227865

KAKOUSIS, K.; PASPALLIS, N.; PAPADOPOULOS, G. 2010. A survey of software adaptation in mobile and ubiquitous computing. *Enterprise Information Systems*, **4**(4):355-389.

http://dx.doi.org/10.1080/17517575.2010.509814

KNAPPMEYER, M.; KIANI, S.; REETZ, E.; BAKER, N.; TONJES, R. 2013. Survey of Context Provisioning Middleware. *Communications Surveys Tutorials, IEEE*, **15**(3):1492-1519.

http://dx.doi.org/10.1109/SURV.2013.010413.00207

KOUTKIAS, V.G.; CHOUVARDA, I.; TRIANTAFYLLIDIS, A.; MALOUSI, A.; GIAGLIS, G.D.; MAGLAVERAS, N. 2010. A Personalized Framework for Medication Treatment Management in Chronic Care. *IEEE Transactions on Information Technology in Biomedicine*, **14**(2):464-472

http://dx.doi.org/10.1109/TITB.2009.2036367

LEITE, C.R.M. 2011. *Arquitetura Inteligente Fuzzy para Monitoramento de Sinais Vitais de Pacientes: Um Estudo de Caso em UTI.* Natal, RN. PhD Thesis. Universidade Federal do Rio Grande do Norte, 136 p.

LOPES, J.; SOUZA, R.; PERNAS, A.; COSTA, C.; BARBOSA, J.; YAMIN, A.; GEYER, C. 2014. A Middleware Architecture for Dynamic Adaptation in Ubiquitous Computing. *Journal of Universal Computer Science*, **20**(9):1327-1351.

WEISER, M. 1991. The Computer for the 21st Century. *Scientific American*, **3**(265):94-104.

YE, J.; DOBSON, S.; MCKEEVER, S. 2012. Situation identification techniques in pervasive computing: A review. *Pervasive and Mobile Computing*, **8**(1):36-66.

http://dx.doi.org/10.1016/j.pmcj.2011.01.004

Submitted on October 24, 2014
Accepted on December 9, 2014