# An Integrated Development Environment for Building Semantic Context-Aware Applications

*Taciano P. A. Alcântara*[2], *Nazim Agoulmine*[4], *Anilton S. Garcia*[3], *Raquel Silveira, Luiz O. M. Andrade*[2]
*Leonardo M. Gardini*[1], *Mauro Oliveira*[1]

[1]*State University of Ceará (UECE) - Fortaleza, Brazil*
[2]*Federal University of Ceará (UFC) - Fortaleza, Brazil*
[3]*Federal University of Espírito Santo (UFES) - Vitoria, Brazil*
[4]*Université d´Evry-Val-D´Essonne - Evry, France*

taciano@ufc.br, nazim.agoulmine@ufrst.univ-evry.fr, odorico@saude.gov.br, anilton@inf.ufes.br, lgardini@gmail.com, amauroboliveira@gmail.com

***Abstract*—Context-awareness is an important feature for e-health applications since it allows taking into account users's situation and environment in the decision-making. However, the development of such applications introduces another level of complexity that often complicates their development and maintenance. This paper proposes an Integrated Development Environment (IDE) to assist developers in implementing e-health context-aware applications and enriching them with semantic information using ontologies about contexts. This work presents the architecture, specification and prototyping of the proposed IDE as well as the results obtained in Lariisa, a project about governance and decision-making in e-health systems.**

***Keywords—IDE; context-awareness; semantic; ehealth; system; decision-making***

## I. Introduction

Context-aware Systems [8][16] have proved to bring positive results to support advanced management and decision-making, including empowering health services [21]. This type of system is capable to process contextual information, in other words, information that characterizes a given situation where a user is and supports the offering of intelligent services.

However, the development of context-aware applications is highly complex and the development process involves the use of knowledge representation concepts, inferences, among others [1]. Moreover, these applications may interact with various and potentially heterogeneous context data sources (context providers), such as fixed and wireless sensors, mobile devices, set-top boxes, etc., that may increase the complexity of the development process and therefore increase the development costs.

The objective of this paper is to propose an Integrated Development Environment (IDE) to model knowledge using ontologies and implement applications that use this knowledge to provide context aware services. The proposed environment, a powerful contextual representation tool, has several features that are mandatory such as maintaining knowledge databases, setting rules and actions, information management, integration with context providers, etc. These features are unified in a single environment to support the development of context-aware applications.

The resulting IDE is used in the context of the Lariisa project context [13], a governance and decision-making framework for public e-health care systems. A case study for the proposed IDE is a simple application to monitor dengue cases an infectious tropical disease caused by the dengue virus. [3].

This paper is organized as follows: Section II describes the concepts of context-aware systems and ontologies. Section III presents related work. Section IV shows the architecture, specification and prototyping of the proposed IDE. Finally, Section V concludes the paper and discusses future work.

## II. Context and Ontologies

### *A. Context and Context-aware Computing*

With the wide deployment of mobile devices, such as smartphones and tablets, users are performing more and more tasks on the move with these devices. During these journeys, mobile devices are capable to gather numerous valuable information about the user behavior such where the user is or what the user is doing, and this information can be used to offer personalized services and information. This type of information, which characterizes a situation of the user, can be used by context aware application to customize their services and make them more accurate to the user profile and situation [2],

Aiming in assisting users in their day-to-day tasks, context-aware applications can interact with ubiquitous systems to collect context information about them. A simple example of such system is the use of sensors that detect the presence of people and automatically switch on lighting in a room according to the location and time.

Three important aspects of context are defined as information related to where a user is, with whom he is, and what resources are available in the surrounding [16]. These aspects may change continuously, and a huge quantity of information could be derived from them, such as: luminosity, noise level, network connectivity, communication costs, bandwidth, social status, etc.

The user context needs to be modeled using some semantic technique. A context model defines types, names, properties and attributes of the entities involved in context-aware applications, such as users, the environment, the devices, time, and so on. The model attempts to predict representation, search, exchange and interoperability of context information among applications. A well-designed model is the key to context-aware system [6].

### *B. Ontologies*

The amount of information available on the web is increasing every day. More and more people are connecting on the internet, creating and sharing information. However, traditional information recovery does not reflect the data semantic, their relationships and the knowledge that they represent. To have a sustainable growth, there is a need to adequately manage this huge mass of information. Semantic Web helps computer to understand the meaning of information stored/transmitted in on the web [4].

Building an application based on semantic web needs the creation and implementation of technology standards to establish semantic concepts that make possible sharing information between two or more systems. It is necessary to create mechanisms that are capable to describe the meaning of data and specify its encoding. One of these mechanisms is defined using ontologies.

Information science studies the phenomenas related to the information in its various aspects - trying to understand and follow its outspread. By doing that, in the future, information can be available and managed by computer systems. Using ontologies, especially in the computer science area, makes possible the communication between people and computer systems that participate in the same knowledge field - but not necessarily share the same conception of elements of this domain.

An important motivation to use ontologies is the garantee of reliability surrounding vocabulary concepts or languages that are used in certain environments. Thus, using formal representation, consistency verification automation of application becomes possible making them more reliable.

## III. RELATED WORKS

A Context-Aware Web Content Generator Based on Personal Tracking [4] is a context-aware system for the creation, annotation and sharing of multimedia content.

There are several applications that use context information to enrich and organize multimedia documents. This information might be the proximity of people or objects in a photo, the current temperature, the date/time, etc. This type of metadata can be already obtained from embedded sensors of mobile devices or from the web. With this added information (tags), context-aware applications can better organize the multimedia objects, providing user-friendly visualization of the content, and suggesting annotations for document indexation [9][10][11].

Figueiredo [25] proposed an infrastructure for development context-aware applications, but he not consider semantic data. His project focus is creating rule-based trigger for context sense. Other scenario [26] proposed a model driven development method for developing context-aware pervasive systems, building system functionality and context information together, with high level of abstraction.

The ubiquitous computing paradigm addresses mobile networks and distributed systems and demand for fast and reliable systems that are accessible anywhere and at anytime [17]. Ubiquitous systems must be able to execute in at least two environments: (i) volatile environments in which mobile devices suddenly appear and disappear; and (ii) dynamic environments in which the network topology, their components and services are constantly changing [18].

These infrastructures provide mechanisms to collect, process and share context information used to infer user's situation. In our research, we realized that a common solution for ubiquitous systems coordination can be based on shared memory space and event-based models [19].

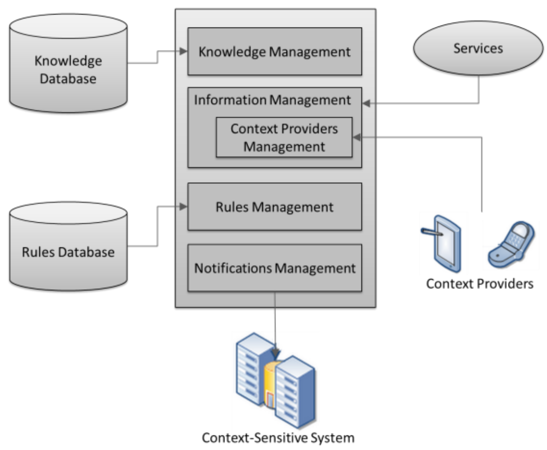


Fig. 1. IDE Architecture

## IV. BUILDING CONTEXT-AWARE APPLICTIONS BASED ON ONTOLOGIES

This Section presents the proposal of Integrated Development Environment (IDE) to help developers to design and implement context-aware applications using ontologies. The following sections present the general architecture of the IDE and its various modules.

### *A. Architecture*

The architecture of the proposed IDE is composed of a set of modules addressing differerent aspects of the context-aware applications development (Figure 1).

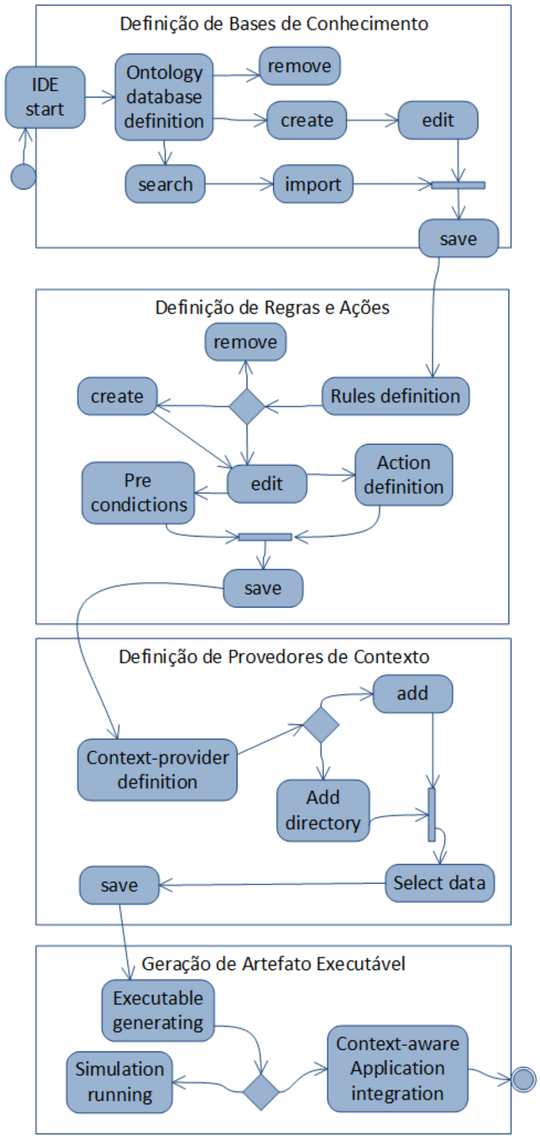


Fig. 2. Activity Diagram of the proposed IDE

The main flow to develop a contex-aware application starts with database knowledge definition, followed by rules and actions definitions, context providers integrations e executabe artfact generating.

In the knowledge database definition the developer can create (or import), remove and edit ontologies. In rules and actions the developer can create, remove or edit rules with his pre-condictions and actions. In the context providers definition the developer can add and select context providers or context providers directory. Finally, the user can make an executable context aware aplication.

### *B. Modules Description*

Figure 3 shows the main screen of the proposed IDE. It contains several shortcuts to access the main modules of the IDE.

#### *1) Knowledge Database*

Knowledge databases are used to model context representation. Indeed, a well defined model is crucial to capture the available concepts of the real environement. As shown in Figure 1, the Knowledge Management module permits to use several tools to represent knowledge in the knowledge database. These tools allow to read the available ontologies and to reason on them (inferences). It is possible to search for specific ontologies in the database using specific tools provided by the environment – just using key-words.

The IDE is capable to interact with a variety of ontologies based databases available in the Internet (like SchemaPedia [22], Swoogle [23], Protégé Ontology Library [24]). The developer has an option to choose one or more available ontologies in a search list, and to import them for his/her own application. The developer can add or remove new databases depending on his needs.

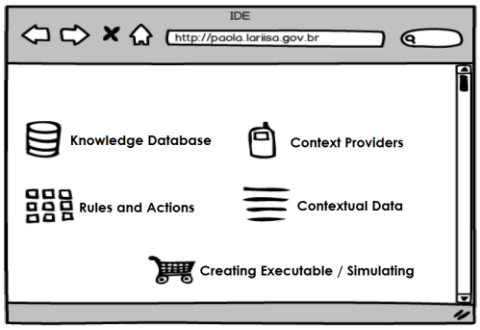


Fig. 3. Home Screen Prototype

The developer may also create new ontologies. The IDE provides a simple OWL (Ontology Web Language) editor to allow him to edit the ontologies without leaving the environement.

If the developer requires a more developped one, he/she can use third party softwares, e.g.: a graphic editor of ontologies [15] to edit an ontology, and then import it to the IDE editor to continue his work.

#### *2) Context Providers*

Context providers are agents able to collect contextual information and send it to a context-aware application or a context server. Context providers make use of raw data that will be analysed to represent contextual information.

Context providers are devices capable to provide any contextual information. As an example, smartphones or cell phones with GPS are able to provide geolocation information about the user position as well as his phone number.

IDE provides an interface to the developer to manage context providers, including features like: add context providers; add directory context providers; select context providers information; select directory information from context providers; remove context providers; and remove context providers directory.

It is also possible to add a group of context providers by just typing the address of the providers' directory. This directory has a number "N" of similar devices able to provide the same type of information. For example: temperature sensors installed in an environment providing at regular time the local temperature.

It is therefore possible to provide this information to a context aware application in the IDE using a directory that represents all of these sensors – because all of them will provide the same type of information: temperature.

*3) Rules and Actions*

The Rule and Action Management module checks, during each context change, whether a rule was assigned or not. In the positive case, it triggers the corresponding actions. With this module, the developer can define how a rule is checked and how to manage the corresponding Actions.

A Rule represents a specific feature that the system infers when a contextual information is received. An example of rule related to the health area is the diagnosis of dengue infection [3]: the system detects a possible infection of a patient if he/she has at least three symptoms of dengue and lives in an area where there is a focus of dengue cases. Preconditions are assumptions to satisfy any rule, and, when a rule condition is satisfied, it results in an action trigerring.

Action is one of the main features of a context-aware system – it is where a system reacts (thourgh actions) to changes in the context, without having necessarily a human intervention.

Action Management is called by the Rules Management (both in the same module) where some rule is reached and step in to notify a context-aware application about changes in the context. This module definies the relationship with Actions Management and how works the notification to the application.

*4) Contextual Data (Information)*

This module uses specialized techniquues to run queries in ontologies and inferences. This module also defines how contextual data will be provided by context providers.

After selecting context providers, these context providers are able to send contextual information. The storage and recovery of contextual information is the responsability of the Information Management Module.

Features from the user point of view: consult data stored on the contextual database; change data stored in the contextual database; and remove data stored in the contextual database. Services offered by this module require that a user (e.g. the developer) understand the data query language or that the integrated systems implement a query API.

*5) Executable and Simulation*

After configuring the knowledge databases, rules, actions, context providers and how to access the information, the developer can generate/create an executable of the project. This executable is linked to a context-aware application and it is responsible to collect contextual information, process it, and add notifications to the application that is assigned to receive them (the notifications).

The developer can also use the executable module to simulate an operation within the IDE. The environment still offers mechanisms to emulate a context-aware environment with fictitious context providers and context-aware applications.

The executable created by this module is the output of the IDE. Using this executable an application can work with context features, interacting directly with the executable to establish communication with the sensor, knowledge database, notifications, etc. It hides the complexity of the application in the interacting with the context providers and the heterogeneity aspects to address..

## V. PROTOTYPING IMPLEMENTATION

### *A. Knowledge Database Management Module*

The proposed IDE uses ontologies languages such as OWL, RDF and RDFS to manipulate knowledge in the databases with Jena [12], a Java framework designed for semantic web, used to represent knowledge and handle ontologies through defined classes, entities, properties, etc. supports the creation of inference rules.


Fig. 4 – Implementation aspects of the Knowledge Database Management Module

The developer can also import ontologies directly from the web and include them in IDE. The developer can also edit ontologies directly in the platform and all manipulation of the ontologies concepts via graphic interface will be reflected in the knowledge database through the Jena Ontology API.

Jena was chosen because it is a very popular framework to semantic web and works with object-oriented programming.

### B. *Context Provider Management Module*

A strategy was developed [1] to reflect contextual data in the knowledge databases. A consulting adapter (query adapter) was designed to translate information from WebServices (LISA[7] integration) to perform queries based on HTTP protocol. This required to standardize the way that information in the knowledge databases is derived the information provided by the various context providers (this is performance using LISA) [7].

The Joseki (an open source tool that runs over a HTTP server, supports GET and POST operations and translates queries requests to the SPARQL language – used to perform queries on knowledge databases) was used to perform queries in the knowledge database. An overview of the Context Providers Management Module implementation is presented in Figure 5. Figure 6 shows an example of insert data from context provider to database using SPARQL query. In this example, information of dengue focus and his localization are stored in database.

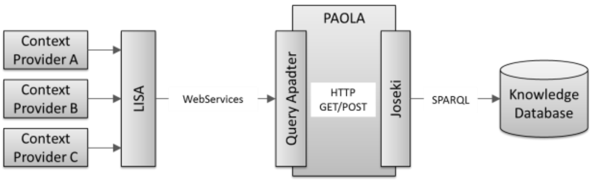


Fig. 5. Implementation aspects of the Context Providers Management Module

### C. *Information Management Module*

This module supports the insertion of data coming from context providers and the queries that context aware application can trigger on the knowledge databases. Applications (or external services provided by applications) may manipulate knowledge database information directly using the HTTP protocol – without using any query adapter

The Information Management Module also uses the Joseki to perform queries on the knowledge database (as well as the implementation of the Context Providers Management Module). Similar queries, like Figure 6, are written for search contextual data.

```
PERFIX ld:
<http://lariisa.gov.br/ld/elements/1.1/>
INSERT DATA
{ <http://example/larissa/dengue>
ld:localization “07°13'06"S-39°16'18"W” ;
ld:infection “yes” .
}
```

Fig. 6. Example of SPARQL query

### D. *Rules Management Module*

To manage the rules, the developer uses the Rules Management Module graphical interface of the proposed IDE. The rules are represented using a Jena’s framework module: a general-purpose rule engine. It makes possible to perform inferences on a knowledge database or detect changes of context to trigger a set of actions.

Rules are stored in files following the syntax defined by Jena and the rules are handled through an API.

Figure 7 show an example of rule created in IDE for contextual analysis to Dengue symptoms in a city (or other geographic region).

```
(hasSintoma min 3 SintomaDengue) and
(hasLocalizaçãoEspacial in São Paulo)
```

Fig. 7 – Example of rule in IDE

### E. *Actions Management Module*

The proposed IDE uses the object-oriented pattern Observer to manipulate actions. This pattern allows that interested systems are advised about any change in the condition of any rule defined by the developer.

With this module, the developer can operate as follows: Define which rules will trigger which actions; Register applications (listeners) to receive actions.

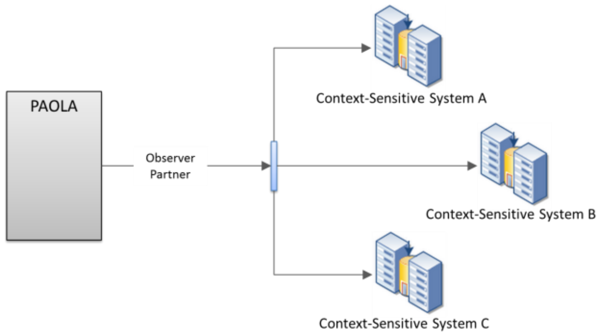


Fig. 8. Implementation aspects of the Actions Management Module

Figure 8 shows a use case where three context-aware applications/systems (listeners) are connected to the proposed IDE, and ready to receive notifications. The IDE provides an API, regardless of the programming languages of the context aware applications/system, allowing them to receive any information that are interested in to adapt their service to the situation of the users.

## VI. CONCLUSION

The focus of this paper was to specify and implement a Proof of Concept prototype of an Integrated Development Environment (IDE) for context aware applications.

The integration of the proposed IDE modules (Knowledge Database, Context Providers, Actions & Rules, Information Management and Creating Executable & Simulating), allow to create a development environment that facilitate the process of design, development and text of context-aware applications.

Considering all these features, the proposed IDE has a great potential to facilitate the development of new context-aware applications in any environment and particularly in the area of e-health if associated with proper medical sensors [20].

The prototype has been used on the Lariisa project in two ways: the first relates to geo-referencing for context-awareness [4]. The second relates to migrating a software platform that will allow the publication and integration of open data related to public health in a cloud environment [14].

As a future step is suggested work in the usability sense of system and better integration of modules.

## REFERENCES


[1] Alcântara, T. P. A. (2012) "Paola: Uma Plataforma para Desenvolvimento de Aplicações Baseadas em Ontologias para o Projeto Lariisa", Dissertação (Mestrado Profissional Computação Aplicada), Universidade Estadual do Ceará, Brasil.

[2] Wang, X. H., Zhang, D. Q., Gu, T. and Pung, H. K. (2004) "Ontology Based Context Modeling and Reasoning using OWL" In: Proceeding of the Second IEEE Annual Conference on Pervasive Computing and Communications Worshops, Singapore.

[3] Antunes F. (2011) "Sisage: Um componente do Lariisa de Gestão e Vigilância Epidemiológica em instâncias de agravo Dengue no Estado do Ceará", Dissertação (Mestrado Profissional em Computação Aplicada), Universidade Estadual do Ceará, Brasil.

[4] Braga, r. B.; costa, s. M. M.; viana, w.; andrade, r. M. C.; martin, h. "A Context-Aware Web Content Generator Based on Personal Tracking", 2011

[5] Berners-Lee, T., Hendler, J. and Lassila, O. (2001) "The Semantic Web" In: Scientific American.

[6] Biegel G. and Cahill V. (2004) "A Framework For Development Mobile, Context-Aware Applications" In: Second IEEE Annual Conference On Pervasive Computing And Communications, Dublin, p. 361-365.

[7] Bezerra, joão batista; Oliveira, a. M. B. ; Andrade, o. M.; Barreto, i.; Moura filho, c. O. . Integrating Mobile Devices In a Brazilian Health Governance Framework. In: International Conference on Advances of Information & Communication Technology in Health Care (ICTHC 2011), 2011, Jakarta (Indonésia). International Conference on Advances of Information & Communication Technology in Health Care (ICTHC 2011). Jakarta (Indonésia): ACEEE Indonesia Sectio, 2011.

[8] Dey, A. K. (2001) "Understanding and Using Context" In: Personal and Ubiquitous Computing, UK, Springer Verlag London Limitek, p. 4-7.

[9] Viana, W., Miron, A., Moisuc, B., Gensel, J., Villanova-Oliver, M., Martin, H.: Towards the semantic and context-aware management of mobile multimedia. Multimedia Tools and Applications (2010) 1{39 10.1007/s11042-010-0502-6.

[10] O'Hare, N., Smeaton, A.F.: Context-aware person identication in personal photo collections. Trans. Multi. 11 (2009) 220{228

[11] de Figueir^edo, H., Lacerda, Y., de Paiva, A., Casanova, M., de Souza Baptista,C.: Photogeo: a photo digital library with spatial-temporal support and self-annotation. Multimedia Tools and Applications (2011) 1{27 10.1007/s11042-011-0745-x.

[12] Jena, A. (2012), "Apache Jena: Java framework for building Semantic Web Applications". Disponível em: http://jena.apache.org/. Acesso em 16 de dezembro de 2012.

[13] Oliveira, M., Hairon, C., Andrade, O., Moura, R., Sicotte, Denis, J-L., Fernandes, S., Gensel, J., Bringel, J. and Martin, H. (2010) "A Context-Aware Framework For Health Care Governance Decision-Making Systems" In: IEEE International Symposium on A World Of Wireless Mobile And Multimedia Networks, Canada.

[14] Oliveira, M. (2012) "Cloud Lariisa, A Platform for Data Integration of Public Health Systems in Cloud Computing Environment" In: 1st Workshop on ADVANCEs in ICT Infrastructures and Services, Brasil.

[15] Protégé (2012), "Ontology editor and knowledge-base framework base framework". Disponível em: http://protege.stanford.edu Acesso em 16 de dezembro 2012.

[16] Salber, D., Dey, A. K. and Abowd, G. D. (1999) "The Context Toolkit: Aiding the Development of Context-Aware Applications" In: Proceedings of the SIGCHI conference on Human Factors in Computer Systems, USA, p. 434-441.

[17] A. Corradi, E. Lodolo, S. Monti, and S. Pasini, "Dynamic reconfiguration of middleware for ubiquitous computing," in Proceedings of the 3rd international workshop on Adaptive and dependable mobile ubiquitous systems. ACM, 2009, pp. 7–12.

[18] M. E. Maia, L. S. Rocha, and R. M. C. Andrade, "Requirements and challenges for building service-oriented pervasive middleware," conference on Pervasive services, pp. 93–102, 2009.

[19] G. Abri, L. Ferrari, M. Mamei, and F. Zambonelli, "Uncoupling coordination: Tuple-based models for mobility," In Mobile Middleware. Taylor and Francis CRC Press, London, U.K., pp. 229–256, 2005.

[20] N.Agoulmine, J.Deen, J.Lee, M.Meyyappan "UHealth Smarthome - Innovative solutions for the management of the elderly and chronic diseases.," IEEE Nanotechnology Magazine 1932-4510/11/$26.00-2011IEEE", September 2011. On-line: lien.

[21] A. D. Wood, J. A. Stankovic, G. Virone. "Context-Aware Wireless Sensor Networks for Assisted Living and Residential Monitoring" IEEE Network (2008)

[22] Schemapedia http://schemapedia.com/

[23] Semectic web search http://swoogle.umbc.edu/

[24] Protege Ontology Library http://protegewiki.stanford.edu/wiki/Protege_Ontology_Library

[25] H. F. Figueiredo. Uma Infraestrutura de Suporte a Aplicações Cientes de Contexto com o Enfoque no Usuário Final (2009). Dissertação de Mestrado (Mestrado Acadêmico em Ciência da Computação). Universidade Federal de Campina Grande, Campina Grande, Brazil.

[26] E. Serral, P. Valderas, V. Pelechano. Towards the Model Driven Development of context-aware pervasive systems.