Journal of Universal Computer Science, vol. 24, no. 12 (2018), 1758-1775
submitted: 8/3/18, accepted: 15/11/18, appeared: 28/11/18 © J.UCS

A Cross-Device Architecture for Modelling Authentication Features in IoT Applications

Darwin Alulema
(Universidad de las Fuerzas Amadas, Sangolquí, Ecuador
doalulema@espe.edu.ec)

Javier Criado
(Applied Computing Group, University of Almería, Spain
javi.criado@ual.es)

Luis Iribarne
(Applied Computing Group, University of Almería, Spain
luis.iribarne@ual.es)

Abstract: The Internet of Things has presented a rapid development, due to the over-crowding of hardware and software platforms, greater deployment of communications networks, development of data analysis tools, among others. This development has led to a boom in applications focused on areas as varied as Smart Cities, Smart Agro, Smart Buildings, Smart Home, and Smart Health, in which people and things are interconnected. This is one of the reasons by which a review of the main technologies involved in the emergence of the Internet of Things must be carried out to determine those characteristics allowing that interconnection, but without neglecting security. This issue allows the user to feel con dent to use these new services. In this work, we propose a cross-device architecture that integrates technologies and implementations in homes, and uses basic authentication as a security scheme. To validate the cross-device proposal, a case study scenario has been designed, including and integrating digital-TV (DTV), Smart Phones and wearables devices for monitoring users physical activity.

Keywords: Model engineering, Digital TV, T-Health, Internet of things, Security.
Categories: B.4.2, D.2.10, I.6.5

1 Introduction

The rapid development of technology has allowed its inclusion in the daily life of people, promoting the development of smart devices and management platforms, which allows a "Smart World". One of the main problems for the development of this intelligent world is interconnection. This should be simple and must allow the creation of applications that run on heterogeneous devices [Mainetti, L. et al. 2015]. In addition, software systems should be easy to use and instinctive, but the biggest challenge for software engineers is how to develop software with high-reliability [Hsu, C. and Huang, C. 2018].

In an interconnected world, the systems that combine physical and computer components to create complex behaviors are ubiquitous and increasingly complex. This has created new potential security problems, which require the application of new techniques in both design and execution time. Hence, security and safety are the

main challenges due to the serious consequences of the attacks. For this reason, at the design stage, the risks derived from accidental events in the operation of the systems must be taken into account, as well as the risks derived from an attacker to the system [Wolf, M. 2018]. Possible attacks against the Internet of Things fall into three main categories based on the objective of the attack: a) Attacks against a device, b) Attacks against communication between devices and servers, and c) Attacks against servers [Jeyanthi, N. and Thandeeswaran, R. 2017]. However, the delivery of software with errors as a critical security problem should also be considered, because it can become a gateway for attacks on the system [Mishra, D. and Mishra, A. 2004].

When considering interoperability there exists the emergence of two main problems [García, C. G. et al. 2014]: the complexity of software development and the fragmentation of platforms. As a solution to this problem, the concepts of "multiplatform" (or cross-device) [Ribeiro, A. and Rodrigues, A. 2014] and Model-Driven Architecture (MDA) [Benouda, H. et al. 2016] can be applied to reduce the difference in the scope of the problem and the implementation of software. That is why the systems can be described to one level that allows developers the reusability of programming processes, underlying technology and facilitates the interoperability with other external systems [Troya, J. et al. 2013]. However, it is also necessary to establish a test protocol to detect errors during the software development process [Mishra, D. and Mishra, A. 2004], so that software suppliers can know if their products are reliable before being put on the market [Ding, Z. 2016].

In this paper we propose a cross-device architecture based on models, applied to the Internet of Things (IoT), with a basic authentication scheme. In addition, a concept test bench is proposed, with a T-Health system. This paper is structured as follows. Section 2 reviews some related work. Section 3 defines the proposed architecture and model and Section 4 describes the developed test platform. Finally, Section 5 outlines the main conclusions and future work.

2 Related work

The main tool of the evidence-based paradigm is the systematic review of the literature, which provides a framework for the systematic search of the literature [Bachy, Y. et al. 2015], allowing categorizing, classifying and performing thematic analysis. This technique has been applied to the authentication mechanisms and the architectures of IoT platforms. For the Systematic Review process we have proposed the following questions:

- RQ1: What are the platforms used for the IoT?
- RQ2: What mechanisms are used for authentication
- RQ3: What technology exists for the interconnection between platforms?
- RQ4: What technology exists for software modeling?

The search engine selected to perform the document search was the Scopus database. This search engine indexes several catalogs of publications IEEE Xplore, Science Direct, Springer Link, among others. During the search, all those articles published between 2004 and 2018 have been taken into account. The studies included

must meet the following conditions: a) Complete articles, b) That belong to the "Computer Science" branch, and c) Written in English.

The process of classifying the articles was carried out in two phases: (a) first, the search of all the articles with the three search chains was carried out. In this first filtering, only Title + Abstract + KeyWords were considered, obtaining 342 articles; (b) in the second filter, only the articles of interest that truly contribute an architecture were considered. The final set was reduced to 28 items of interest.

There are many platforms for IoT, some of the most popular: Amazon AWS, ARM Bed, Microsoft Azure IoT, Google Brightness / Weave, Calvin Ericsson, Apple HomeKit, Eclipse Kura and Samsung SmartThings, which are used for the development of smart applications [Ammar, M. et al. 2018]. The convergence and integration of these platforms can be resolved with the design of service-oriented architectures [Bosin, A. 2012]. One of these integrations is developed in [Luo, S. et al. 2014] that proposes a cloud system for medical monitoring. This convergence increases the use and potentially connects billions of devices that could generate a large amount of data at a very high speed and some of the applications may require a very low latency. Hence, some proposals are based on edge computing, to reduce the response time of the system, passing the task of preceding the information to the edge devices [Pan, J. and McElhannon, J. 2017].

The exponential increase in security flaws could present serious threats to the health and safety of patients using these technologies, due to the use of information in systems that must respect the confidentiality and privacy of medical information [Jeyanthi, N. and Thandeeswaran, R. 2017]. One of the reasons why these failures may occur is the limitation that many small and medium organizations have of implementing software quality assurance models that they develop [Mishra, A. and Mishra, D 2006]. In addition, organizations increasingly distribute their software development processes, facing new technical, social and cultural challenges [Mishra, D. and Mishra, A. 2011]. That is why identification is crucial to control access to the service and establish trust mechanisms between the object and the service in the cloud. Some of these proposals are based on pattern recognition [Hu, P. et al. 2018]. Other applications propose anonymous authentication schemes based on the use of anonymous certificates, for when the communication is direct between the nodes of an IoT network [Vijayakumar, P. et al. 2017].

Smart Health is a type of application that is very popular for its immediate usefulness in the wellbeing of people, with the particular case of telemedicine, which allows medical processes automation [Hossain, M. et al. 2017]. Some proposals are based on user authentication and encryption of the information generated by the devices [Rivas, C. et al. 2015]. In addition, to guarantee security in these systems, it is necessary to ensure the integrity of the information, which may be affected by the identity impersonation of the transmitting entity [Bachy, Y. et al. 2015]. In [Avila, K. et al. 2017] the authors propose a SOA-based interoperability to standardize services and create scalable architectures.

There is a need to consider security in IoT systems, because everything is interconnected, each node of a network becomes a possible victim of an attack. In addition, even when working with different platforms, digital TV has not received much attention, considering that it is the most widespread household appliance. For this reason, an architecture is proposed for the development of applications and allows

that security is not lost sight of, in any of its stages. This is intended to achieve reliability in the information, since the platforms would be interoperable.

3 Semantic based retrieval using meta data

One of the areas of the Internet of Things in which the need for interoperability is observed is the Smart Cities. In which there are multiple management platforms, but not all are interoperable [ONTSI 2016]. To address the problem of interoperability, we have considered technological platforms. For this, two phases have been determined: 1) Identification of reference points of each platform, and 2) Definition of an architecture that groups the characteristics of the platforms into categories.

3.1 Phase 1: Identification of reference points

To identify the reference points, the Digital TV, Smart Phone, Smart TV and wearables platforms have been considered to determine the characteristics of the IoT nodes. Some requirements that platforms must meet have been identified:

a) *Communication between systems:* a layered architecture allowing platforms to establish mechanisms for interconnection, such as the one proposed in the OSI model.
b) *Security:* platforms must allow the implementation of security controls, such as layer security, lifecycle controls for devices, authentication framework, among others [Ammar, M. et al. 2018].
c) *Application development:* because there is a wide variety of development tools in the market, it is essential to use technologies common to all platforms, such as HTML5, JavaScript or Java.

As digital TV is one of the platforms that has had the least development in the IoT, we propose a new reference for this integration with the acronym IoTV (*i.e.*, IoT + TV). The standard selected has been ISDB-T, which is in the implementation stage in the majority of Latin American countries. This standard defines recommendations for every stage of the system shown in Figure 1. However, these have not been developed for the purpose of integrating into the Internet of Things. In security, the norm that regulates it is NBR15605, which only proposes the control of copies of content to combat piracy, without establishing security plans for systems that will have large volumes of data and customers [ABNT 2004].

3.2 Phase 2: Definition of an architecture and meta-model

To define the architecture and the metamodel, a scenario has been considered (Figure 2), in which multiple devices (digital TV, Smart Phone and wearables) interact to provide a reliable service. In the scenario, the reactions that users have with their wearables devices (IoT nodes) and the applications provided are identified. The provider, in addition to offering the applications to the user, defines how their services work and the means by which they are distributed. In [Alulema, D. et al. 2017] a first approach to the model is presented. The proposed architecture meets the following requirements [Mainetti, L. et al. 2015]:

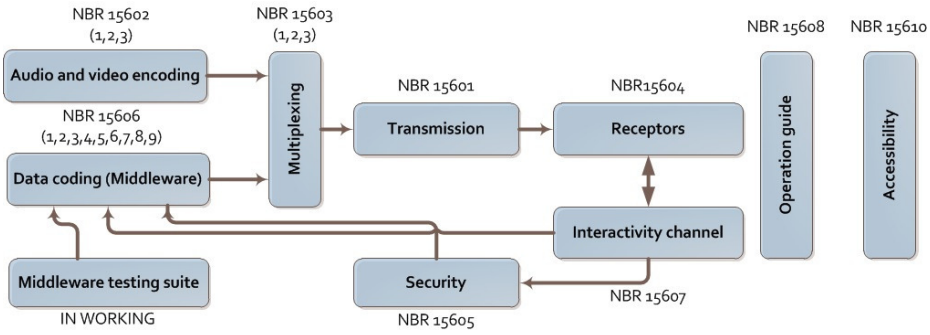

Figure 1: Technical standards for the digital TV ISDB-T standard [ABNT 2004]

a) *Reduce barriers to entry:* The proposed architecture must use technologies common to all platforms. This allows a greater number of developers to not have to learn a language for each platform.
b) *High usability of the architecture:* The layering of the architecture allows a greater specialization of the suppliers. This makes it easy to implement redundant and ubiquitous network environments so that the user can access resources anytime, anywhere.
c) *Control of the devices:* The architecture based on services, allows applications to easily access resources, according to the needs of the user.
d) *Low computational load:* The architecture allows the user to reduce the consumption of computational and memory resources because it allows access to the service from external servers.

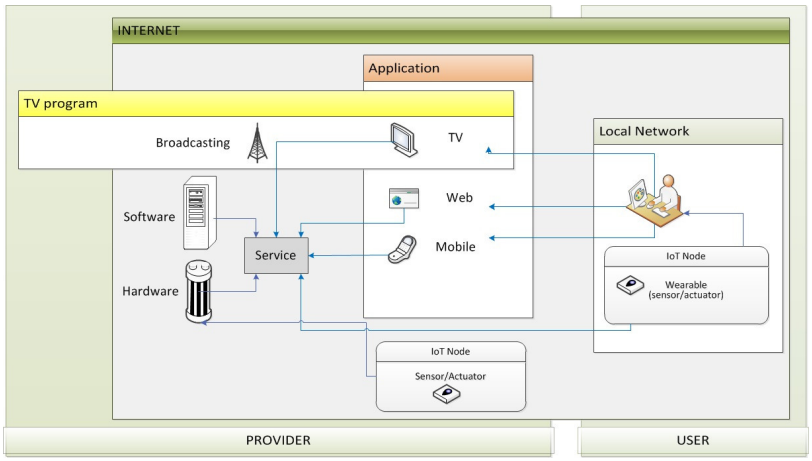

Figure 2: Analysis scenario used for the construction of the methodology

4 An architecture for cross-device IoT applications

The explanation of the proposed architecture is presented in three subsections. The first one explains the division in layers and how each system grouping the functioning of a solution for IoTV is constituted. The second one explains the subsystems in which the general architecture systems are divided. Finally, the third subsection explains the metamodel that represents the domain of IoTV.

4.1 The architecture

In Phase 1, an architecture based on layers was defined, which allows us to relate and group the characteristics of the platforms. This allows grouping the particularities of each platform. Digital TV has typical characteristics of a Broadcastig service. Smart Phone and Wearables devices cover software and hardware aspects. For this reason three layers are proposed: (1) **Service Layer**, which covers everything related to the broadcasting transmission system, (2) **Interoperability Layer**, which covers the system's software development environment, and (3) **Interconnection Layer**, which covers the hardware used to access the network.

Figure 3 shows the layers that make up the architecture and that group all the elements of the technologies used for the analysis. This distribution groups the upper layers of the digital TV (Transmission and reception, transport, source coding) in the Service Layer. The Interoperability Layer corresponds to the top layers of the Client-Server architecture (Application Layer and Transport Layer), the lower layers of the digital TV (Middleware and Interactivity Channel), and the upper layers of the Smart Phone and SmartTV (Kernel, Libraries, Application Framework, Applications, Connectivity platform). The Interconnection Layer corresponds to the lower layers of the Client-Server architecture (Internet Layer and Access Layer), the lower layers of the Smart Phone and SmartTv (Mobile Central Network and Access Network), and for the case of the devices Wearables the Node IoT layer.

The **Service Layer** refers to broadcasting systems such as TV, covering the whole process of coding, multiplexing, and construction of data carousel and incorporation to the radio frequency and transmission stage. Protocols such as OFDM (Orthogonal Frequency Division Multiplex) are applied to encode the signal to the transmission medium. Here it is important to consider that it is also susceptible to attacks [Bachy, Y. et al. 2015] by means of an illegal transmission on the carrier frequency assigned to the television station. The data used for the construction of the carousel are generated in the Interoperability layer. This layer defines the characteristics of the "Deployment System", which describes the characteristics of the service provider, which delivers the application to the television station so that it is inserted into the data carousel and can be transmitted.

The **Interoperability layer** considers the application protocols (HTTP, TCP and UDP). This allows the use of service-based architectures for application development. In addition, it allows implementing encryption schemes such as SSL, TLS and HTTPS. In this stage, authentication mechanisms must be established. To provide interoperability, REST web services must be implemented. This layer defines the characteristics of the "Management System", which allow the interactivity of the platforms through web services. It also allowed access to services provided by different providers. To provide security, a basic authentication is proposed, which

allows the validation between pairs of connected IoT devices that exchange information [Alaba, F. A. et al. 2017].

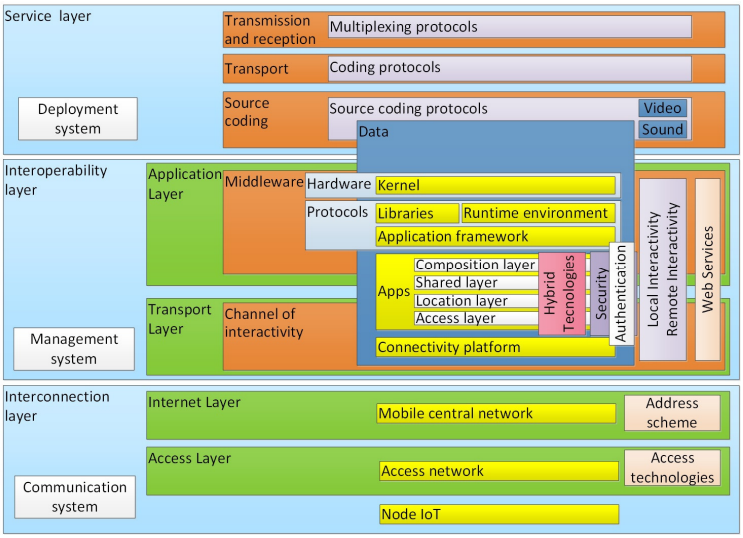

Figure 3: Cross-device architecture divided into layers

The **Interconnection Layer** allows the use of all access and network technologies. This layer has its own protocols for IoT deployment (WiFi, ZigBee, Lora, Bluetooth Low Energy, LowPAN, COAP and MQTT). Although wireless access systems are the most common, they are also the most insecure and susceptible to attack. For this reason it is necessary to use protocols such as 802.11i. In this layer, the characteristics of the "Communication System" are defined, in which the IoT Node interacts with the outside world, a sensor or an actuator. The Nodes can have features to connect to the network directly or when they have limited resources, they can require a device with access to the network that acts as a "Gateway".

4.2 Definition of the architecture subsystems

Each of the systems has been separated into different subsystems, to identify the stages that must be implemented for the deployment of an application. Figure 4 shows the interaction of each of the subsystems.

The **Coding and Transport subsystem** covers the process by which a television station and the service provider encode the audio, video and data signal to a signal that can be processed. Next, the data carousel is built. In this stage, all the control frames are incorporated into the digital signal, by means of a Playout server. Finally, the digital signal is modulated at the frequency of the channel through which the multiplexed signal will be transmitted.

The **Reception subsystem** depends on the middleware that has the hardware of the television receiver, according to this, you can only watch the video and audio signal, or you can run the application. In addition, if you have a connection to the

Internet network you can run the application with remote interactivity, by consuming the services of the application.

The **Interactivity subsystem** describes the entire process that will allow the consumption of a web service on IoTV. The information retrieved here is sent through the interactivity channel so that only the user who requested it gains access.

The **Network Access subsystem** describes the network architecture and the IoT Node. The Node IoT can be found on the user's or provider's side. The IoT Node can be connected to personal area networks (PAN), metropolitan area networks (MAN) or wide area networks (WAN).

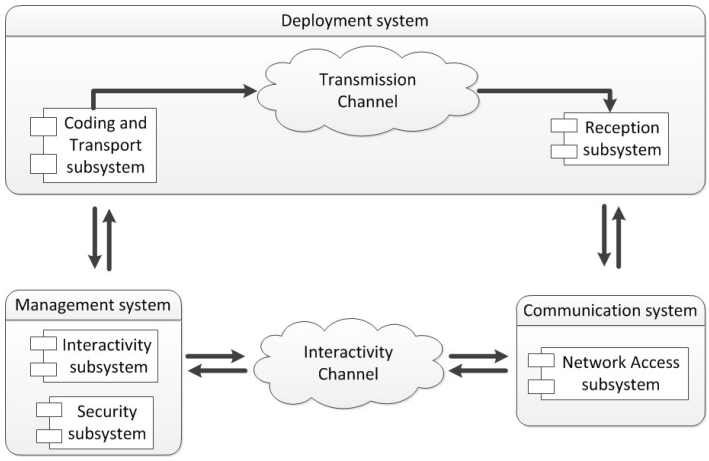

Figure 4: Interaction of the subsystems of the architecture

The **Security Subsystem** allows validation of the credentials of the users, to discriminate and secure the information of the system. Security controls are recommended for IoT implementations [Russell, B. et al. 2015]. The mechanisms in order of priority to guarantee security are: authentication, data encryption, HTPS protocol and DPWS protocol [Avila, K. et al. 2017]. For this reason, a basic authentication process is recommended, which consists of three phases: (1) Key generation phase, (2) Identity establishment phase of the device, and (3) Access implementation phase of the authenticated devices. Although authentication does not completely prevent attacks, it reduces the risk [Ammar, M. et al. 2018].

Identity verification is very important in the creation of trust in people who access computer systems because they authenticate that the person is who they say they really are [Clark, P. et al. 2010]. Authentication methods are usually divided into three major categories, depending on what they use for identity verification [Ribeiro, S. and Suiama, D. 2011]: (a) Authentication based on biometric data (digital, retina, iris, voice, face, DNA, etc.); (b) Token-based authentication (token, identification card, passport, etc.); and (c) Knowledge-based authentication (passwords, security phrases, PIN, etc.). The most basic authentication model is a user and password that should only be known by the user. Obviously, this approach is the most vulnerable to all types of attacks, but also the cheapest.

In Figure 5, the flow associated with an authentication process is shown. The user can access the system from his mobile App or from the TV. If it is the first time, the user can only check in from App mobile because it is an easier way to use for data entry. When the user has registered, they can enter the system from any of the platforms. With the difference that in the case of TV, the entry is made through the token generated in the registration process, while the mobile works with the combination of user and password.

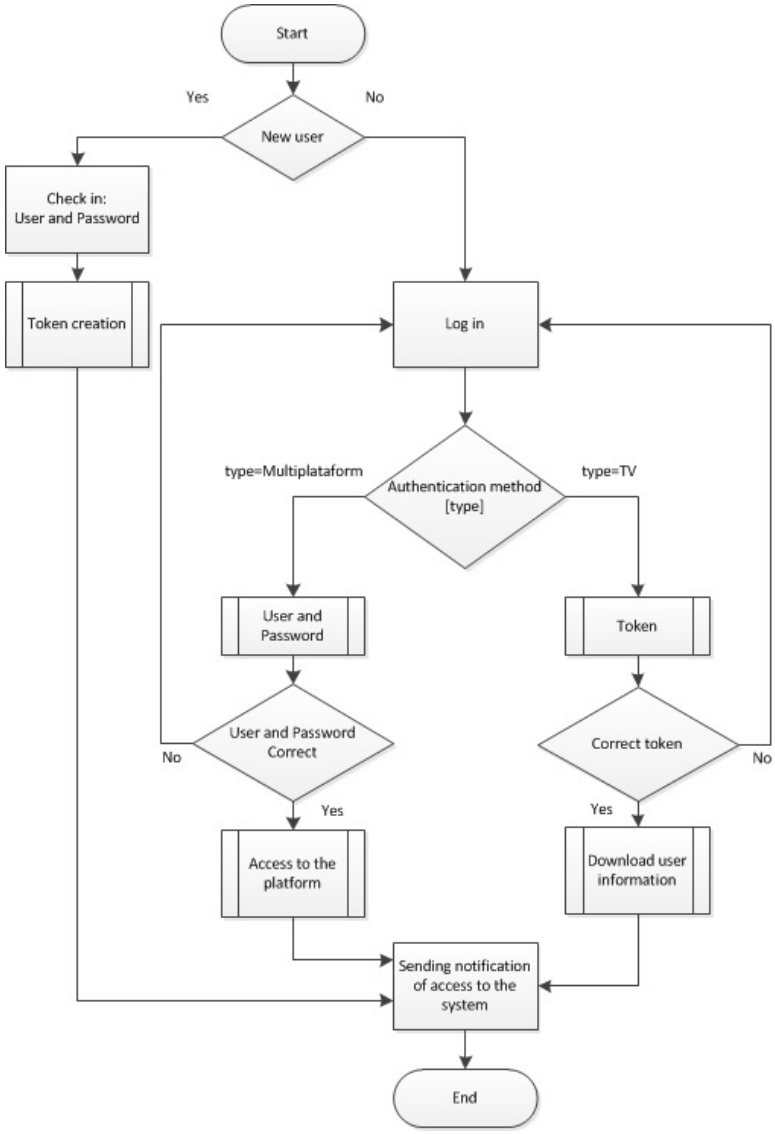

Figure 5: Basic authentication process proposed for architecture

4.3 A metamodel for cross-device IoT scenarios

For the design of the metamodel, we consider the recommendations of the W3C, which mentions the service-oriented model (SOM). SOM explains the relationships between an agent and the services it provides and requests [W3C 2004]. In addition [Alulema, D. et al. 2017] present an approximation of a multiplatform architecture. In Figure 6, the metamodel used to represent the IoTV domain is presented. The metamodel considers three actors: user, provider and IoT node.

The user will consume the services offered by the provider. Interactions with the system depend on its connectivity capacity, and three scenarios can be considered:

a) *User without return channel:* Access the application transmitted via Broadcasting, and interacts in a limited way, viewing program information.
b) *User with return channel:* This has the possibility to authenticate to the system. Once validated, access is granted to the personal and system information display. The return channel can be also used to request dynamic information from the application, which is updated by the service provider.
c) *Multiplatform User:* Enter from a platform other than TV, mobile and Web. User can edit, enter, modify or delete information.

The Provider is the one offering the service. The interaction it has with the system depends on its ability to develop, disseminate and offer a service, and three scenarios can be considered, which can act separately or integrated into a single actor:

a) *Broadcasting Provider:* Provides the Broadcasting signal through which the TV signal and the application are sent.
b) *Application Provider:* Develops final services for users, who can consume them in any of the proposed platforms. In addition, it determines the security schemes that will be implemented.
c) *Platform Provider:* Develops intermediate services, which can be consumed by other providers to develop final applications.

The NodeIoT is a non-human actor that can be found on the user's side when it consumes a final service and contributes to its information, to have a more enriched experience. When the node is on the provider's side it serves to offer a service that can be consumed by the user. It can fulfill the role of a sensor or actuator or both at the same time. The NodeIoT can connect to the system network in two ways:

a) *PAN network:* is the scenario in which the NodeIoT is connected through an intermediary "Gateway", which is responsible for connecting to the service. Establish a Personal Area Network.
b) *LAN / WAN network:* is the scenario in which the NodeIoT connects directly to the service, but with the credentials of the owner user.

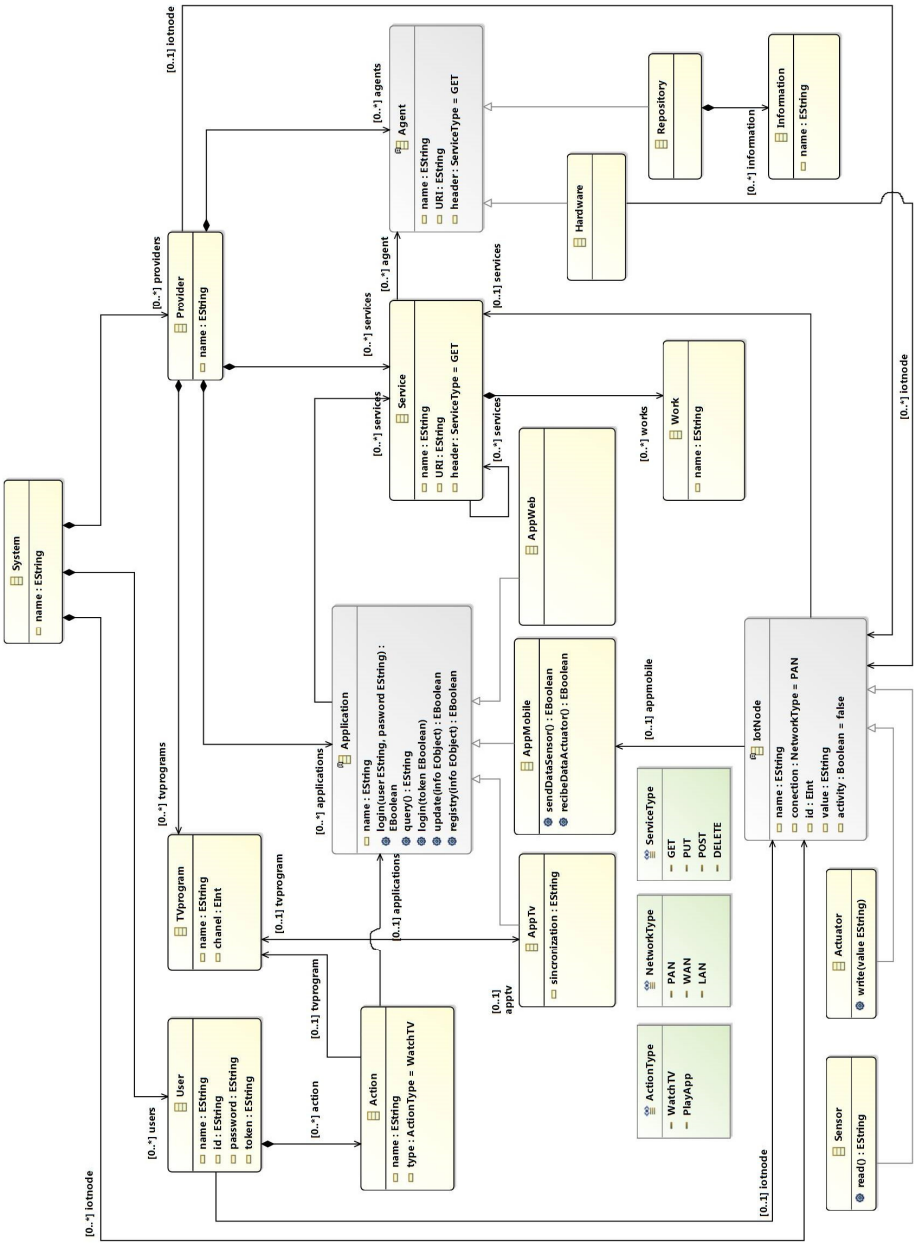

Figure 6: A metamodel connecting some IoT devices

The user will consume the services offered by the provider. Interactions with the system depend on its connectivity capacity, and three scenarios can be considered:

For the representation, the Deployment System has not been represented, because it corresponds completely to the transmission infrastructure of the TV. For the Administration System, composed of the Interactivity and Security subsystems, the User and Provider classes have been considered. In the case of User, it is related to the NodeIoT Class, since users and providers can have their own nodes, and with the Action class, which defines the two types of actions that the user can have, simply watch the TV program or interact with the applications implemented in each platform.

The Provider has a direct relationship with the TVProgram, Application, Service, Agent and NodeIoT classes. These classes represent the products that the provider could offer. The TVProgram class represents the program in which the TV application is embedded. The service defines the web services consumed by the applications, the agent represents the hardware service and the storage (for data volumes generated), and the NodeIoT represents the sensors or actuators, which according to the service that can be designed interact with the environment.

5 A case study scenario

To validate the metamodel it has been considered a T-Health scenario, in which the organisms of a state promote the population to perform physical activity and to reach the widest possible audience, they use television and a mobile application. This context seeks the development of IT apps, in which people and things are connected promoting the use of tools that measure physical activity on a large scale, allowing us to have useful data for new services.

The main T-Health areas, presented in [Rivas, C. et al. 2015], includes: alarms, remote consultation, monitoring of vital signs, monitoring of activities of daily life, detection of falls, early detection of diseases, exercises at home, support for human caregivers, accompaniment and entertainment, robots at home, TV as a gateway to social and health services, and analysis of measurements. For the Testbed, an IoTV scenario was designed, which allows remote consultations, monitoring of vital signs and analysis of measurements. The Cross-Device scenario allows access from the Smart Phone or TV, and a basic authentication scheme is implemented for users. Figure 7 illustrates the test architecture implemented.

The architecture serves as a guide for the convergence of the proposed requirements since the service layer allowing defining the coding, multiplexed and REMUX stages of the audio, video and data signals. For this test scenario, the ISDB-Tb standard was used, which defines the MPEG2 and MPEG4 schemes. In this, the data carousel is built with the application that is transmitted through Broadcasting, so that the user receives the TV signal and, if desired, executes the application. For the implementation, the VillageFlow server was used, which reproduces all the process that must be done in the transmission station, until generating the Radio Frequency signal that will be transmitted [FS24 2017].

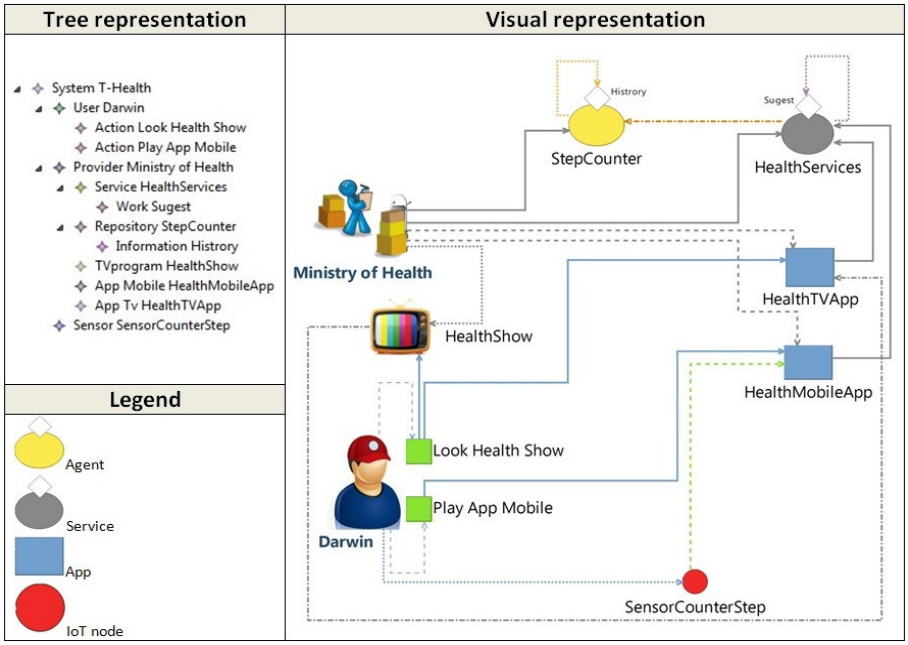

Figure 7: Architecture of cross-device tests with mobile and Digital-TV

The interoperability layer allows us to abstract the mechanism of interoperability of the platform since digital television uses the Ginga and the Java mobile platform. This interoperability is achieved through the use of Web Services REST, for communication with the repositories where the information of the system is stored. In addition, the security scheme that depends on the platform with which the system is accessed is defined. In the case of the TV, it is authenticated by means of a token generated at the time of registration, and in the case of the mobile application, by means of the couple user/password.

The interconnection layer defines the Internet access mechanism. In this case, the User has two connections: the first one is for the TV that can use WiFi or Ethernet, and the second is for the Smart Phone that can use WiFi or LTE. The IoT Node, which is on the user side, is a node that depends on the Smart Phone to send the data to the system. Finally, the Provider must have a broadband connection to allow its servers to handle the data traffic that can be generated.

Before implementing the proposed system, several tests were accomplished with different types of TV platforms. As shown in Table 1, the system requires that the devices have connectivity and that the TV receivers recognize the Ginga Middleware.

According to the proposed scenario, an instance of the metamodel has been established, whose representation is illustrated in Figure 8. In this case, a user "Darwin" has been created, who can see the program that is broadcasted through Broadcasting and execute the integrated application in the signal, or run the mobile application, with which you can register, consult historical data or feed the system

with sensor data to count the steps, associated with Smart Phone. There is also a "Ministry of Health" provider, which develops the applications for TV and Smart Phone, and provides the Web Services that is linked to the agent of the authentication repository, used to offer the basic authentication scheme and storage of information.

Platform	V	C	R
EiTV smartBox(ET-SBX03)	✓	✓	✓
MundyHome (DC-012B)	✓	✖	✓
TV simulator	✓	✓	✖

Table 1: Execution tests (V: Video; C: Connectivity; R: Carousel)

In Figure 9, the flow of the case study scenario is presented, with the test system for the authentication. In the first sequence, the new user has to perform the Check In, from the Smart Phone. At this moment the user is created and assigned a Token, which must use authentication from the TV. Then the user has the possibility to enter the system from both platforms. You can also run the TV application without the need to authenticate, but only when the Broadcasting signal is transmitted. When the user has already authenticated, they can make use of the system resources. To upload information to the "NodeIoT" sensor system, the user must be authenticated from the Smart Phone to which the sensor is connected.

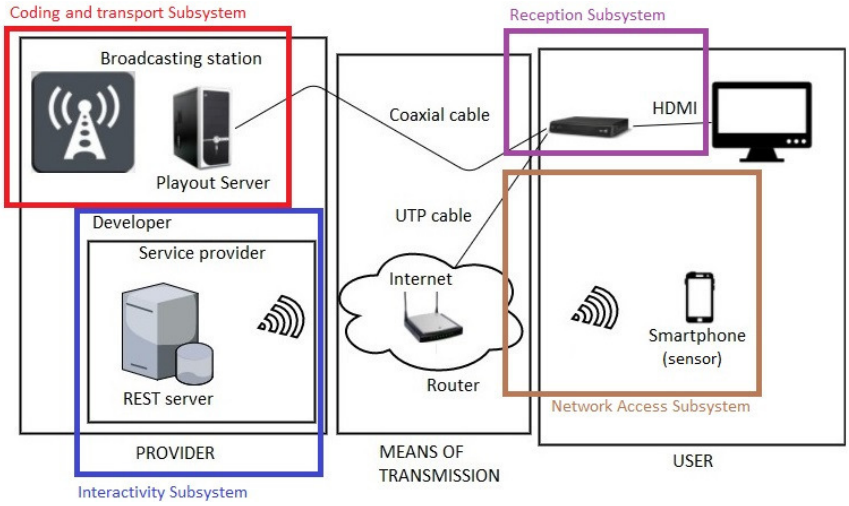

Figure 8: Instance example of the proposed metamodel

6 Conclusions and future work

This paper proposes a cross-device architecture that combines (1) the use of modelling techniques to describe and support the development of IoT applications and (2) the inclusion of authentication mechanisms to provide users with a safe access to

their personal information and ensure their privacy. Furthermore, we take into account different kind of possible IoT nodes including TV, thus enabling the integration of both technologies and raising the concept of IoTV. With this aim, our approach defines a layered architecture, which is divided in a set of interaction subsystems and a metamodel is proposed to describe cross-device IoT scenarios.

.

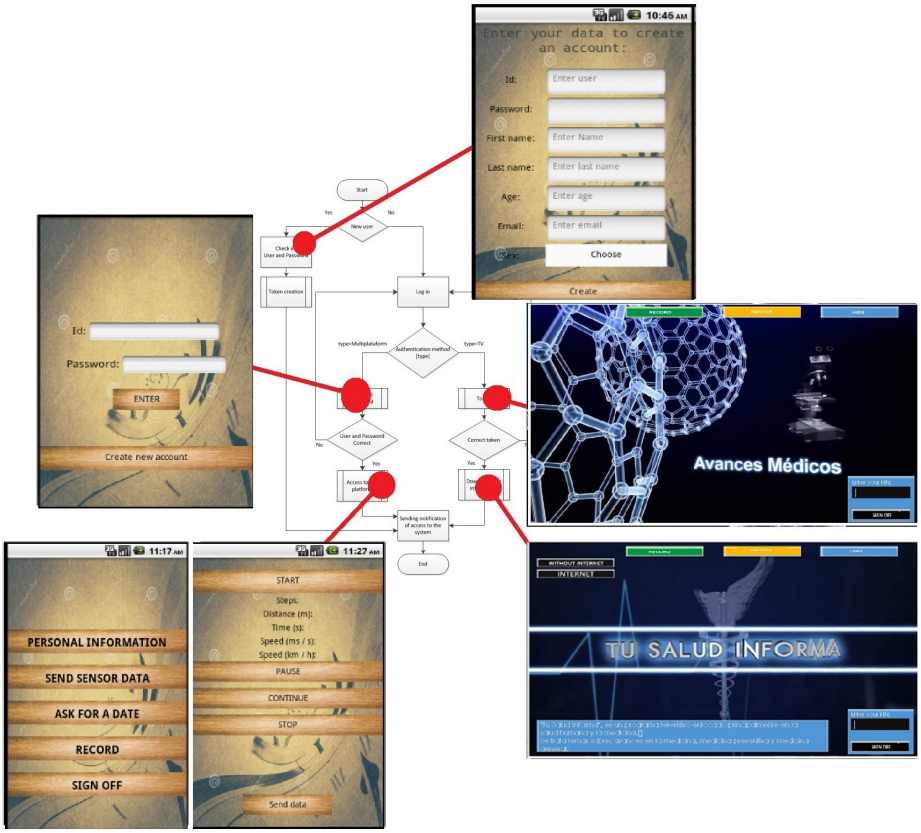

Figure 9: Execution of the test according to the flow for authentication

To validate the proposal, we have tested the use of the Smart Phone, digital TV and a sensor that counts the steps taken by the user. In this way, it has designed a T-Health system that can be accessed from the Smart Phone or from the TV when the broadcast signal is received. The system presents a specific interface for the mobile phone with the physical activity history. The TV presents the data in real time and the suggestions sent online about health. The field of health has been considered, because it is of great importance for people, and for all the information it can generate for government institutions to define plans for the benefit of society.

As future work, we consider developing a set of tools that allow the generation of code automatically or semi-automatically. This would facilitate the development process, by leveraging implementation task without the need of developers to be

experts in all the involved areas. Another line of improvements is related to the technical standards of digital TV, since they have not been updated to incorporate new features, for example, connecting with IoT devices. This update focused on the interconnection of platforms is necessary since the demand on more complex applications is growing and if these updates are not made, the actual TV standards could be disused. The third line is the analysis of the performance of the application since it is necessary to evaluate the behaviour of applications in real environments

Acknowledgements

This work has been funded by the EU ERDF and the Spanish Ministry MINECO under the AEI Projects TIN2013-41576-R and TIN2017-83964-R.

References

[ABNT 2004] ABNT (2004): 'NBR 15605', Associacao Brasileira de Normas Técnicas, p. 209.

[Alaba, F. A. et al. 2017] Alaba, F., Othmana, M., Targio Hashema, I.A. and Alotaibib, F. (2017): 'Internet of Things security: A survey', Journal of Network and Computer Applications, 88(December 2016), pp. 10-28.

[Alulema, D. et al. 2017] Alulema, D., Iribarne, L. and Criado, J. (2017): 'A DSL for the Development of Heterogeneous Applications', 2017 5th International Conference on Future Internet of Things and Cloud Workshops (Fi-CloudW), pp. 251-257, IEEE.

[Ammar, M. et al. 2018] Ammar, M., Russello, G. and Crispo, B. (2018): 'Internet of Things: A survey on the security of IoT frameworks', Journal of Information Security and Applications. Elsevier Ltd, 38, pp. 8-27.

[Avila, K. et al. 2017] Avila, K., Sanmartin, P., Jabba, D. and Jimeno, M. (2017): 'Applications Based on Service-Oriented Architecture (SOA) in the Field of Home Healthcare', Sensors, 17(8), p. 1703.

[Bachy, Y. et al. 2015] Bachy, Y., Basse, F., Nicomette, V., Alata, E., Kaâniche, M., Courrège, J.C. and Lukjanenko, P. (2015): 'Smart-TV Security Analysis: Practical Experiments', Proceedings of the International Conference on Dependable Systems and Networks, 2015-Sept., pp. 497-504.

[Bailey, J. et al. 2007] Bailey, J., Budgen, D., Turner, M., Kitchenham, B., Brereton, P. and Linkman, S. (2007): 'How Software Designs Decay: A Pilot Study of Pattern Evolution', Proceedings - 1st International Symposium on Empirical Software Engineering and Measurement, ESEM 2007, pp. 449-451.

[Benouda, H. et al. 2016] Benouda, H., Azizi, M., Esbai, R. and Moussaoui, M. (2016): 'MDA Approach to Automate Code Generation for Mobile Applications', Mobile and Wireless Technologies. pp. 241-250.

[Bosin, A. 2012] Bosin, A. (2012): 'An SOA-BasedModel for the Integrated Provisioning of Cloud and Grid Resources', Hindawi Publishing Corporation, Advances in Software Engineering, pp. 1-19.

[Clark, P. et al. 2010] Clark, P., Cook, G., Fisher, E., Fulp, J., Linhof, V. and Irvine, C. (2010): 'New Pathways in Identity Management', IEEE Security and Privacy 8 (6):64-67.

[Ding, Z. et al. 2016] Ding, Z., Xu, T., Ye, T. and Zhou, Y. (2016). 'Online Prediction and Improvement of Reliability for Service Oriented Systems' IEEE Transactions on Reliability 65(3):1133-1148.

[FS24 2017] FS24 (2017): Village Flow Encoder-Mux-Remux.
Available at: http://www.fs24.com.ar/village-ow-encoder-mux-remux-isdb-t/.

[Garcia, C. G. et al. 2014] García, C., Espada, J., Núñez, E. and García, V. (2014): 'Midgar: Domain-specific language to generate smart objects for an internet of things platform', Proceedings - 2014 8th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, IMIS 2014, (June), pp. 352-357.

[Hossain, M. et al. 2017] Hossain, M., Riazul Islam, S.M., Alic, F., Kwakc, K. and Hasana, R. (2017): 'An Internet of Things-based health prescription assistant and its security system design', Future Generation Computer Systems, Vol. 82, Pages 422-439. Elsevier.

[Hu, P. et al. 2018] Hu, P., Ning, H., Qiu, T., Xu, Y., Luo, X. and Sangaiah, A.K. (2018): 'A unified face identification and resolution scheme using cloud computing in Internet of Things', Future Generation Computer Systems, Vol. 81, Pages 582-592. Elsevier.

[Hsu, C. and Huang, C. 2018] Hsu, C. and Huang, C. (2014): 'OptimalWeighted Combinational Models for Software Reliability Estimation and Analysis', IEEE Transactions on Reliability, 63(3):731-749.

[Jeyanthi, N. and Thandeeswaran, R. 2017] Jeyanthi, N. and Thandeeswaran, R. (2017): 'Security breaches and threat prevention in the internet of things', (March), p. 52.

[Luo, S. et al. 2014] Luo, S., Cheng, L. and Ren, B., (2014): 'Medical Monitoring and Managing Application of the Information Service Cloud System Based on Internet of Things', Journal of Software, 9(7):1802-1809.

[Mainetti, L. et al. 2015] Mainetti, L., Mighali, V. and Patrono, L., (2015): 'A Software Architecture Enabling the Web of Things', IEEE Internet of Things Journal, pp. 445-453.

[Mishra, A. and Mishra, D. 2006] Mishra, A. and Mishra, D. (2006): 'Software Quality Assurance Models in Small and Medium Organisations: A Comparison', International Journal of Information Technology and Management, 5(1):1-4.

[Mishra, D. and Mishra, A. 2004] Mishra, D. and Mishra, A. (2004): 'Simplified Software Inspection Process in Compliance with International Standards', Energetik, 31(7):21-23.

[Mishra, D. and Mishra, A. 2011] Mishra, D., and Mishra, A. (2011): 'A Review of Non-Technical Issues in Global Software Development', International Journal of Computer Applications in Technology, 40(3):763-771.

[ONTSI 2016] ONTSI, (2016): 'Development of Methodology and studies on the interoperability levels of the main service management platforms of smart cities'. Edited by energy and tourism Ministry of Industry, Spain.

[Pan, J. and McElhannon, J. 2017] Pan, J. and McElhannon, J. (2017): 'Future Edge Cloud and Edge Computing for Internet of Things Applications', IEEE Internet of Things Journal, 4662(c), pp. 1-11.

[Ribeiro, A. and Rodrigues, A. 2014] Ribeiro, A. and Rodrigues, A. (2014): 'Evaluation of XIS-Mobile, a Domain Specific Language for Mobile Application Development', Journal of Software Engineering and Applications, 7(11):906-919.

[Ribeiro, S. and Suiama, D. 2011] Ribeiro, S. and Suiama, D. (2011): 'A Method for Identifying and Analyzing Authentication Mechanisms and a Case Study', DPPR 2011: Advances in Digital Image Processing and Information Technology, pp. 449-459.

[Rivas, C. et al. 2015] Rivas, C., Anido, L. and Fernández, M., (2015): 'An Open Architecture to Support Social and Health Services in a Smart TV Environment', IEEE Journal of Biomedical and Health Informatics, 21(2):549-560.

[Russell, B. et al. 2015] Russell, B., Garlati, C. and Lingenfelter, D., (2015): 'Security Guidance for Early Adopters of the Internet of Things (IoT)', Mobile Working Group Peer Reviewed Document, (April).

[Troya, J. et al. 2013] Troya, J., Vallecillo, A., Durán, F., and Zschaler S. (2013): 'Model-driven performance analysis of rule-based domain specific visual models', Information and Software Technology, 5(1):88-110.

[Vijayakumar, P. et al. 2017] Vijayakumar, P. et al., (2017): 'Computationally efficient privacy preserving authentication and key distribution techniques for vehicular ad-hoc networks', Cluster Computing, 20(3):2439-2450.

[W3C 2004] W3C (2004): Web Services Architecture Usage Scenarios. https://www.w3.org/TR/ws-arch-scenarios/

[Wolf, M. 2018] Wolf, M., (2018): 'Safety and Security in Cyber - Physical Systems and Internet-of-Things Systems', Proceedings of the IEEE, 106(1):9-20.