User Profile for the Internet of Things based on Ontologies

DOI: http://dx.doi.org/10.17981/ingecuc.20.1.2024.07
Artículo de Investigación Científica. Fecha de Recepción: 17/08/2023, Fecha de Aceptación: 03/11/2023

Yazmin Andrea Pabón Guerrero https://orcid.org/0000-0002-0033-0347
Universidad Carlos III de Madrid, España.
ypabon@inf.uc3m.es

Líder Julian Rojas Bolaños https://orcid.org/0000-0001-8009-6732
Universidad Carlos III de Madrid, España.
lirojasb@inf.uc3m.es

Miguel Angel Niño Zambrano https://orcid.org/0000-0002-5916-2552
Universidad del Cauca, Popayan, Colombia.
manzamb@unicauca.edu.co

To cite this paper
Y. Pabon-Guerrero, L. Rojas-Bolaños & M. Niño-Zambrano "User Profile for the Internet of Things based on Ontologies" DOI: *http://dx.doi.org/10.17981/ingecuc.20.1.2024.07*

Abstract

Determining the context in which the user is located is key to personalizing services in the Internet of Things. Therefore, a generic, semantic, interoperable mechanism that manages the user profile is required. A user profile management mechanism for IoT is proposed that uses ontologies to model user preferences and their context according to their location and the activities they are executing. This enables users to establish preferences on smart objects that create personalized services with minimal human intervention. The methodology used in creating this user profile ontology facilitates the rapid building of a formal, evolutionary model. To validate the solution, a case study adopted a semantic interaction scenario of smart objects, which incorporated real users and measured their ability to personalize the services provided. The results suggest that ontology offers a good solution for this kind of scenario that adapts its services to user characteristics and context. Testing verified that services were automatically adjusted within an acceptable time (52 seconds) in comparison with manual adjustments that might require several minutes.

Key words

User Profile, Context, Ontology, Internet of Things (IoT), Semantic Web

1 INTRODUCTION

Users currently encounter difficulties in finding services or information in the IoT because the information does not meet their particular needs at either a specific time or place [1]. Determining the context in which the user is located is an important part of personalizing services since there are several devices involved, both physical and virtual. This obliges a system to understand the context of the user to adapt its behavior [2] to provide timely services or relevant information [3], thus allowing autonomous behavior of the system with minimal human intervention [4] [5] [6].
Studies found in the literature have focused on gathering context information in ontologies and other semantic structures, focusing more on the information of the service domain than on the preferences of the users themselves. Studies such as those carried out by Wei et al. [7] and Ara et al. [8] seek to model the context through ontologies

and artificial intelligence techniques, to provide personalized services to users. But they focus very little on the particular preferences of users and their relationship with objects in the environment, which allows a closer approximation to their interests and therefore, an identification of the services offered by the IoT that best suits them [6].

Others Guinard et al. [9], Van et al. [10], and Ashibani et al. [11] take user profile into account but limit it to establishing roles and permissions to access services and devices and not configuring these services according to the preferences of the users. Other studies meanwhile implement user profile but take it from social networks where information may be incomplete or erroneous, without applying a filter to validate this information [12].

The research question is therefore posed: How can the interaction between users and IoT objects be improved in such a way as to generate a better experience with the services they offer?

The present article puts forward the proposal of a services context model for the IoT, focused on user profile, which allows managing the information of each user and their interactions with the objects in the IoT to generate services better adjusted to the characteristics or preferences of the users.

The following section, Section II, presents related work. In Section III, the methodology used in the creation of the User Profile ontology for the IoT is presented, and in Section IV, the proposed ontology. Section V presents the architecture of the system that contains the proposed ontology. Section VI presents the evaluation made on the proposed model. Finally, conclusions and future work are presented in Section VII.

2 RELATED WORK

In the literature, several studies were found that perform personalization of services in the Internet of Things, with three common components being identified: context, key concepts in the user profile model, and semantic interoperability. Each of these components is presented below.

2.1 Managing the Context for User Profile in IoT

Perera, et al. [13] proposed a life cycle of context in the IoT, which is composed of the phases of acquisition, modeling, reasoning, and distribution. Given the importance of the context life cycle for the management of user profile in this research, it was decided to make a comparative analysis, in which these phases are identified in the related work (See Table 1).

Table 1: Comparative analysis of context life cycle for IoT user profile

Source: own authorship

Author	Context Life Cycle					
	Acquisition			Modeling	Reasoning *	Distribution
	Sensed	Derived	Manual			
Wei and Jin [7]	X	-	X	Ontology	DBN	REST
Ara, et al. [8]	X	-	-		SPARQL and SWRL	-
Kim, et al. [14]	-	X	X		IE	-
Biamino [15]	X	-	X		DL	-
Lin, et al. [3]	-	-	X		-	-
Console, et al. [16]	-	-	X		SWRL	-
Skillen, et al. [17]	-	-	X		SWRL	-
Golemati, et al. [18]	-	-	X		Rule-based	-
Otebolaku, et al. [19]	X		X		SWRL	Repository
Proposal presented in this article	X	X	X		SPARQL and SWRL	REST
Corno, et al. [20]	X	X	X	Device-Centric & People-Centric	Rule-based	-
Prasad, et al. [21]	X	X	X	Data	Agents & Query	SOAP

* DBN: Dynamic Bayesian Networks, IE: Inference Engine, DL: Description Logics

Regarding the acquisition, most of the studies acquire the primary context of the raw data received by the sensor. The characteristics and preferences of the user are meanwhile obtained manually, by asking the user directly. In the case of the present project, the same strategy is used to combine the data of the sensors with the data provided by the user, to store them in the user profile ontology. As regards modeling, the most popular technique is through ontologies, because these have high interoperability between systems and additionally, they make it possible to relate the user profile concepts semantically. We, therefore, chose to use this modeling technique in the present project. The most recent research takes advantage of acquiring context from all possible sources. Additionally, web services are used for distribution, and the context is stored in dynamic databases since the context changes over time.

Regarding reasoning, when modeling user profile using ontologies most studies lean toward ontological and rule-based reasoning. The latest research [21] uses intelligent agents that monitor user activities and generate a data query about their context.

Finally, with regards to distribution, although most literature does not provide any information related to this phase, it was decided to opt for RESTful web services since this architecture is one of the most widely used to distribute services in IoT. Additionally, a formal process is proposed for the distribution of context, along with its validation in a particular case study.

2.2 Managing the IoT User Profile through Ontologies

A substantial part of the research found modeled user profiles through ontologies so that after studying them, concepts in common were identified and included in this work.

As can be seen in Table 2, few studies bring together all of the concepts identified in the exploratory phase of this research, indicating an opportunity to generate a more complete proposal based on each of these elements, to form a user profile ontology that is reusable and therefore applicable to several different contexts in the IoT. Meanwhile, in the majority of studies, except that proposed by Lin et al. [3], the *preferences* concept [includes] only musical preferences, from movies, from text display, whereas in this work the preferences of users regarding the objects at their disposal are also taken into account.

The most recent works [19] [21] involve user profiles with more variables related to devices and security aspects. In this proposal, relationships with the states and data of IoT devices have been modeled through the OOS ontology. Security aspects have been left out for research scope and future work.

Table 2: Concepts of user-profiles found in the literature.

Source: own authorship

Author	Concepts of User Profile									
	Personal Information	Preferences	Activities	Location	Object Service	Conditions of Health	Environment	Time	Physical variable	Security aspects
Biamino [15]	X	X	X	X	X					
Kim, et al. [14]	X		X	X	X					
Golemati, et al. [18]	X	X	X	X		X				
Skillen, et al. [17]	X	X	X			X				
Wei and Jin [7]	X	X		X	X					
Console, et al. [16]			X	X	X					
Ara, et al. [8]					X					
Lin, et al. [3]	X	X	X	X		X				
Ning, et al. [22]	X	X	X	X						
Otebolaku, et al. [19]	X		X	X			X	X	X	
Prasad, et al. [21]	X			X				X		X
Proposal presented in this article	X	X	X	X	X	X	X	X	X	

Considering the above, the present research proposes a user profile ontology that brings together and complements the key concepts identified in other research, offering a broader representation of the static and dynamic situation of the user profile, as well as focusing on the interactions with smart objects, which help define broader preferences and customize services better.

2.3 Semantic Interoperability in IoT Smart Objects

In the IoT domain, the Semantic Sensor Network (SSN) and Sensor, Observation, Sampler, and Actuator (SOSA) ontologies define concepts and realizations that support the semantic interoperability process in IoT in a modular way, taking as a basis some specifications proposed by the Open Geospatial Consortium (OGC) with the Semantic Web Enable (SWE) initiative and the SensorThings API[1].

Niño [23] meanwhile proposes a *"model of semantic interaction between objects of the Web of Things to provide users with semantic abstractions of objects at different levels, so that they share common domains of knowledge and exchange their information, serving as a basis to create services for users"*.

In this study, the Semantic Object Ontology (SOO[2]) was developed, being aligned with SSN/SOSA due to its standardization. In SOO. It proposes a mechanism for programming and customizing user services, using the ECA (Event - Condition - Action) concept (See Figure 1). This mechanism was reused to store user preferences regarding the behavior of smart objects in the user's environment.

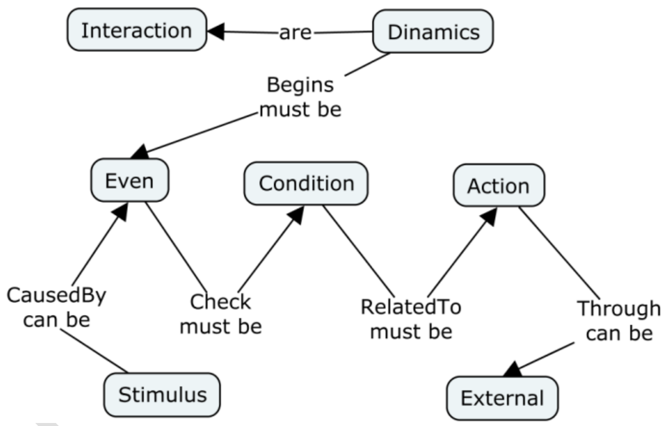

Figure 1: Interoperability Concepts in the SOO
Source: own authorship

Guerrero et al. [24] implemented the architecture proposed by Niño [23] to create a test scenario, deploying three smart objects (temperature regulator, light regulator, and humidity regulator) to create relationships between them and in this way offer new services. This work was oriented to test the viability of the architecture, obtaining a semantic interaction scenario with good response times. As future work, it proposes to exploit the potential of ontologies, since it only uses them to give a semantic representation of the smart object.

The present project reuses the SOO, because it models the concepts related to the smart objects, and proposes a User profile Semantic Model in the IoT, to semi-automatically manage the start-up and configuration of the services offered to the users, using as a basis the development carried out by Guerrero et al. [24], adding the possibility that the services are based on user preferences. To do this, the SOO was extended, as presented in the following sections.

[1] https://www.ogc.org/standard/sensorthings/
[2] ftp://facfiet.unicauca.edu.co/InternetofThings/ontologias/ObjetoSemantico/oos.owl

3 PROPOSED METHODOLOGY

To build the ontology in rapid, evolutionary versions, it was decided to use the methodology created by Niño [23], which is an adaptation of Methontology [25]. The proposed ontology was manually conceptualized, formalized, and supported by CASE tools. (See Figure 2).

The methodology can be understood in two stages: a stage of formal design abstraction and another stage of implementation. Each stage has steps, and some steps have tasks. The tasks are supported by tabular templates such as those proposed by Methontology, but with additional extensions and details to improve ontology documentation. In the implementation stage, it is conceptually designed in a case tool, which later allows exporting the ontology in OWL format. The advantage of this is to focus on the design and relationship of concepts rather than on implementation terms. Next, the steps and tasks are described:

- **Formal abstraction stage**

The ontology is specified, conceptualized, and formalized. To do this:

- Step 1: The questions on the specification of the ontology are answered, to know the requirements.
- Step 2: Decisions are taken on whether ontologies and taxonomies can be reused.
- Step 3: To have a clear idea of the knowledge to be modeled, a textual or graphic description can be made in the company of experts or a valid source of knowledge. A tabular form is filled out for each of the six tasks in this step with precise details and descriptions of the modeled elements.
- Step 4 – task 1: The CmapTools COE[3] program is used, which allows the conceptual model to be generated graphically and following the specification of COE design patterns, to facilitate the implementation process. This specification avoids having to define all the Methontology templates.

- **Implementation stage**
  - Step 4 – task 2: Export the Ontology in OWL format through the export utility of the CmapTools. Then it's Open the ontology in the Protégé[4] program, with which its correspondence with the conceptualization is revised. In case additional adjustments are required these are made at this stage.
  - Step 5: Finally, it verifies the operation of the ontology, considering the questions that are expected to be answered, if possible, using the reasoner integrated in Protégé

[3] https://cmapdownload.ihmc.us/coe/Web_InstallersV5.0/install.htm

[4] https://protege.stanford.edu/

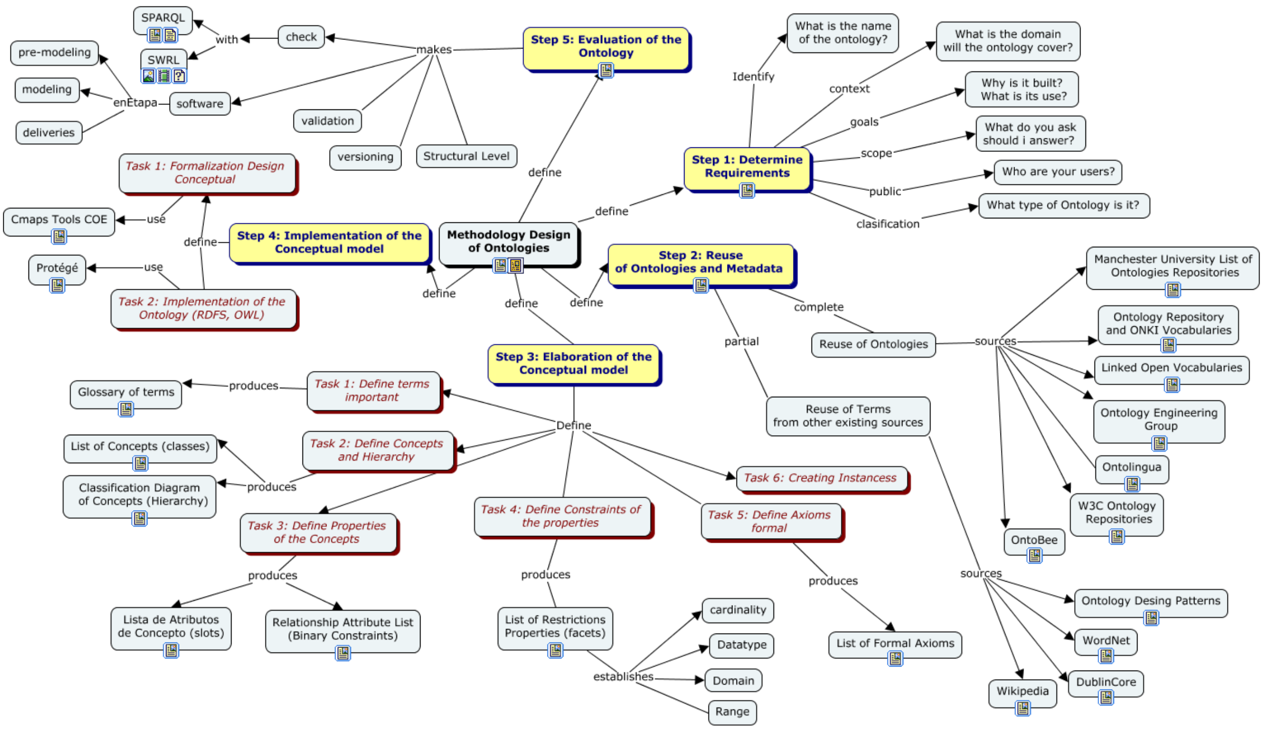

Figure 2: Methodology Used

Source: own authorship

4 ONTOLOGICAL USER PROFILE MODELING FOR THE IOT

The requirements that the user profile model for the IoT should have to be reusable and extensible were identified by studying related research. Personal information of the user, their health condition, activities carried out, profession, and information about the places frequented should all be included. To make a user profile in the IoT meaningful, a record should be kept of the smart objects, the services they provide to the user, and the interactions with the user. Furthermore, a model distribution mechanism should be proposed so that objects, services in the IoT, and other interested parties can use it.

An IoT service context model focused on the user profile was therefore created, which uses ontologies to manage personal information, as well as interactions between users and the services offered by the IoT objects, to generate services better adjusted to the characteristics or preferences of the users.

Given the above and considering the context in which the user profile model was built, the different stages were presented following the methodology chosen for building the model.

As a first instance, context, objectives, scope, target audience, and classification of the ontology were identified and defined using the questions in Table 3.

Table 3: Specification of the user profile ontology for the IoT

Source: own authorship

Specification	Description
What is it called?	User Profile Ontology for the IoT
What is its prefix?	UPO
What domain will the ontology cover?	This ontology makes it possible to model user profile in such a way that certain information that describes it is available so that smart objects or services automatically configure their services according to user preferences.
Why is it built?	This ontology is built to obtain information about the preferences, tastes, interests, and behaviors of the user.
What is its use?	To personalize the services and display the contents and processes required to facilitate the user's interaction with the IoT resources around them.
What questions does it have to answer?	What devices belong to the user? What services can they offer according to the context? How to adapt the service according to the user's preferences? What services can be better adapted to the user according to their preferences? What actions can the IoT objects perform at a certain moment according to the user? How are the spaces normally inhabited by the user configured? What are the user's preferences for a given space? With what objects does the user interact according to the activity he/she is carrying out?
Who are its users?	Anyone who is the owner of the IoT resource, or those users authorized by the owner.
Ontology type	Specific domain ontologies: describe the vocabulary of a specific knowledge domain.
Available	https://zenodo.org/record/8253872

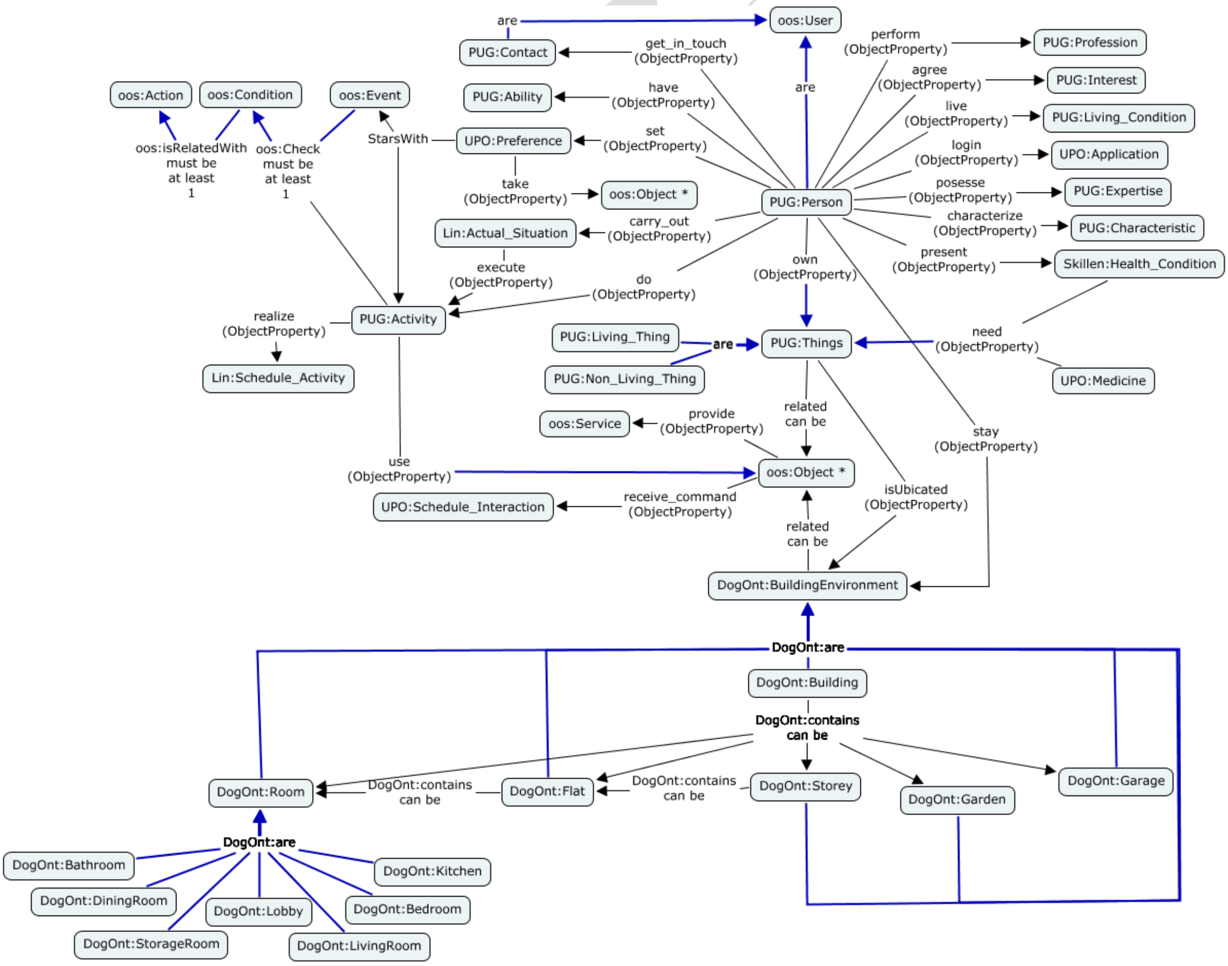

Figure 3: User profile ontology for the IoT

Source: own authorship
* The Object concept was cloned to facilitate the reading of the conceptual model

Taking related work into account, high compatibility was found with the ontology created by Golemati et al. [18], which describes the user extensively. Studying this ontology, concepts, hierarchies of classes, and synonyms necessary for the present work were identified. These concepts are presented in Figure 3 and with PUG prefix. Golemati's ontology represents a large part of the concepts necessary for user modeling, but given the scope with which it was modeled, it did not consider dynamic aspects of the user (current situation, health conditions, etc.) or considerations related to the IoT (user's smart objects, services offered by objects, etc.).
Other ontologies were studied that considered these concepts. The ontology proposed by Lin et al. [3] stood out and the concepts *DynamicSituation, current_time, and current_location* were taken from it and summarized in the *Actual_Situation* concept. *Health_Condition* was taken from Skillen et al. [17]. These are presented in Figure 3: User profile ontology for the IoT, with Lin prefix and Skillen prefixes respectively.
Exploring other ontologies that might enrich the User Profile ontology for the IoT, DogOnt [26] was found. In this ontology, the concept of BuildingEnvironment "*makes it possible to represent domestic environments*", giving the possibility of finding out the place the user is in, to be able to adapt nearby services (DogOnt prefix).
The user profile ontology developed in the present work was furthermore aligned with the Semantic Object Ontology [23] to extend the knowledge that objects have regarding the user (OOS prefix).
As a result of the above, the User Profile ontology for the IoT shown in Figure 3: User profile ontology for the IoT was obtained. The complete ontology is composed of 72 concepts, 35 object properties, 117 data properties, and 623 axioms, including the alignments carried out. Some concepts are intentionally duplicated to make the ontology readable.
The user profile ontology for the IoT presents as a central axis the concept of the person. The other concepts are thus directly related to it.
Below, the ontology is broken down into three different perspectives to facilitate its understanding.

4.1 The perspective of concepts that describe the Person's concept.

The first part refers to the concepts that describe a person, as shown in Figure 4

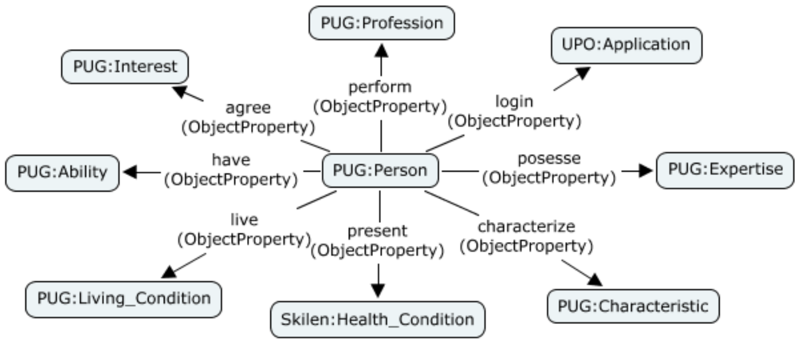

Figure 4: Concepts of the ontology that describe the person.
Source: own authorship

The *Person* concept contains the basic information of the user, such as date of birth and name.
To describe the external morphology of the user, the *Characteristic* concept is used, which stores the name and value of the characteristic, for example, eye color, blue.
The *Health_Condition* concept stores the name of the health condition and the type of condition (illness or allergy).
The *Living_Condition* concept contains information on the user's type of home and address.
The *Profession* concept refers to the professions the user has, storing the place where he exercises these and the type of profession (manual or mental), etc.
The *Interest* concept contains the interests of the user, in both personal (musical genres) and professional (economic news) areas.
The *Ability* concept refers to the skills of the user, for example, the ability to make origami figures. There is also the *Expertise* concept, which stores the level of user experience in different fields, for example, the level of computer experience.
The ontology also contains the *Application* concept to store the access credentials corresponding to the applications that the user uses. This concept is incorporated into this proposal.

Characteristic, *Health_Condition*, *Interest*, *Ability*, and *Expertise* have an attribute called score, which allows the user to assign the degree of importance of each of these instances.

4.2 The perspective of the dynamic situation of the person

Furthermore, there is a set of concepts in the ontology that refer to the activities that the user is performing at a certain moment and the objects with which he or she is interacting (Figure 5).

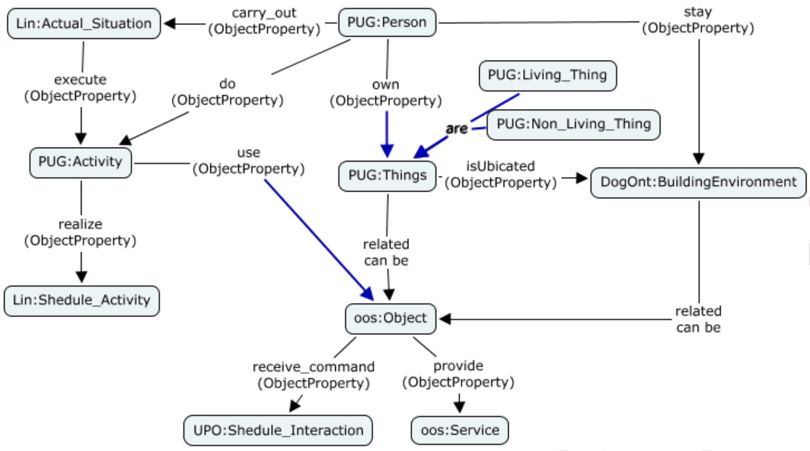

Figure 5: Concepts of the ontology that represent the dynamic aspects of the user.
Source: own authorship

The *Actual_Situation* concept stores the current time, the user's location, and their mood, as well as the activity they are carrying out, to be clear about their current situation.
The *Activity* concept refers to the different activities carried out by the user (running, playing sports, reading). To complement the Activity concept, the schedule of each activity is also stored in the *Schedule_Activity* concept.
The *Object* concept stores the different smart objects belonging to the user. In this concept, the object's ID and its IP address are stored to connect to them when the user requires them. In addition, the services provided by these objects are obtained through the *Service* concept.
The *Schedule_Interaction* concept stores the date and time in which the user activates or deactivates a service of the smart object. This is so that later the object becomes aware of repetitive actions and can execute these automatically. This concept is incorporated into this proposal.
Moreover, the concepts of *Activity* and *Object* are related to discovering which smart objects the user uses when performing a certain activity. It should be noted that smart objects can be related to a thing, a building, or a part of the house.
The *Things* concept refers to the things that belong to the user and these can be either *living things* (pets, houseplants) or *non-living things* (cars, tables). These are related to an *Object*.
Thing is an abstraction of a higher level than Object concept since Object concept refers to the physical things that are not a smart object.
The *BuildingEnvironment* concept stores the distribution of the different spaces the user inhabits. This concept is related to the *Things* and *Objects* concepts to find out where they are specifically located within the user's environment.

4.3 The perspective of user preferences

The User Profile ontology makes it possible to define preferences regarding the behavior of smart objects a user has at their disposal (Figure 6).

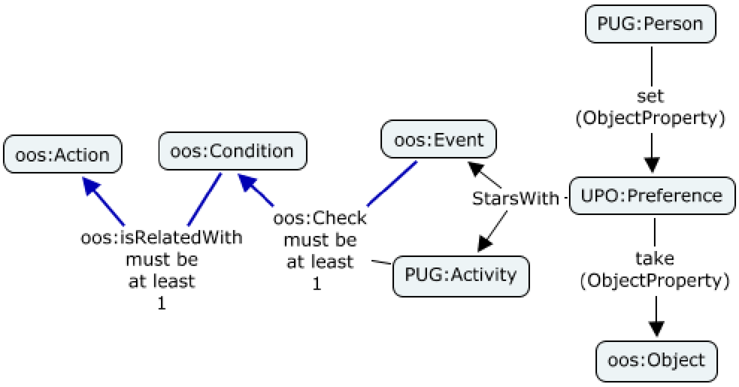

Figure 6: User Preferences
Source: own authorship

The *Preference* concept stores the name of the preference and its status (on, off). A preference can start with an Event or with an Activity. This concept is incorporated into this proposal.
The *Event* concept is related to a resource of the object that is to be monitored. For example, in the Temperature Regulator object, the event can be defined by the temperature sensor. The *Condition* concept meanwhile defines the condition that triggers the action, for example, temperature greater than 24 °C. Finally, the *Action* concept corresponds to a resource of the object that changes its state according to the event and the condition defined. For example, fan on. As such, the user preference is as follows: *When the temperature is higher than 24 ° C, then turn on the fan.*
The behavior of a smart object can be related to a *preference* for a certain *activity*: for example, *when reading (Activity) starts (Condition), then turn on the light (Action).*
In this model, the actuator-type resources are the only ones that have an on/off or true/false, or 1/0 state, since the sensors always return a quantitative measurement of the physical variable to which they are related. As such, to create preferences where the event resource is an actuator, we would have the following: *When the air conditioner turns on, then close the windows.*
The preference concept stores a minimum and maximum range around the value compared to the condition, so that the action is not unstable when the sensor value varies slightly. For example, were the Temperature preference to be set at 24 °C, where if this value is exceeded, the fan turns on, and if it is below, the fan turns off, experimentally what happens is that the temperature sensor on being at the limit of the measurement (23.9 °C or 24.1 °C), the fan will continuously turn on and off, creating a feeling of system instability. If the preference includes more than one possible event, a preference would be created for each event.
If two preferences collide in system-understood instructions, it would be possible to use *score_preference* to define which preference to run, but in terms of the experiment, it was not considered, so it is up to the user to create preferences that are not conflicting with each other.

4.4 Example of instantiation

In the following, a motivational scenario is presented to show the instantiation of the ontology.
María (Name), a 30-year-old (Age) female (Gender), is a full-time (FullTime_Profession) teacher (Profession), in Popayán (City_Profession) who spends her weekend at her country house (House_Type_Living) knitting in her workshop (Ability, Ability_Type Physical).
Most of her fabrics are made of blue wool since blue is the color of her eyes (Characteristic Name_Charact Eyes, Value_Charact Blue). While weaving, she listens to an audiobook at medium volume (Preference When Weaving Starts, then Audio Player Turned On) that teaches her to pronounce words in English (Interest, Interest_Type Work). As she likes knitting, she is always in a good mood (Mood) and the time often goes by so quickly that without realizing it there is no longer any natural light. So, she switches on three lightbulbs (Schedule_Interaction 10/25/2017 Command On, ID_Resource Lightbulb) to continue with her work, since she has a visual problem (Health_Condition Name_Health: Myopia; Type_Health: Illness).
María enjoys watering the houseplants herself. However, while she is working, they are watered by a device that takes care of it. During the night it is very cold, so Maria set a preference to keep the temperature at 25 °C (Preference When Temperature < 25 °C then Turn on the Heater).
Table 4 summarizes the assertions that would be made in the ontology automatically, either through manual or automatic data acquisition.

Table 4: Example of Ontology Assertions
Source: own authorship

Type / Concept	Data property Assertion
Person	Facebook “facebook.com/maria.iot.77”^^xsd:string
	Place_of_birth“Popayán ”^^xsd:string
	Surname “Iot ”^^xsd:string
	Date_of_birth “03/10/1987 ”^^xsd:date
	Phone “3210000111 ”^^xsd:string
	Gender “female ”^^xsd:string
	Name_person “ maria ”^^xsd:string
	Email “perfilusuarioiot@gmail.com ”^^xsd:string
Preference	Name_preference “Temperature ” ^^xsd:string
	Max_value_preference 25
	Min_value_preference 21
	Score_preference 85.0f
Interest	Type_interest “work ”^^xsd:string
	Name_interest “LearnEnglish ”^^xsd:string
	Score_interest 100
Activity	Score_activity 100.0f
	Temporary_activity “partialtime ”^^xsd:string
	Hours_per_week_activity 10.0f
	Name_activity “weave ”^^xsd:string
	Type_activity “ physic”^^xsd:string
Actual_situation	Current_location “ sewing workshop”^^xsd:string
	Mood “ happy”^^xsd:string
	Current_time “ 09/02/2017 14:00pm”^^xsd:date

5 IMPLEMENTATION OF MODEL

To facilitate interaction between the user and IoT devices, the ontological model described above was used in a home automation environment. This environment has a semantic interaction scenario, which consists of a set of smart objects equipped with sensors and actuators, as well as the necessary programming to make use of the ontology and expose their services so that others can consume them on first establishing a contract between those involved.

Performing reasoning on the ontology, it was possible to configure the services that smart objects provide automatically or with minimal user intervention. This is possible since smart objects are using hardware with capabilities to store and process ontologies in an operating system like Raspbian.

5.1 Design of the high-level system

To do this, the high-level design presented in Figure 7 was defined, which consists of:

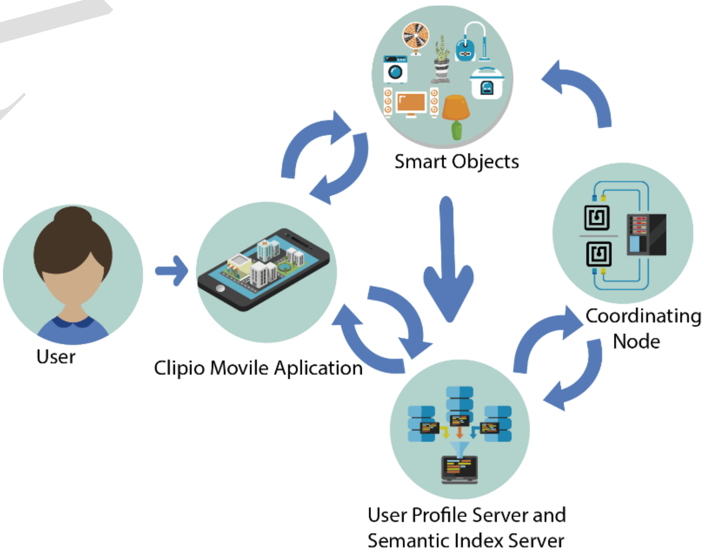

Figure 7: High-level design[5]

[5] Image adapted from https://www.freepik.es/

Source: own authorship

User Profile Server (UPS): this component makes it possible to access, share and reuse all ontologies of users registered in the system. Additionally, performs reasoning process based on the user's ontology is carried out in this component, to obtain new preferences. *UPS* performs most of the processing in the ontology, to take the load off the smart objects that have storage and processing restrictions. For example, it is in charge of storing the ontologies and carrying out complex queries on them, in addition to managing user authentication.
Semantic Index Server (SIS): this component stores all the information from the smart objects deployed in the system, from which each object obtains its respective metadata that describes it.
Smart Object (SO): all physical or virtual entity that can share any type of information about themselves and have processing capacity. All smart objects have the same source code and the difference between one and another lies in a module called Executables, which contains the logic to access sensors and actuators of each object. This makes it possible to create a new smart object quickly and easily by simply adding the necessary code to perform appropriate handling of sensors and actuators. If the user made any changes to their preferences, the smart object updates the ontology and sends it to the *UPS.*
Coordinating Node (CN): a device capable of detecting the presence of the user, sending a request to the UPS to download the ontology, and sending it to each of the objects that it coordinates. In this way, they personalize their services and activate them when the user is near, or the conditions specified in the user's preferences are met. This object is at the main entrance of the building and has two Near Field Communication (NFC) readers. One registers user input and the other output.
Clipio Mobile Application (CMA): An Android application to register in the system, and this information is then stored in the ontology. It also enables the control of smart objects and the creation of preferences if desired.
The operation of the system is shown in the following (See Figure 8). Everything begins when the user makes use of *CMA* by entering their personal, professional, and health information as well as the buildings and objects belonging to them, to open up the possibility of customizing the services with more accuracy. This information is stored in the UPS.

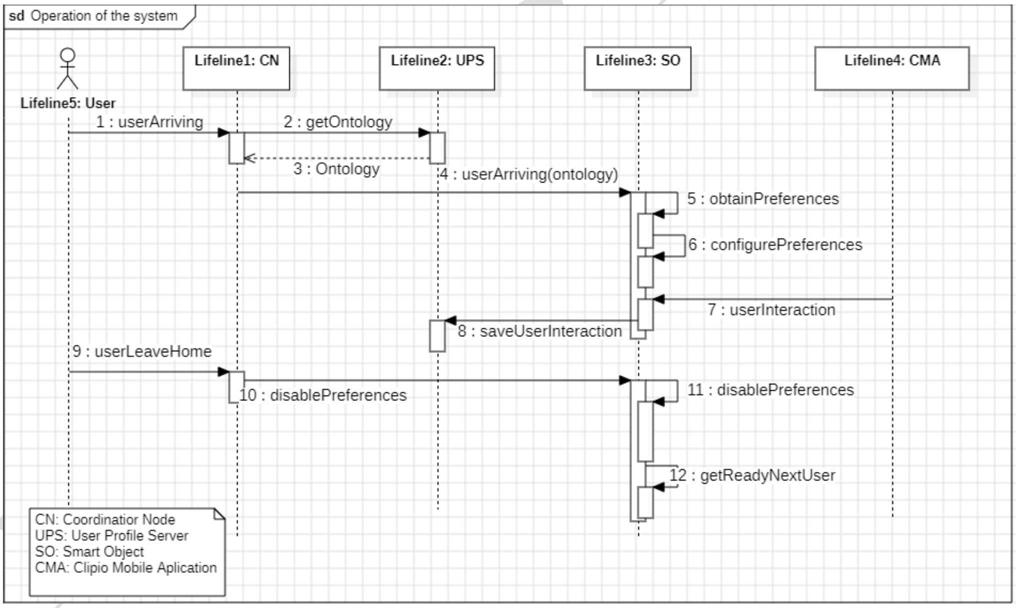

Figure 8: Operation of the system
Source: own authorship

Once the information is recorded, the user can interact with the smart objects. These interactions are stored in the user's ontology to find a pattern and be able to personalize the services.
When arriving at the building, the user makes use of an NFC tag, in which his identifier is stored, used by the Coordinating Object to retrieve the ontology of the user in question, making a web request to the UPS.
When the Coordinator receives the ontology, it verifies the objects that it coordinates and alerts them that the user has arrived home. Each object receives the ontology and makes a query to obtain the preferences belonging to that user and then activates them.

On leaving the house, the user uses his NFC tag again. The coordinator then sends the request to the objects to deactivate the preferences of this user, ready for the arrival of the next user.
An example is presented below using the smart object called "Light Regulator", which features a Light sensor, Proximity sensor, a Clock sensor, and a Lightbulb.
As mentioned above, the object is alert to the interactions with the user, who for this example comes into the room every day between 7 and 8 pm, always turning on the bulb via CMA. After a few days, the object registers this as a recurring behavior. It, therefore, creates a preference, which has the following format:
When it is around 7 pm, then turn on the light bulb. This preference will be activated as long as the user is at home.
The above is translated into a preference, which has three components: Event, in this case, it uses the clock; Condition, that the clock marks 7 pm; and Action, which turn on the bulb. This step will be carried out when the ontology arrives at the UPS, so that the next time the coordinator requests the ontology, it will contain the preference, and the object will load it when the ontology reaches it.

6 EVALUATION

The evaluation was based on the smart object semantic interaction scenario experiment proposed by Guerrero et al. [22], which presents three smart objects with home automation features in an office.
The previous scenario was updated in various aspects of its functionality, mainly in the management of ontologies, to facilitate interoperability, distribution, and reasoning of the user profile. Additionally, the internal architecture of the software was modified to allow communication by more than one protocol. It currently uses HTTP and MQTT, but other protocols, such as COAP, can easily be implemented, thereby expanding the architecture proposed by Guerrero et al. [22].
In the new scenario, four smart objects were created and programmed to adapt their services according to the characteristics and preferences of the user, through the developed ontology (Figure 9). The hardware used in the smart objects consists of:

- **Temperature Regulator:** Raspberry Pi 3B with Raspbian operating system, GROVE PI expansion board, GROVE MCP9808 temperature sensor, 2 GROVE v1 relays connected to a fan and heater.
- **Plant Moisture Regulator:** Raspberry Pi 3B with Raspbian operating system, GROVE PI expansion board, GROVE v1 soil moisture sensor, GROVE v1 LED to simulate plant watering.
- **Light regulator:** Raspberry Pi 3B with Raspbian OS, tsl2591 light sensor, grove v1 relay Raspberry
- **Coordinating Object:** Raspberry Pi 3B with Raspbian operating system, two MFRC522 RFID readers, 16x2 i2c display.

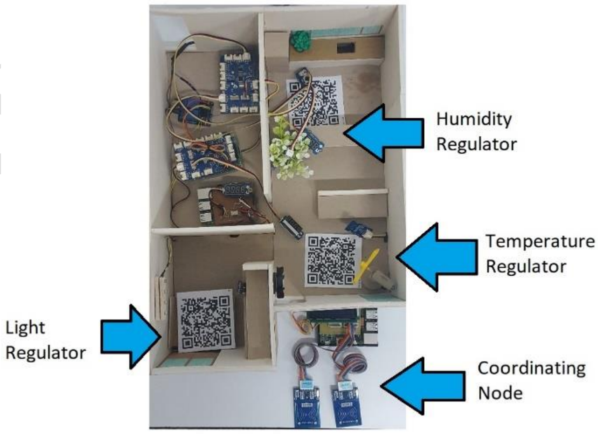

Figure 9: Scenario of semantic interaction of smart objects.
Source: own authorship

To properly systematize the evaluation, a case study was created that includes the previous scenario, following the methodology proposed by Runeson [27]. This presents four stages, described below:

6.1 Design

The objective of the study is to verify if from the information stored in the semantic model, the services in the IoT can be customized to offer a better user experience.

This case study was descriptive/holistic. Given that the context in which the test unfolds is a domotic environment in which we can have different types of users with different knowledge about the subject, the group was sought to be heterogeneous, selecting three types of users: those with only basic knowledge of the use of smartphones, systems engineering students and users with knowledge in IoT. The unit of analysis of the case study corresponds to the hardware/software system that implements the semantic model, measuring qualitative and quantitative aspects.

For the qualitative aspect, the opinion of the users about their interaction with the system is taken into account, for which a survey was designed as a survey instrument according to the Likert scale [28].

To monitor the activity of each smart object and the interaction between them, a software tool was used to store the created messages called LogPanel [24], which uses a MySQL database.

The information compiled from the instruments defined in the previous paragraph was analyzed using the following indicators:

Average Efficiency According to the User (AEAU): taken as the average of the rating (R_u) given by the n users regarding the perception of the time the system took to configure the services (Equation 1).

$$AEAU = \frac{\sum_{u=1}^{n} C_u}{n}$$

Equation 1. Average Efficiency According to the User (AEAU)

Average Global Configuration Time of the Scenario (AGCTS): average time the system took to configure all the services of a certain user (t_u) from the moment of arriving home to when the objects are ready to execute the services. This is obtained from the stored information of the n users in the LogPanel tool (Equation 2).

$$AGCTS = \frac{\sum_{u=1}^{n} t_U}{n}$$

Equation 2. Average Global Configuration Time of the Scenario (AGCTS)

Average of Adjustment of the Services to the Preferences of the User (AASPU): taken as the average of the given rating (R_u) by the n users regarding how satisfied they are in terms of the behavior of the services (Equation 3).

$$AASPU = \frac{\sum_{u=1}^{n} C_U}{n}$$

Equation 3. Average Adjustment of the Services to the Preferences of the User

Indicator of Effectiveness in Services (IES): reference is made to the number of services (preferences) configured by the system over the total of preferences established by the user. For the above, the preferences stored in the ontology are contrasted with those configured by the system, observing the records in the LogPanel tool (Equation 4).

$$\text{IES} = \frac{No.\ Services\ Config}{No.\ Services\ \text{that were expected}\ config} * 100$$

Equation 4 Indicator of Effectiveness in Services (IES)

Average User Intervention (AUI): taken as the average of the rating (R_u) given by the n users concerning the perceived ease when setting their preferences on the scenario. The facility is understood in this study as the minimum intervention made by the user to configure their services (Equation 5).

$$AUI = \frac{\sum_{u=1}^{n} C_u}{n}$$

Equation 5. Average User Intervention (AUI)

Average Service Coverage (ASC): consists of the number of services available over the number of services configured correctly according to the characteristics and preferences stored by the user (Equation 6).

$$ASC = \frac{Num.\ of\ available\ services}{num.\ of\ services\ configured\ correctly}$$

Equation 6. Average Service Coverage (ASC)

6.2 Preparation

In this stage, the hypotheses, and the variables to be measured in the study were identified (See Table 5), as well as their conceptual and operational definition. The hypotheses proposed were:

- H_1: the application of the user profile semantic model (UPSM) generates services suited to the preferences and characteristics of the users in an IoT scenario.
- H_2: the user profile semantic model (UPSM) contains the information necessary for the IoT scenario to be self-configured with minimal user intervention.
  Based on these hypotheses, the independent variable: *User profile model*, and the dependent variables: *Adequate Services* and *User intervention* emerge.

Table 5: Variables to be measured in the case study.
Source: own authorship

Independent variable	User profile semantic model
Conceptual definition	The User Profile Semantic Model is stored in each smart device and used to manage the information of the preferences and characteristics of the users, allowing the objects of an IoT Scenario to consult it to adapt their services.
Operational definition	The use of the Semantic Model in each Smart Object (SO) is activated or deactivated.
Dependent variable	Appropriate Services
Conceptual definition	A service is adequate if it fits the preferences of a user and is executed at the right time.
Operational definition	Check if the service was configured correctly (obtained from IES). Verify if the service was activated in the reference time (obtained from AGCTS). Verify if the user is satisfied with the execution time (obtained from AEAU). Verify if the user is satisfied with the behavior of the service (obtained from AASPU).
Dependent variable	User intervention level
Conceptual definition	The number of times user intervention has been requested to perform some configuration of the services.
Operational definition	The user makes the necessary configurations and gives an idea of how complex the process was (obtained from AUI).

6.3 Execution and Collection of Data

In this stage, an execution plan was designed consisting of eight steps, in which the users interacted in a controlled way with the scenario, and the developed instruments were applied.
The users who participated have the following characteristics: five students of Systems Engineering at the University of Cauca, four people with knowledge about the use of mobile applications, and a user with knowledge of IoT.
The execution and collection of information in the case study was carried out in two iterations. In the first iteration, the beta test of the scenario was carried out. From the data produced by this stage, a series of problems related to the usability of the mobile application and high response times were identified and corrected.
In the first iteration, it was not possible to configure all of the user's preferences, because the smart objects had to make complex queries presenting high response times. It was therefore decided to execute these queries in the UPS, improving response times and in this way configuring all of the user's preferences correctly.
Therefore, in the second iteration, response times of the scenario and user satisfaction in terms of the configured services improved. See Table 6.

Table 6: Total Results
Source: own authorship

Hypothesis	Indicator	Iteration I	Iteration II

H_1	AGCTS (seconds)	84.20	52
	AEAU (Likert 1 - 5)	3	3.6
	AASPU (Likert 0 -10)	7.8	9
	IES (Percent)	87	100
H_2	AUI (Likert 0 -10)	7.4	9.2

It must be borne in mind that the previous experiments were performed with the independent variable activated, that is, with the user profile model operating. This study did not experiment with the dependent variable deactivated, since the data provided by the study by Guerrero et al. [24] did not use the user profile model proposed in this paper.

Additionally, the scenario proposed by Guerrero et al. [24] did not contemplate the measurement of the previous hypotheses, therefore, it is not possible to make the comparative analysis quantitatively. However, a qualitative analysis can be made:

- User preference settings had to be done manually every time a user arrived at the deployed scenario. In contrast, in this proposal, only the configuration is made in the first interaction with the scenario. Subsequently, these configurations are automatically remembered by smart objects on identifying the user.
- Each time the user wanted to interact with the scenario, he had to activate his preferences and deactivate the preferences of previous users. On the other hand, in this proposal, the coordinating node identifies the user and proceeds automatically to activate the preferences in the smart objects of the scenario.
- In the previous cases, the configuration times of the scenario proposed by Guerrero et al. [24] were of the order of several minutes. In this proposal, however, the times oscillate around a minute.
- Users did not show a favorable position regarding their perception of the capacity of the system in configuring and executing their preferences, regarding the efficiency and effectiveness of the system deployed by Guerrero et al. [24].

6.4 Case study conclusions

In the first iteration, it was observed that the overall configuration time of the scenario was 84.20 seconds. Complementing this with the response that users gave regarding AEAU, we can infer that the user liked the configuration time; even though in iteration 2 the configuration time was improved to 52 seconds, the users maintained the same favorable perception. The foregoing shows that to achieve better user qualification, the reduction in the execution times of the scenario needs to be drastic, which could be achieved by optimizing the implementation of the scenario.

Regarding the Average adjustment of the services to the preferences of the user, it can be noticed that in the first iteration, the users gave a score of 7.8 (acceptable). Comparing this with the second iteration, the user's perception improved substantially, with a score of 9, i.e. an adjustment tending to excellent. This is because, for the second iteration, improvements were made in terms of the efficiency of the system to adjust the services according to the preferences of the user. Specifically, processing of the ontology of the user profile in terms of their preferences, which was done in the smart object itself, was transferred to the UPS.

Regarding the Indicator of effectiveness in services, in both iterations, the totality of the services requested by the users was configured. However, the fact that the configuration efficiency in the first iteration was not as good made the users think that the number of services requested had not been configured (87%). For the second iteration, users determined that 100% of the services requested were properly configured, this was due to the improvement made that was mentioned in the previous paragraph. Efficiency in the configuration and execution of services were therefore deemed to be important elements for users to feel comfortable in an IoT scenario that adjusts to their preferences.

It was possible to observe from the evidence found that H_1 is supported since it was possible to configure the preferences of the users using the proposed semantic model. However, user satisfaction is affected by the efficiency of the system in general.

Regarding user intervention to configure their preferences in the scenario, in the first iteration, only an acceptable rating was obtained (7.4), because users found difficulty when using the *CMA* interface, whereas in iteration 2 the rating increased to 9.2 tending to excellent. This is because the interface was improved using Human Computer Interface principles. This allowed the user to carry out the configuration of their services more efficiently (with fewer interactions with the interface). Thus, for H_2 we can say that the model provides the information necessary to configure the services according to the user's profile, reducing their intervention. However, a poorly designed interaction interface can generate an erroneous perception in the user concerning the ease of configuration of the services.

7 CONCLUSIONS AND FUTURE WORK

This research proposes an ontology that models the characteristics and particularities of users to personalize the services offered by objects of the IoT. The proposed ontology models dynamic and static characteristics, as well as the characteristics of the locations in which the user interacts with smart objects.

According to the results obtained previously, it can be verified that the application of the semantic user profile mode generates services adapted to the preferences and characteristics of users in an IoT scenario with acceptable times.

To configure the services of the users in the IoT scenario, the proposed model was found to adequately support the necessary elements for it. However, concerning the reduction of user intervention, this not only depends on the completeness of the model, but an important part lies in providing suitable interfaces for user interaction with the system. While the proposed model makes it possible to configure services adequately based on user preferences, in a scenario of smart IoT objects, more experiments aimed at the scalability of the system in terms of the number of services, objects, and users, need to be carried out, to establish the weaknesses and possible improvements to be made in the model.

The proposed model and its implementation are designed for spaces in which objects interact with only one person at a time. It is necessary to propose a solution in which there are several users in the same space and each one with different preferences. The reason for the above is that user preferences can generate conflicting services, for example, if one user wants the fan to turn on when the temperature is above 18 °C, and another user wants the fan to always be off. To resolve the above, it is necessary to establish automatic decision mechanisms to define which user to attend to where conflicts arise.

The intention of future work is the implementation of different reasoning techniques to deduce services more accurately, since the ontology stores both static and dynamic information of the user, as well as the information of the smart objects with which it is related. In addition, it evaluates the proposed model in a real scenario, which requires the knowledge of its users to be able to personalize its services.

8 CRediT AUTHORSHIP CONTRIBUTION STATEMENT

Yazmin Andrea Pabón Guerrero: Software, Validación, Análisis Formal, Investigación, Curación de datos, Escritura – Borrador original y Visualización. **Líder Julián Rojas Bolaños:** Software, Validación, Análisis Formal, Investigación, Curación de datos, Escritura – Borrador original. **Miguel Ángel Niño Zambrano:** Conceptualización, Metodología, Análisis formal, Escritura – Revisión y edición Supervisión y Administración del Proyecto.

FINANCIAL

This work was supported by the Universidad del Cauca, with a start date of 01/01/2021 and a completion date of 12/01/2022.

9 ACKNOWLEDGMENTS

The authors would like to thank the University of Cauca for their support in the execution of this project and the research and development group GTI.

10 REFERENCES

[1] K.-L. Skillen, L. Chen, C. D. Nugent, M. P. Donnelly, W. Burns, and I. Solheim, "Ontological user modelling and semantic rule-based reasoning for personalisation of Help-On-Demand services in pervasive environments," *Futur. Gener. Comput. Syst.*, vol. 34, pp. 97–109, 2014, doi: http://dx.doi.org/10.1016/j.future.2013.10.027.

[2] A. Wagner, J. L. V. Barbosa, and D. N. F. Barbosa, "A model for profile management applied to ubiquitous learning environments," *Expert Syst. Appl.*, vol. 41, no. 4 PART 2, pp. 2023–2034, Mar. 2014, doi: 10.1016/j.eswa.2013.08.098.

[3] C.-H. Lin, P.-H. Ho, and H.-C. Lin, "Framework for NFC-Based Intelligent Agents: A Context-Awareness Enabler for Social Internet of Things," *Int. J. Distrib. Sens. Networks*, vol. 2014, p. 16, 2014, doi: 10.1155/2014/978951.

[4] M. G. Kibria and I. Chong, "Context-awareness provisioning to support user-centric intelligence in Web

of Object platform," in *Information and Communication Technology Convergence (ICTC), 2015 International Conference on*, 2015, pp. 388–392, doi: 10.1109/ictc.2015.7354570.

[5] A. B. Rodriguez, A. R. Tena, J. Garcia-Alonso, J. Berrocal, R. F. Rosco, and J. M. Murillo, "Using Beacons for Creating Comprehensive Virtual Profiles BT - Ubiquitous Computing and Ambient Intelligence," 2016, pp. 295–306.

[6] J. Berrocal, J. Garcia-Alonso, C. Canal, and J. M. Murillo, "Situational-Context: A Unified View of Everything Involved at a Particular Situation BT - Web Engineering," 2016, pp. 476–483.

[7] Q. Wei and Z. Jin, "Service discovery for internet of things: a context-awareness perspective," *Proceedings of the Fourth Asia-Pacific Symposium on Internetware*. ACM, Qingdao, China, pp. 1–6, 2012, doi: 10.1145/2430475.2430500.

[8] S. S. Ara, Z. U. Shamszaman, and I. Chong, "Web-of-Objects Based User-Centric Semantic Service Composition Methodology in the Internet of Things," *Int. J. Distrib. Sens. Networks*, vol. 2014, p. 11, 2014, doi: 10.1155/2014/482873.

[9] D. Guinard, M. Fischer, and V. Trifa, "Sharing using social networks in a composable Web of Things," in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2010 8th IEEE International Conference on*, 2010, pp. 702–707, doi: 10.1109/percomw.2010.5470524.

[10] D. van Thuan, P. Butkus, and D. van Thanh, "A User Centric Identity Management for Internet of Things," in *IT Convergence and Security (ICITCS), 2014 International Conference on*, 2014, pp. 1–4, doi: 10.1109/ICITCS.2014.7021724.

[11] Y. Ashibani, D. Kauling, and Q. H. Mahmoud, "A context-aware authentication framework for smart homes," in *2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE)*, 2017, pp. 1–5, doi: 10.1109/CCECE.2017.7946657.

[12] L. Yao, "A Propagation Model for Integrating Web of Things and Social Networks," in *Service-Oriented Computing - ICSOC 2011 Workshops*, 2012, pp. 233–238.

[13] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context Aware Computing for The Internet of Things: A Survey," *Ieee Commun. Surv. Tutorials*, 2013, [Online]. Available: http://arxiv.org/abs/1305.0982.

[14] B. Kim, T. Kim, H.-G. Ko, D. Lee, S. J. Hyun, and I.-Y. Ko, "Personal genie: a distributed framework for spontaneous interaction support with smart objects in a place," *Proceedings of the 7th International Conference on Ubiquitous Information Management and Communication*. ACM, Kota Kinabalu, Malaysia, pp. 1–10, 2013, doi: 10.1145/2448556.2448653.

[15] G. Biamino, "A Semantic Model for Socially Aware Objects," *Adv. Internet Things*, 2012.

[16] L. Console *et al.*, "Interacting with social networks of intelligent things and people in the world of gastronomy," *ACM Trans. Interact. Intell. Syst.*, vol. 3, no. 1, pp. 1–38, 2013, doi: 10.1145/2448116.2448120.

[17] K.-L. Skillen, C. Nugent, M. Donnelly, L. Chen, and W. Burns, "Using Ontologies for Managing User Profiles in Personalised Mobile Service Delivery," in *Health Monitoring and Personalized Feedback using Multimedia Data*, A. Briassouli, J. Benois-Pineau, and A. Hauptmann, Eds. Cham: Springer International Publishing, 2015, pp. 245–264.

[18] Maria Golemati, Akrivi Katifori, Costas Vassilakis, George Lepouras, and C. Halatsis, "Creating an Ontology for the User Profile: Method and Applications," *Proc. First Int. Conf. Res. Challenges Inf. Sci.*, 2007.

[19] A. Otebolaku and G. M. Lee, "A Framework for Exploiting Internet of Things for Context-Aware Trust-Based Personalized Services," *Mob. Inf. Syst.*, vol. 2018, 2018, doi: 10.1155/2018/6138418.

[20] F. Corno, L. De Russis, and A. M. Roffarello, "How do end-users program the Internet of Things?," *Behav. Inf. Technol.*, vol. 41, no. 9, pp. 1865–1887, 2022, doi: 10.1080/0144929X.2022.2071169.

[21] S. P. Gochhayat *et al.*, "LISA: Lightweight context-aware IoT service architecture," *J. Clean. Prod.*, vol. 212, pp. 1345–1356, 2019, doi: 10.1016/j.jclepro.2018.12.096.

[22] H. Ning, F. Shi, T. Zhu, Q. Li, and L. Chen, "A novel ontology consistent with acknowledged standards in smart homes," *Comput. Networks*, vol. 148, pp. 101–107, Jan. 2019, doi: 10.1016/J.COMNET.2018.11.004.

[23] M. A. Niño-Zambrano, "Interacción semántica de objetos en la web de las cosas," Universidad del Cauca, Popayán - Cauca - Colombia, 2016.

[24] S. Guerrero-Narváez, M. Niño-Zambrano, D.-J. Riobamba-Calvache, and G.-A. Ramírez-González, "Test bed of semantic interaction of smart objects in the web of things," *Futur. Internet*, vol. 10, no. 5, 2018, doi: 10.3390/fi10050042.

[25] O. Corcho, M. Fernández-López, A. Gómez-Pérez, and A. López-Cima, "Building legal ontologies with METHONTOLOGY and WebODE," *Law Semant. Web*, 2005, [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-540-32253-5_9.

[26] D. Bonino and F. Corno, "DogOnt - Ontology Modeling for Intelligent Domotic Environments," in *The*

Semantic Web - ISWC 2008: 7th International Semantic Web Conference, ISWC 2008, Karlsruhe, Germany, October 26-30, 2008. Proceedings, A. Sheth, S. Staab, M. Dean, M. Paolucci, D. Maynard, T. Finin, and K. Thirunarayan, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 790–803.

[27] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," *Empir. Softw. Eng.*, vol. 14, no. 2, p. 131, 2008, doi: 10.1007/s10664-008-9102-8.

[28] R. H. Sampieri, C. F. Collado, M. del P. B. Lucio, and McGRAW-HILL, *Metodología de la Investigación*, Sexta Edic. 2014.