The Journal of Systems & Software 144 (2018) 295–313

Contents lists available at ScienceDirect

The Journal of Systems & Software

journal homepage: www.elsevier.com/locate/jss

Supporting end users to control their smart home: design implications from a literature review and an empirical investigation

Danilo Caivano[a], Daniela Fogli[b], Rosa Lanzilotti[a], Antonio Piccinno[a,*], Fabio Cassano[a]

[a] Dipartimento di Informatica, Università degli Studi di Bari Aldo Moro, Bari, Italy
[b] Dipartimento di Ingegneria dell'Informazione, Università degli Studi di Brescia, Brescia, Italy

ARTICLE INFO

Keywords:
Ambient intelligence
Internet of things
Smart home
Rule-based paradigm
End-user development

ABSTRACT

Designing tools that allow end users to easily control and manage a smart home is a critical issue that researchers in Ambient Intelligence and Internet of Things have to address. Because of the variety of available solutions, with their advantages and limitations, it is not straightforward to understand which are the requirements that must be satisfied to effectively support end users. This paper aims to contribute to this topic through a systematic and rigorous activity based on two main pillars of the empirical research in software engineering: i) a literature review addressing design and evaluation of tools for smart home control oriented to end users, and ii) an experimental study in which three tools, that emerged from the literature review as the most suitable and widespread, were compared in order to identify the interaction mechanisms that end users appreciate most. On the basis of the obtained results, a set of design implications that may drive the development of future tools for smart home control and management are presented.

1. Introduction

Ambient Intelligence (AmI) and Internet of Things (IoT), with their specific application to the *smart home*, are becoming reality (Atzori et al., 2010; Sadri, 2011), and a plethora of smart home controllers and smart devices, like light bulbs, thermostats, humidity sensors, locks, plugs, cameras, etc., are now available on the market. Researchers keep studying new intelligent behaviors for such smart systems, designing novel software architectures or reengineering existing ones to deal with the data (and big data) these systems generate (Bianchi et al., 2000), trying to enhance existing software processes (Caivano, 2005; Caivano et al., 2001) or investigating new solutions, for example, for network and data security, just to name current issues in AmI and IoT. However, a very critical topic is concerned with the creation of tools for smart home control and management that address the needs of non-technical people, in order to make this technology widely available to anyone (Barricelli and Valtolina, 2015; Brich et al., 2017; Coutaz and Crowley, 2016; Desolda et al., 2017; Ghiani et al., 2017). Indeed, making all people able to install, configure, and use the devices of a smart home is fundamental for the success of any hardware/software solution in this field, regardless its sophistication in terms of algorithms or architecture.

The research reported in this paper addresses this issue. Several software tools have been proposed so far, whose goal is to make home inhabitants able to manage their smart home, i.e. to allow them to configure the various devices of the home, by modifying their behavior over time according to family's preferences and changing habits. Our research aims at analyzing such tools and at identifying those ones that, at least partially, fulfil the above goal, by highlighting the interaction mechanisms that users appreciate most, so that design implications for creating software tools that allow end users to easily operate on the smart home can be identified.

With the term *end user* we refer to the increasing number of non-technical people that, without being expert in computer technologies or programming activities, nor willing to be, want to use computer systems for their daily activities, i.e., for work as well as for entertainment or other purposes (Ardito et al., 2012). The research presented here capitalizes on years of experience in End-User Development (EUD) (Cabitza et al., 2014; Costabile et al., 2003, 2006, 2007), a discipline that “encompasses methods, techniques, methodologies, situations, and socio-technical environments that allow end users to act as professionals in those domains in which they are not professionals” (Fischer et al., 2017). Examples can be found in several application domains, including the case of smart software systems through which home inhabitants are enabled to create and control their smart home by using cheap off-the-shelf devices, smart objects and smartphones. In most cases, home inhabitants are examples of end users as described

* Corresponding author.
E-mail addresses: danilo.caivano@uniba.it (D. Caivano), daniela.fogli@unibs.it (D. Fogli), rosa.lanzilotti@uniba.it (R. Lanzilotti), antonio.piccinno@uniba.it (A. Piccinno), fabio.cassano1@uniba.it (F. Cassano).

https://doi.org/10.1016/j.jss.2018.06.035
Received 31 October 2017; Received in revised form 2 June 2018; Accepted 13 June 2018
Available online 23 June 2018
0164-1212/

above, thus they require tools adequate to them. The workshop on End User Development in the Internet of Things Era (EUDITE), held at the International Conference on Human Factors in Computing Systems (CHI 2015) (Tetteroo et al., 2015), has inspired the work reported in this article.

The objective of our research is to empower home inhabitants to adapt the behavior of their home to their needs. This means that such end users should be enabled not only to create simple commands (e.g., "at 7 a.m. rise shutters") but also they should be provided with powerful mechanisms that, without requiring any programming activity, allow them to monitor dangerous situations (e.g., to check gas leak, intrusions, etc.), carry out tele-assisted services, and perform environment management tasks (e.g., temperature or lighting setting).

This paper contributes to the research on IoT for the smart home in several ways.

- First, it presents a review of the literature from 2010 to 2016 aimed at identifying those tools that appear to best support end users in managing and controlling smart devices. From the literature review, it emerged that a variety of tools for smart home control have been proposed so far. However, most of tools could not be deeply evaluated, due to their nature as research prototypes. A high-level analysis was then performed on the selected tools, based on some characteristics and design principles for smart home control. This analysis indicated three tools as suitable for an additional investigation: Atooma,[1] IFTTT,[2] and Tasker.[3]
- A further contribution comes from the experimental study performed to compare these three tools by involving real users, in order to identify the interaction mechanisms that end users find more usable for managing the rules through which they control the smart home.
- Both the literature review and the experimental study were instrumental to derive implications for the design of future tools supporting non-technical people to install, configure and use the devices of a smart home.

The paper is structured as follows. Section 2 presents related work with respect to the three main contributions of this paper. Section 3 illustrates the planning phase, the conducting phase and the results of the literature review, as well as how we addressed the threats to its validity. Section 4 describes the experimental study carried out to compare Atooma, IFTTT and Tasker. Section 5 analyses the potential threats to the validity of the comparative study, by discussing how they were minimized. Section 6 presents the design implications derived from the literature review and the experimental study. Finally, Section 7 concludes the paper.

2. Related work

In the following, we analyze the related work with reference to the main contributions of our paper; therefore, we first discuss existing surveys and reviews on IoT and AmI; then, we consider experimental studies of tools proposed to address end users' needs in smart home control, by paying attention to comparative studies; finally, we consider literature work that propose design principles and guidelines in this research field.

2.1. Literature reviews in the IoT and AmI research areas

Literature reviews in the IoT research field have been proposed to analyze enabling technologies, communication protocols, architectures and application issues (Al-Fuqaha et al., 2015; Atzori et al., 2010; Gubbi et al., 2013). The paper (Atzori et al., 2010) is probably one of the first contribution in this direction: it analyzes IoT as a complex scenario resulting from the synergy of different research areas (e.g., telecommunications, computer science, electronics, etc.); then, it presents the Service Oriented Architecture (SOA) approach for the IoT middleware and discusses the enabling technologies available at that time (before 2010); finally, it puts the attention to the potentialities offered by IoT for the development of applications in a variety of domains, from transportation, to healthcare, from smart environments (including the home) to the personal and social domain. The key enabling technologies and most interesting application domains are deepened in (Gubbi et al., 2013), with specific attention to cloud-based implementation of proposed solutions. Al-Fuqaha et al. (2015) provide a thorough summary of the most relevant protocols and application issues, by presenting case studies to illustrate how the different protocols fit together to deliver desired IoT services; they also explore the relationship between IoT and other emerging technologies (e.g., big data analytics and cloud computing).

The above works do not take into consideration tools and user interfaces specifically oriented to end users; in other terms, there is not any human-computer interaction (HCI) perspective discussed in them. Other interesting surveys, which partially consider this aspect, are those presented in the AmI research field. For example, Cook et al. (2009) organize their review of AmI technologies in five areas: sensing, reasoning, acting, privacy and security, and human-computer interaction. As far as the last area, the authors discuss technologies and proposals for context-aware computing and natural interfaces, namely all those interfaces that mimic human-to-human communication through motion tracking, gesture recognition, emotion recognition, speech processing, and so on. However, the issue related to user's intervention in the direct modification of smart environment's behavior is not dealt with in the review. The survey by Sadri (2011) is structured around the variety of AmI projects proposed in the literature, and focuses on the artificial intelligence techniques aimed at endowing an environment with intelligent behaviors, such as user profile learning, recognition of recurrent users' behaviors, agent and multi-agent architectures, affecting computing. In particular, it is observed how rule-based approaches are often adopted to implement inference techniques and behavior selection strategies, particularly in the field of smart homes for elderly care. However, also in this work, little space is left to user-oriented tools for smart environment control.

This paper focuses on this issue by considering the smart home as the paradigmatic smart environment, whose behavior should be configured and adapted by its inhabitants. We will observe that the rule-based approach is usually adopted to provide users with interfaces for smart home control; most of these tools are inspired to iCAP, one of the first works that proposed trigger-action programming (i.e., "if-then" rules) to support the personalization of smart home behavior (Dey et al., 2006).

2.2. Experimental studies

Several tools for IoT and smart home control have been experimented with real users. One of the most cited studies on IFTTT is that presented in (Ur et al., 2014). It illustrates a usability test with 226 participants and demonstrates that the average user of IFTTT can successfully create simple trigger-action rules for smart home management. Coutaz and Crowley (2016) have recently designed and experimented in real-world conditions a system for smart home control, called AppsGate. This system has been installed by the researchers in different households, which have then provided their opinions and suggestions for improvement in subsequent interviews. The two authors, who are married in real life, personally deployed AppsGate in their home, and provided their own impressions about the system. The article by Brich and colleagues describes the results of a contextual inquiry study

[1] http://support.atooma.com/.
[2] https://ifttt.com/.
[3] http://tasker.dinglisch.net/.

carried out in 12 households, which compares rule-based and process-oriented notations (Brich et al., 2017). The results showed that rule-based notations are usually enough to express simple automation tasks but not sufficiently powerful to support complex ones. In addition, participants using the process-oriented notation appreciated its expressiveness and its capability to manage more complex tasks involving several devices in their homes. Fogli et al. (2017) present a comparison between Apple Home and ImAtHome, a prototype application developed by the authors; the experiment, carried out with 30 users according to a between-subject protocol, demonstrates that ImAtHome is perceived by the users as easier to use and to learn, even though their performances were pretty good in both cases.

There are also studies that aimed at comparing different interaction paradigms or even different tools, in order to identify the most preferred by end users. Among them, Dahl and Svendsen (2011) performed a comparison among three paradigms for rule composition, namely filtered list, wiring composition and jigsaw puzzle composition. The most intuitive paradigm resulted to be the filtered list, where condition-action compositions are obtained by selecting conditions and actions from available lists. In addition, participants considered jigsaw puzzle composition the most playful and engaging type of interaction.

Alternative notations have been recently explored in (Desolda et al., 2017), which addresses the more general problem of smart-object composition in IoT scenarios, trying to orchestrate the behavior of multiple devices to create new services. Desolda and colleagues proposed a new model for composing rules, based on the classic questions "Who?", "What?", "When?", "Where?" and "Why?" ("5W"). Starting from this model, and inspired by some of their previous works, (Ardito et al., 2015; Desolda et al., 2016), they implemented three visual composition mechanisms for expressing Event-Condition-Action (ECA) rules, whose usability was tested during a comparative study (Desolda et al., 2017). The findings highlighted that some features of the evaluated composition paradigms suit the expertise of end users. Moreover, the study identified the proposed "5W" paradigm as the preferred candidate paradigm among those analyzed, even if it would require some adaptations when adopted in specific domains in order to improve its effectiveness. The authors also extended this approach to support users in synchronizing the behavior of ecologies of smart objects. The proposed approach consists in visual mechanisms that guide domain experts in transferring their knowledge to the smart objects by defining a semantic layer on top of them. Such knowledge consists in a set of custom attributes that can be used to simplify the definition of ECA rules.

The study carried out by Lucci and Paternò, which compared Tasker, Locale and Atooma, found that even though Tasker resulted the best tool in terms of expressiveness, Atooma was the one with the highest number of successful performances, resulting thus easier to use than Tasker and Locale (Lucci and Paternò, 2014). More recently, the study presented by Cabitza and his colleagues focused on Atooma and IFTTT and involved a sample of young participants (students in economics and computer science disciplines) (Cabitza et al., 2017). The results of that study showed that no difference existed between IFTTT and Atooma as far as efficiency and effectiveness are concerned, but participants were more satisfied with the Atooma interface than with the IFTTT one. In the comparative study presented in this paper, we will discuss the additional evidences that we collected about the usability of Atooma, IFTTT and Tasker in allowing end users to compose and manage trigger-action rules; furthermore, we will underline lacks and drawbacks that we identified in these tools and that future design of systems for smart home control should avoid.

2.3. Design principles and guidelines

In the ethnographic study carried out with real families by Davidoff and colleagues (Davidoff et al., 2006) it is highlighted how user interfaces for smart home control are usually designed for one single user. However, usually more than one person inhabits a house, and thus a multi-user approach is advocated, in order to deal with household collaborative activities. As a consequence, seven design principles for the design of smart homes are proposed in (Davidoff et al., 2006). They represent a seminal work in this area, since they are general enough to be applied in a variety of systems for smart home control. For this reason, such principles are adopted in this paper to analyze the tools identified through our literature review, and they will be illustrated in detail in Section 3.1.

The results of the study described in (Coutaz and Crowley, 2016) provides insights about the most important features for end users, such as adaptation to one's own needs, integration with the rest of the home, easy management for seasonal programs, periodic change of programs and deletion of obsolete programs/rules, availability of multimodal interfaces and machine learning features, possibly differentiated according to the users' roles.

Additional issues to be taken into consideration during design are listed in (Brich et al., 2017), such as the need of debugging and simulation features, conflict resolution support, privacy and security issues, and the importance of mediating between user's need of control and lack of programming experience. The latter means that user interfaces should be suitable to less tech-savvy users, but, at the same time, users should be able to understand and control their environment.

Other design implications discussed in (Desolda et al., 2017) recall those suggested in (Brich et al., 2017; Coutaz and Crowley, 2016), such as the accommodation of different user's skills through different complexity levels, and the management of access policies that allow defining proper restrictions on object access by users. Furthermore, other design implications are derived from the experimentation of their novel paradigm for smart object composition in IoT scenarios: the need of an adequate expressive power and representation of composition rule; the importance of avoiding too much constraints on the order of rule element specification; the usefulness of assistance mechanisms, such as the possibility of creating groups of rules associated to typical scenarios of use, the availability of pre-defined system configurations, and context-aware recommendations.

Ghiani et al. (2017) describe the design of an authoring environment that allows end users without programming experience to create complex trigger-action rules for different IoT applications. System design capitalizes on literature analysis and direct interactions with stakeholders in different domains, which allowed researchers to identify a set of requirements. Most of them recall the issues underlined by other research studies (Brich et al., 2017; Coutaz and Crowley, 2016; Desolda et al., 2017). In addition, they highlight the importance of providing users with multimodal user interfaces and user-oriented terminology.

Finally, Sas and Neustaedter (2017) present a study of Do-It-Yourself (DIY) practices in smart homes. Findings suggest three interesting implications: designing complex but transparent technologies, which can be tinkered with (i.e., played, repaired, or improved) by inhabitants, designing transparent DIY kits for making them, standardizing communication protocols.

3. Literature review

Our literature review addressed tools for IoT and smart home control that support end users in installing, controlling and using such tools. Adapting the approach for systematic mapping studies presented in (Bafandeh Mayvan et al., 2017), the review process was structured in two main phases. In the first phase, named *planning phase*, researchers define all the protocols and strategies needed for conducting the review. In particular, the most important part of this phase is defining the *research question* to answer in the review process, namely the question that determines the goal of the review. Next, the search process has to be specified: it includes the protocols and decisions that are relevant to conduct the review, such as the definition of the *search string* and the *search strategy*. The selection criteria have also to be defined: they encompass *inclusion* and *exclusion criteria*, and the *selection strategy* to identify relevant papers referring to the tools that answer the research question. The second phase is called *conducting phase*, in which the protocols and the strategies defined in the planning phase are applied. Preliminary results of the literature review discussed in this section were presented in (Fogli et al., 2016a).

The following two sub-sections describe these phases, while Section 3.3 presents the results of the review and Section 3.4 discusses the threats to the validity and the solutions we adopted to reduce them.

3.1. Planning phase

The *planning phase* of our literature review is illustrated in this section by describing all the components from the research question to the selection strategy.

- *Research question.* Which are the software tools that allow end users to easily manage and configure the behavior of a smart home?
- *Search string.* The search string is specified on the basis of the following main and related concepts:
  - Main concepts: *tool; smart home; internet of things.*
  - Related concepts: *device, instrument, appliance, gadget; smart house, home automation, digital home; IoT.*

  Consequently, the following search string is defined:

```
((tool OR device OR instrument OR appliance OR gadget)
AND ((smart OR digital OR automation) AND home OR
house))
AND (internet of things OR iot))
```

- *Search strategy.* The search string is used to automatically search title, abstract and keywords of the publications that appear in the journals listed in Table 1 and in the conference proceedings listed in Table 2, which are the most relevant for our goal. Indeed, most of the selected journals and proceedings represent publication venues focused on EUD, IoT and AmI; furthermore, Tables 1 and 2 include the most high-ranking journals and widely-known conferences in HCI. The search is performed using the Scopus database, which includes all publications in Table 1 and Table 2.

A manual search is also performed on the proceedings of the CHI 2015 workshop on End User Development in the Internet of Things Era (EUDITE), which are especially relevant with respect to IoT and EUD topics, but these proceedings are not indexed by Scopus.

We consider publications from 2010 to 2016, because the needs and expectations of non-technical people in domains like IoT and smart home are addressed primarily in recent years.

- *Inclusion and exclusion criteria.* A retrieved publication is considered only if it meets all the following inclusion criteria:
  1. The publication presents a tool that can be used for the management of a smart home.
  2. The publication describes a tool oriented to end users.
  3. The publication describes a tool capable to interact with other systems and devices.

  Moreover, a publication is not considered if the following exclusion criterion is satisfied:
  1. The publication presents a low-fidelity prototype[4] or a tool not available for evaluation.
- *Selection strategy*. To identify relevant publications that describe the tools answering our research question, the following selection strategy, consisting in two steps, is defined. As first step, title, keywords and abstract of the publications retrieved by the automatic and manual search is considered and the decision if they refer to a tool of interest is made by taking into account both inclusion and exclusion criteria. If title, keywords and abstract do not contain enough information to take a decision, the full-text of the retrieved publication is considered. The relevant publications are analyzed in order to identify the tools answering our research question.

Table 1
Journals searched in the review.

Journals
- ACM Transactions on Computer-Human Interaction (ACM)
- Future Generation Computer Systems (Elsevier)
- IEEE Pervasive Computing (IEEE Computer Society)
- International Journal of Ad Hoc and Ubiquitous Computing (Inderscience Enterprises)
- International Journal of Human-Computer Studies (Elsevier)
- International Journal of Pervasive Computing and Communications (Emerald Group Publishing)
- International Journal of Smart Home (Science and Engineering Supporting Society)
- Journal of Ambient Intelligence and Smart Environments (IOS Press)
- Journal of Systems and Software (Elsevier)
- Personal and Ubiquitous Computing (Springer Verlag)
- Pervasive and Mobile Computing (Elsevier)
- Sensors (MDPI)

Table 2
Conference proceedings searched in the review.

Conferences
- ACM Conference on Human Factors in Computing Systems (CHI)
- ACM Conference on Ubiquitous Computing (UbiComp)
- European Conference on Ambient Intelligence (AmI)
- IEEE International Conference on Computational Science and Engineering (ICCSE)
- IEEE International Conference on Green Computing (GreenCom) and Communications and IEEE Internet of Things (iThings) and IEEE Cyber, Physical and Social Computing (CPSCom)
- IEEE International Conference on Pervasive Computing and Communications (PerCom)
- IEEE International Conference on Trust Security and Privacy in Computing and Communications (TrustCome)
- IEEE International Conference on Ubiquitous Computing and Communications (IUCC)
- International Conference on Embedded and Ubiquitous Computing (EUC)
- International Conference on Frontier of Computer Science and Technology (FCST)
- International Conference on Future Internet of Things and Cloud (FiCloud)
- International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS)
- International Conference on Mobile and Ubiquitous Systems Computing Networking and Services (MOBIQUITOUS)
- International Conference on the Internet of Things (IoT)
- International Conference on Ubiquitous and Future Networks (ICUFN)
- International Symposium on End-User Development (IS-EUD)
- International Symposium on Pervasive Systems Algorithms and Networks (I-SPAN)

Each selected tool is first classified according to some characteristics that focus on main features for allowing end users to manage different smart devices in a smart home. Hints for the definition of the characteristics come from the analysis of recent literature discussed in Section 2 (e.g., see (Brich et al., 2017; Coutaz and Crowley, 2016; Desolda et al., 2017; Ghiani et al., 2017)). The chosen characteristics are the following:

C1. *License and price*, because the tools should be available to all end users and thus cheap solutions are preferable.
C2. *Flexibility in device management* to allow end users to easily add/modify/remove devices.
C3. *Easy device behavior specification,* to allow end users to control devices.

[4] "A prototype that is sketchy and incomplete, that has some characteristics of the target product but is otherwise simple, usually in order to quickly produce the prototype and test broad concepts (from http://www.usabilityfirst.com/glossary/low-fidelity-prototype/).

C4. *Tool extensibility*, e.g., through plug-ins, for integrating the tool with other systems that possibly already exist in the house.

C5. *Technical support by manufacturer*, in order to help end users solve possible technical issues.

C6. *Integration with different smart devices*, also produced by different manufacturers, to allow the interconnection of the tool with other smart devices that might be in the house.

C7. *Interoperability*, to allow the communication with possible web services already used by home inhabitants, such as web mapping systems, email applications, file sharing services, weather applications, etc.

Since our aim is addressing most of the end users with low-cost solutions, without forcing them to acquire expensive or complex technical solutions, once the tools are classified according to the above characteristics, we will select only those tools that are free of charge (i.e., open source, freeware, commercial providing a free version or a trial version). The resulting set is then analyzed by considering the first six design principles for smart home control presented in (Davidoff et al., 2006), which are very much referred in the literature. A further principle proposed by Davidoff et al. (2006), namely *Participate in the construction of family identity*, is neglected since it is not easy to verify in a limited time, but it would require a longer longitudinal study. The six principles are listed in the following:

P1. *Allow for the organic evolution of routines and plans*. A smart home tool should support the definition of routines and plans describing tasks that home inhabitants carry out daily, weekly, or seasonally, and allow them to modify such routines and plans on the basis of their changing needs.

P2. *Easily construct new behaviors and modify existing behaviors*. The tool should enable the home inhabitants to define new routines and plans or to modify the existing ones in a simple way, so that end users do not have to acquire any technical or programming skills.

P3. *Understand periodic changes, exceptions and improvisation*. The tool should be able to recognize exceptions to the home inhabitants' routines and to deal with improvisation when occasionally users change their own behavior.

P4. *Design for breakdowns*. The tool should be able to recognize when a breakdown is occurring, as well as the cascade of failures that it can determine, and promptly react to them.

P5. *Account for multiple, overlapping and occasionally conflicting goals*. The tool should deal with collaborative tasks of home inhabitants, and flexibly react to possible conflicts that may occur among preferences of different members or different methods for task completion.

P6. *The home is more than a location*. The tool should consider inhabitants' needs and desires emerging when they are outside home (e.g., at work, at school, at gym, during transportation between any of these locations). Thus, the tool should allow them to enter and update information from these various locations as easily as if they were at home.

3.2. Conducting phase

This section describes how the literature review was conducted on the basis of what was planned. The conducting phase started with the processing of the search string in the Scopus digital library, considering the journals in Table 1 and conference proceedings in.

Table 2. Next, the manual search in the EUDITE workshop proceedings was performed since the publications presented at the workshop, which actually inspired our research, were not indexed by Scopus. Both automatic and manual searches retrieved a total of 1057 publications, which were then analyzed by applying inclusion and exclusion criteria to title, keywords, and abstract, and even to the full-text when title, keywords and abstract did not provide enough information for selecting the publications relevant to our goal. The selection process was carried out by two researchers, who individually analyzed the retrieved papers. Double-scoring was conducted on 65% of the resulting papers, yielding an inter-rater reliability value greater than 0.85. Discrepancies were solved during a consensus meeting of the two researchers. This process led to the exclusion of 1009 papers; thus, the publications eventually selected were 48 (their references are listed in Appendix A).

A preliminary analysis of the tools presented in the selected publications showed that they all adopt the *event-condition-action* (ECA) rule paradigm and thus provide user interfaces for trigger-action programming (i.e. if-then rules) (Pane et al., 2001). In other words, such tools support end users (home inhabitants) in the composition of triggers (i.e., events and/or conditions) with actions that a device has to perform, using rules like 'if-condition(s)-then-action(s)' or 'when-event (s)-then-action(s)'. For this reason, for classifying the tools we decided to consider a further characteristic:

C8. *Flexibility of rule for behavior configuration*. Indeed, flexibility of the tool in managing the rules allows home inhabitants to better control the behavior configuration.

The classification process according to the characteristics and the design principles for smart home control was conducted by two researchers. First, each researcher independently analyzed each tool and proposed a classification. Also in this case, the inter-rater reliability on identified themes was satisfactory (>0.85). In a face-to-face meeting, all differences were solved by discussion.

3.3. Resulting tools

As reported in the previous section, 12 tools resulted from the literature review. They are listed and shortly described in Table 3.

Table 4 reports the results of the classification of tools according to the 8 characteristics, where each characteristic is associated with one of the following symbols: "✓" if the characteristic is totally satisfied; "~" if it is partially satisfied; and "×" if it is not satisfied. This table shows that characteristics C2, C3, C4, C5, C7 and C8 are satisfied by almost all tools, while C6 is the characteristic less satisfied by the tools; C1 regards the licence and price.

The classification based on the design principles for smart home control, reported in Table 5, was performed on the reduced set of tools, which are free of charge or commercial providing a trial version. In order to perform this classification, a typical scenario describing how a family could manage and live in its smart home was defined. In addition, it was assumed that each tool was installed both on a central control device (located in the house and connected to the Internet) and on all mobile devices of the family members (also connected to the Internet, even outside the house).

By looking at Table 5, it is evident that Atooma and IFTTT are pretty equivalent, since, besides satisfying P1 and P2, both tools partly meet the principle P6, because they allow detecting the position of the user, while the remote control is absent. However, if we consider their characteristics (see Table 4), IFTTT has more limitations regarding the flexibility in the management of devices and technical support than Atooma; moreover, being created primarily as a web service, IFTTT also requires a constant Internet connection to correctly work. In addition, Atooma allows managing multiple conditions and actions and has a specific area for the smart object management.

We Wired Web satisfies principle P1 only, that is, it supports some form of organic evolution of routines and plans. Indeed, even though it offers a variety of possibilities for tuning triggers and actions, it does not allow end users with limited programming experience to create rules in an easy way (principle P2). Bipio provides the user with this possibility, but it does not satisfy the other four principles. Although itDuzzit is also lacking with respect to design principles P3, P4, P5 and

Table 3
Description of the twelve tools identified through the review.

Tool	Description
Atooma ('A TOuch Of Magic') http://support.atooma.com/	A free Italian mobile application that allows users defining new behaviors by creating rules, according to the ECA paradigm, which can encompass up to five conditions and five actions. It can be integrated with other mobile applications, web services and devices. Developers can use its Software Developer Kit (SDK) to create add-ons.
Bip.io https://github.com/bipio-server/bipio	A web automation framework that exploits the graph metaphor for wiring web services represented as nodes. When nodes are connected by means of arrows, assisting procedures guide users in defining trigger and action properties. Software developers may implement new extensions.
GALLAG Strip http://gallag.wikispaces.asu.edu/	A mobile application that allows users to create sensor-based context-aware rules according to the ECA paradigm. It implements the programming by demonstration (Cypher, 1993) technique in order to enable users to test their rules as they program them. GALLAG-compatible sensors and mobile devices can be added by developers to be used by end users to create new rules.
IFTTT (If This Then That) https://ifttt.com/	A free web and mobile application that allows creating "if-then" rules combining social networks, web services, and smart things. It presents a good integration with any kind of web service. The user can define for each rule only one condition and only one action. Internet connection is needed to define and run rules.
itDuzzit https://developer.intuit.com/itduzzit	A web cloud platform with a rule composition paradigm very similar to IFTTT that lets users build their own custom integration solutions (called "duzzits"). A duzzit can be deployed as an online form, a job, a web service, an email drop box, or a widget. Pre-built duzzits available in the Duzzit Library can be modified through the Duzzit Editor.
Locale http://www.twofortyfouram.com/	An Android application for the automated management of a mobile device. The user can define rules according to the ECA paradigm, but they may include conditions and actions only related to mobile features (the position and orientation of the smartphone, the date and time, the remaining battery power, etc.). It can be extended through external plug-ins to permit integration with web services and other devices connected to the Internet.
Ninjablocks https://ninjablocks.com/	A cloud-based platform consisting of a physical sphere ("Ninja sphere") able to interact with a variety of sensors and actuators, and compatible with other services, such as SMS, email, and Arduino. It implements the ECA paradigm, and the users can choose triggers and actions among limited sets; but a rule can consist of any combination of any of those triggers and actions. Developers may exploit the SDK to extend system functionalities.
Tasker http://tasker.dinglisch.net/	An application for Android that supports the creation of rules with multiple conditions and multiple actions. It allows composing apps and functions available on mobile devices. It makes available more than 200 actions and it is well integrated with many web services and smart devices through specific plug-ins.
Twine http://supermechanical.com/twine/	A physical smart object including standard accelerometers, thermometers, and other sensors, which can be programmed to alert the user in case of problems. It is thought to be left anywhere in the house to collect data. Additional sensors detect floods, leaks, opened doors, and signals from other home systems. "If-then" rules, to control and manage the sensors, can be defined or modified through a web application.
WigWag http://www.wigwag.com/	An open source platform, including a mobile application and a web-based interface, that allows users to monitor and configure devices in a smart home. Users can create "if-then" rules that control the connected devices in a home or office, whether they are right there or far from these places.
We Wired Web http://wewiredweb.com/	An integration-as-a-service web application that supports rule creation. Users can create rules via a Web page, in which multiple triggers and actions composing an "if-then" rule can be selected. End users can also define simple and automatic execution flows. Software developers may extend the application by adding new web services, triggers, and actions via wiring diagrams.
Zipato Home Management http://zipato.com/	A web-based application for a complete home control and automation. It requires its own gateway, called Zipabox, to which many different sensors and actuators can be connected, but they must be compliant with Zipato communication protocol. Users may create "if-then" rules through Zipazle, a visual environment that provides typical programming language constructs (e.g., if, while, logical operators).

P6, it satisfies most of all the characteristics (see Table 4); this means that developers could potentially extend it and possibly make it able to manage multiple targets (principle P5), the remote control and the detection of the user's location (principle P6).

Tasker, enhanced with a specific plug-in (AutoVera), is able to satisfy five out of six design principles. The plug-in is specifically oriented to the management of a smart home and allows Tasker i) to understand abnormal, but intentional, changes to the defined rules (P3); ii) to

Table 4
Classification of the 12 tools resulting from the review, according to their characteristics.

Tool	Characteristics							
	C1. License and price	*C2. Flexibility in device management*	*C3. Easy device behavior specification*	*C4. Tool extensibility*	*C5. Technical support by manufacturer*	*C6. Integration with different smart devices*	*C7. Interoperability*	*C8. Flexibility of rule for behavior configuration*
Atooma	Freeware	✓	✓	✓	✓	~	✓	~
Bipio	Open source	✓	✓	✓	✓	×	✓	✓
GALLAG Strip	Commercial	✓	×	✓	✓	✓	×	✓
IFTTT	Freeware	×	✓	✓	×	~	✓	×
itDuzzit	Commercial & free version	✓	✓	✓	✓	~	✓	✓
Locale	Commercial	×	✓	✓	✓	~	~	×
Ninjablocks	Commercial	✓	✓	✓	✓	✓	✓	✓
Tasker	Commercial & trial version	~	✓	✓	✓	~	✓	✓
Twine	Commercial	✓	✓	×	✓	✓	~	✓
WigWag	Commercial	✓	✓	✓	✓	✓	✓	✓
We Wired Web	Commercial & free version	×	✓	✓	✓	×	✓	✓
Zipato	Commercial	✓	✓	✓	✓	✓	✓	✓

Table 5
Classification of the selected 6 tools, according to the design principles for smart home control.

Tool	Design principles					
	P.1 - Allow for the organic evolution of routines and plans	*P.2 - Easily construct new behaviors and modify existing behaviors*	*P.3 - Understand periodic changes, exceptions and improvisation*	*P.4 - Design for breakdowns*	*P.5 - Account for multiple, overlapping and occasion-nally conflicting goals*	*P.6 - The home is more than a location*
Atooma	✓	✓	×	×	×	~
Bipio	✓	✓	×	×	×	×
IFTTT	✓	✓	×	×	×	~
itDuzzit	✓	✓	×	×	×	×
Tasker	✓	✓	✓	×	✓	✓
We Wired Wed	✓	×	×	×	×	×

Legend of both tables: ✓ totally satisfied, ~ partially satisfied, × not satisfied.

manage multiple objectives (P5); and iii) to perform remote control, position detection and monitoring of the home adequately, even in case of family members' absence (P6). Like the other tools, Tasker is not able to recognize error situations and recover from them (P4).

On the basis of this analysis, Tasker resulted the tool that most satisfies the design principles. Its extensibility opens up a variety of possibilities for customizing it to the smart home case. Atooma and IFTTT are more widespread than Tasker and are under continuous development. With the aim of deriving design implications for creating software tools that support end users to easily manage rules for operating the smart home, an experimental study involving real users was performed to compare Atooma, IFTTT and Tasker. The choice of these three tools depends on the fact that from our literature review they resulted the most suitable tools for end users. In addition, IFTTT was used as baseline being the most widespread tool, as reported in (Desolda et al., 2017). The experimental study and its results are illustrated in Section 4.

3.4. Threats to validity

The main threat to the literature review regards the completeness of the selected publications that arises from the search and the selection process. The search string for the automatic search used terms related to the main concepts "tools", "smart home" and "internet of things" and also other terms that are synonyms, such as "device", "instrument", "smart house", "digital home", etc. Thus, we alleviated this threat by using a search string that combines main and related terms. Furthermore, we considered publication venues related to EUD, AmI and IoT, and most prestigious journals and conferences on HCI based on journal ranking and acceptance rate respectively, since we were interested in identifying tools suitable to end users.

The performed tools' classification could be threatened by researcher biases and misunderstandings. To mitigate these threats, two reviewers independently carried out the classification process and the disagreements on the results were solved by discussion in a face-to-face meeting (see Section 3.2).

4. Experimental study

A study in which a sample of people were required to use the three tools, i.e. Atooma, IFTTT and Tasker, was performed in order to investigate how usable is operating on rules for controlling devices of a smart home and which features hamper the interaction. The study is reported in this section, after a brief description of the three tools.

4.1. The tools

More details on the three tools used in the comparative study are provided in this section.

4.1.1. Atooma

Atooma has been developed by an Italian team; it aims at supporting people to automatize several tasks. Users define rules in which they indicate the action to perform if a specific condition is met. The paradigm used is based on the classic ECA rule. Rule creation is supported by a visual interface, where the user may set up the "if" part with one or more conditions (events that may occur) and the "then" part ("DO" in Atooma) with one or more actions.

The user can create a rule by tapping on the button " + " in the main window of Atooma. Then, the interface related to the IF part of the rule appears (as shown in Fig. 1(a)). At the bottom of this window, five circles representing different categories of elements to be used in the rule are available. Specifically, going from left to right, each circle allows the user:

- to access the mobile sensors, like WiFi, GPS, SMS etc. (pink).
- to access external web applications like Dropbox, Gmail, Facebook etc. (purple).
- to interact with objects, for example, Near Field Connect (NFC) tags, smart watches and IoT devices (blue).
- to access third-party plugins (green).
- to enable triggers about files, directories, images and media etc. (orange).

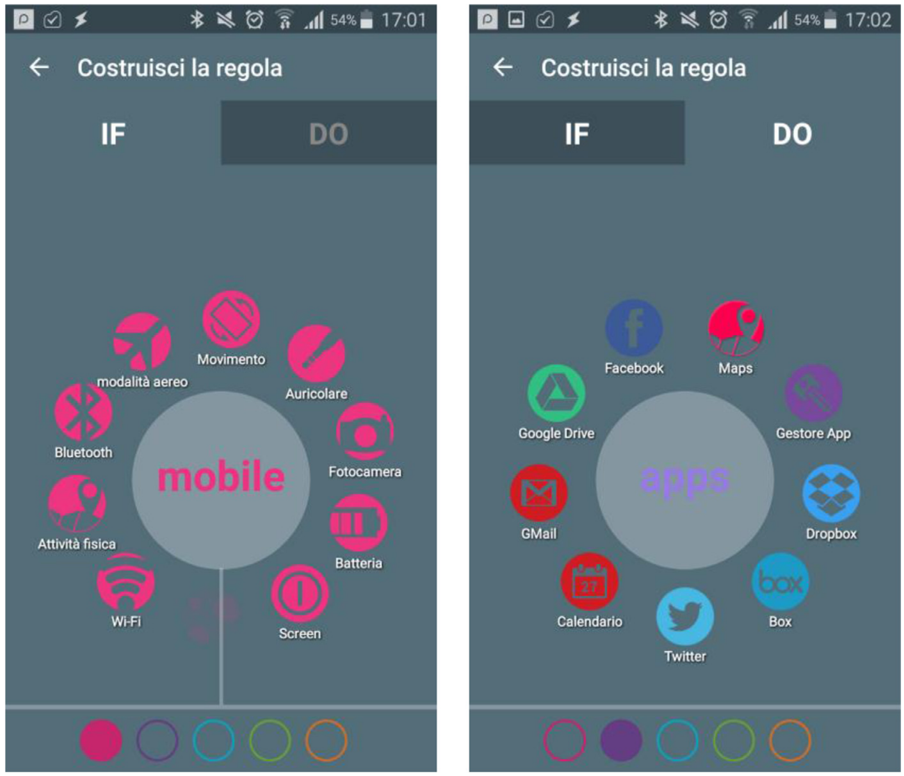
For example, in Fig. 1(a), the user selected the pink circle, since he/she is interested in accessing the smartphone sensors. Thus, the big wheel "mobile" showing all options appeared. The user has to tap the wheel and turn it to browse all the available options. Once he/she taps on the desired option (for example, Wifi), Atooma shows all the related conditions (e.g. for WiFi sensor they are: on, off, connected, and disconnected). By setting possible parameters, the condition is completed. At this point, the user can add another condition to the IF part in a similar way or can go to the DO part of the rule. The procedure to define one or more actions corresponding to the DO part of the rule is the same of the IF part (see for example Fig. 1(b)).

Atooma also provides a "Gallery" of the most selected and recent rules created by the Atooma community that users can choose.

4.1.2. IFTTT

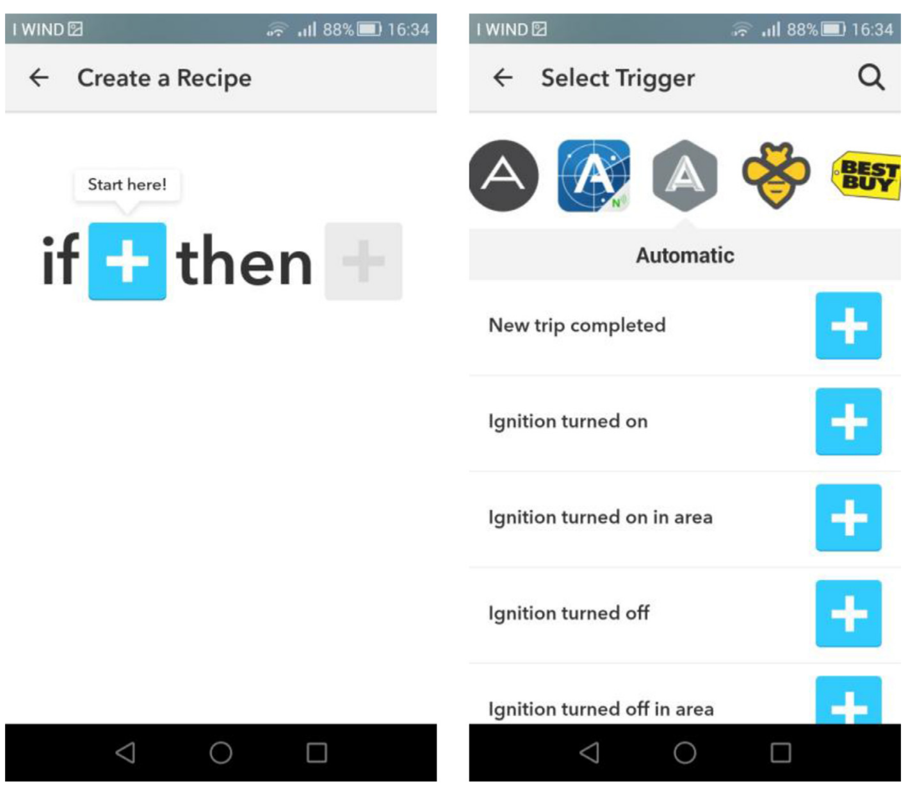
IFTTT is the acronym for "IF This Then That". It is reported to be one of the most used tools to connect and interact with thousands of IoT devices and web services (e.g., see (Desolda et al., 2017)). Also in this case, the paradigm used in this tool is the ECA rule. However, rules (called 'recipes' in IFTTT) may contain only one event (or condition) and only one action, and this has been discussed in literature as an important limitation because only simple rules can be created, not able to fulfil users' needs in many contexts (Cabitza et al., 2017; Ur et al., 2014).

The user can start to create the IF condition of a new rule by tapping on the " + " button close to the IF icon (as shown in Fig. 2(a)). Thus, a



(a) (b)

Fig. 1. The Atooma interface: (a) the IF part and (b) the DO part of the rule.



(a) (b)

Fig. 2. The IFTTT interface for creating a rule (a) and selecting the trigger from the "Automatic" channel (b).

window showing the list of available channels appears. The channels are all the smart objects and applications (for example, Android devices, Gmail app, Dropbox, Nest Cam, etc.) that can be used as triggers in a rule. The user has to select one of the available channels and a new window appears. In Fig. 2(b)[5], for example, the user has selected the Automatic channel, thus the window shows on the top the channels close to the Automatic one and below all its triggers. Once the user taps on the desired trigger, s/he is able to set it for completing the "IF" part of the rule (no other IF condition is allowed anymore). To define the action of the "THEN" part of the rule, the user must follow the same steps as for the IF part. In this case, IFTTT shows only channels that can be used in the action part of the rule as actuators (for example, Philips Hue, Nest Thermostat, etc.). At the end, the user can save the rule. As it happens in Atooma, the IFTTT community shares rules through the IFTTT platform and Facebook.

4.1.3. Tasker

Tasker is a smartphone application allowing its users to define custom actions when a trigger is launched. It was firstly developed for PalmOS in 2007 and then it has been released in 2010 on the Google Play Store. The interface provides a bottom bar with three icons:

- a Magnification lens that allows users to search in the app.
- a "+" symbol that allows users to create new rules.
- a "?" symbol that opens the guide.

On the top bar, there are three different tabs: Profiles, Activities and Scenes (see Fig. 3(a) where they are in Italian).

By clicking on the Profiles tab ("Profili" in Italian, as shown in Fig. 3(a)), the user can specify the "if" part of the rule, thus a new profile containing at least a condition has to be created. A profile can be added by clicking on the "+" symbol in the bottom bar and a pop-up window containing the list of condition categories appears (e.g., Application, Event, Day, Hour, Position and Status, in the bottom-right box in Fig. 3(a) where they are in Italian). The user has to choose one of such categories and its related conditions. Differently from Atooma and IFTTT, in Tasker the actions that can be included in the "then" part of the rule have to be previously created in the Task tab (in Fig. 3(b) "Attività" in Italian). To create the needed actions the user must tap on the "+" button in the Task tab and the window showing a list of available categories of actions appears (see Fig. 3(b)). After selecting the category, the available actions are displayed, and the user can choose and set them accordingly.

Tasker also allows users to create and manage sequences of actions through the Scenes tab ("Scene" in Italian).

4.2. Participants

A total of 20 people agreed to participate in the study. They aged between 14 and 60 years old (mean = 34.3, SD = 14.30, 11 male and 9 female); all of them were familiar with mobile applications and had not used any of the three tools before. Ten participants are digital natives (i.e. aged 14–35 years old), the remaining 10 participants are digital immigrants (i.e. aged > 35 years old). Even if all of them were familiar with mobile applications, digital natives were more prone to the use of mobile phones and their applications. Participants were selected of different age, habits and skills, since any home inhabitant should be supported in effectively creating rules for managing their smart home.

[5] At the time of paper submission the interface of IFTTT is slightly changed with respect to the version we evaluated with users (e.g., now the term "Recipe" has been changed in "Applet").

4.3. Design

The main goal of the study was to investigate on the usability of Atooma, IFTTT and Tasker when the user operates on rules for controlling different devices of a smart home. Specifically, we were interested in investigating interaction mechanisms that hamper users in creating rules as well as those that they appreciate most. Thus, two research questions guided the study:

RQ1) How different are the three tools in terms of *user performance* in composing and managing rules?
RQ2) How different are the three tools in terms of *user satisfaction* in composing and managing rules?

To reach this goal a within-subjects study was performed with tool as independent variable and three within-subject factors: Atooma, IFTTT, and Tasker. In other words, each participant used all the three tools in sequence but in a different order by considering permutations of the three tools according to the Latin-Square design, in order to minimize learning effects (Graziano and Raulin, 2012).

4.4. Tasks

With each tool, the participants were required to perform a total of 5 tasks in the following order:

Task 1: Create a rule that deactivates the WiFi at home if the smartphone is not connected to any online services.
Task 2: Create a rule that sends an SMS to warn a family member when you are close to the home.
Task 3: Modify the rule defined for accomplishing Task 2 to tell a specific person that you are already at home.
Task 4: Delete the rule created for Task 1.
Task 5: Deactivate all the created rules without deleting them.

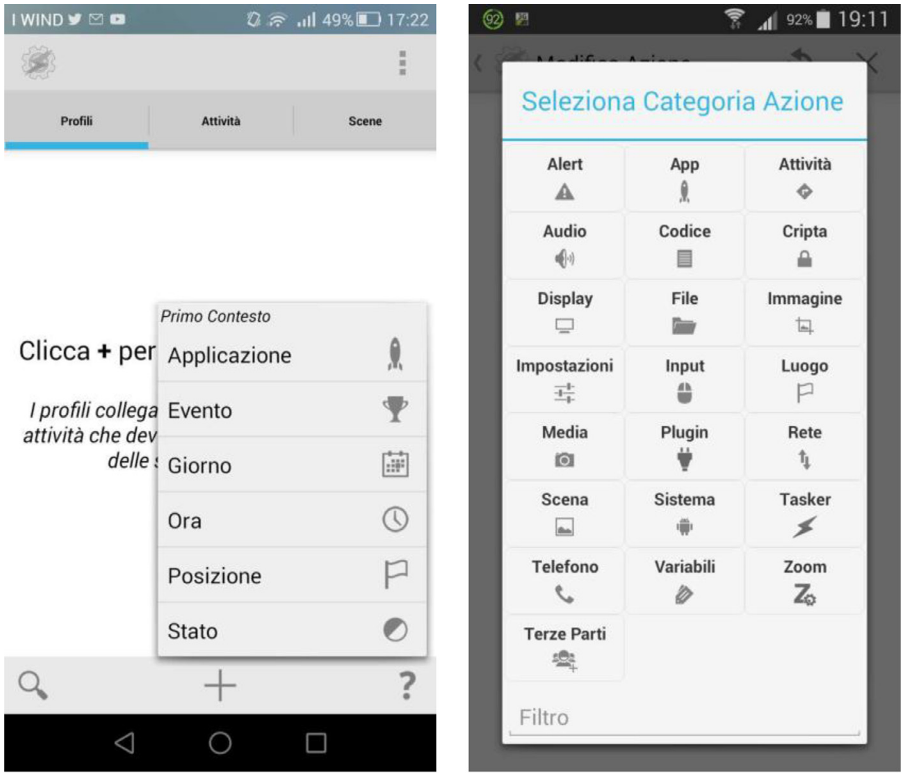
In accordance with the within-subjects design, each participant performed 15 tasks, for a total of 300 trials (20 users × 15 tasks). To improve the external validity of the study, the tasks refer to all the available operations on the rules, i.e., creation, editing, deletion and deactivation of a rule.

4.5. Procedure

A first researcher conducted the study and a second one observed each participant during the interaction with the tool used to carry out the experimental tasks. Fig. 4 shows the observer who is noticing significant behaviors and possible comments of the participant during his interaction with the mobile device. Each study session was held in a quiet room familiar to the participant.

The comparative study lasted 9 days: for 7 days 2 participants were individually observed per day, while for 2 days 3 participants were individually observed. Each study session followed the same procedure. First, a 10-minute presentation was given by the first researcher to introduce participants to the addressed domain and to illustrate the goal of the study. Participants were asked to sign a consent form for audio recordings and photo shoots. After, the researcher introduced the tool to be used by demonstrating two examples of rule creation. Then, the participant performed two training tasks, possibly asking the researchers for help. Both the examples and the training tasks were the same for each tool. After the training, the participant had to execute the five experimental tasks. The participant read aloud the text describing the task and then started task execution. At the end of the five tasks, the participant filled out two questionnaires about the tool used. Before repeating the same procedure with the next tool, the participant was invited to relax for 5 minutes.

A pilot study involving two people was first performed to check and refine the overall procedure.



(a) (b)

Fig. 3. An example of the Tasker Android user interface: (a) the profiles tab and (b) the task tab.

Fig. 4. The observer and the participant who is interacting with the mobile device.

4.6. Data collection and analysis

Both qualitative and quantitative data were collected to analyze user performance and satisfaction on the three tools: i) the set of notes taken by the observer during the task execution, ii) the video recorded during the different study sessions, iii) the answers participants gave to the two questionnaires; iv) the free comments participants provided at the end of the study. Regarding user performances, metrics such as success rate and execution time were considered. Two researchers individually transcribed the notes and integrated them with video analysis and participants' comments. As a result, each researcher built an excel file reporting the following data: user ID (from 1 to 20), user age (digital natives – digital immigrants), system name, task ID (T1–T5), time (in seconds), task success (right - wrong). The coded data were double-checked for reliability, leading to an initial value of 88%. Discrepancies were solved by discussion between the two researchers.

To analyse the performance in accomplishing the tasks, regarding success rate variable the Friedman test was used to test for differences between groups (p-values below 0.05 indicate a significant difference), and in addition, to examine where the differences actually occurred, a post hoc analysis with Wilcoxon signed-rank test was conducted by applying a Bonferroni correction. Concerning the time variable, the one-way repeated measures ANOVAs (all Greenhouse–Geisser corrected) with post hoc pairwise comparisons (Bonferroni corrected) were adopted.

In order to investigate the performance of digital natives vs. digital immigrants for each tool, the chi-square tests were applied (p-values below 0.05 indicate a significant difference). Furthermore, for each tool *t*-tests were applied on the two dichotomous variables with respect to the time spent to create rules (also in this case, p-values below 0.05 indicate a significant difference).

Regarding user satisfaction, two questionnaires were administered during the study: the Systematic Usability Scale (SUS) (Brooke et al., 1996) and Net Promoter Score (NPS) (Nardi and Engeström, 1999). Appendix B reports a description of the two questionnaires. To analyse users' satisfaction, the one-way repeated measures ANOVAs (all Greenhouse–Geisser corrected) with post hoc pairwise comparisons

Table 6
Performance in accomplishing the missions.

		Atooma	IFTTT	Tasker
Success rate	*Right*	76	90	78
		76%	90%	78%
	Wrong	24	10	22
		24%	10%	22%
Time	*Seconds*	192	180	216
		(175)	(184)	(200)

(Bonferroni corrected) were again used (p-values below 0.05 indicate a significant difference).

Data collected were also analysed to identify usability problems of each tool that hampered users in accomplishing the experimented tasks.

4.7. Results

The following sub-sections report the results of the analysis of the data gathered during the study. Specifically, Section 4.7.1 presents the results of the analysis related to the quantitative data, while Section 4.7.2 reports the analysis of qualitative data, i.e. some usability problems that end users experienced during the test.

4.7.1. Quantitative results

The data were analyzed in order to verify if there is any difference in terms of user performance when creating rules with the three different tools. In particular, success rate in rule creation was operationalized as a nominal variable with two values: (1) right, that is, the rule created solved the task; (2) wrong, that is, task concluded without correctly identifying the rule. In total, 100 tasks were performed by the 20 participants in each of the three tool conditions. Table 6 reports the frequency and percentage values for each value of task success, alongside mean and standard deviation (in bracket) of time in the three tool conditions. Time is the number of seconds the participants spent to execute the tasks; specifically, the starting point was considered the moment when the participant read aloud the first task, while the ending point was the moment when the participant saved the rule.

Most tasks were successfully accomplished by creating the correct rule. IFTTT was the tool through which the participants created the rightest rules (90%), followed by Atooma and Tasker with a very similar numbers of correct and wrong tasks (i.e. 76% and 78%, respectively). The Friedman test was used to test for differences between groups. There was a statistically significant difference in success rate depending on which type of tool was used to create the rules, $\chi^2(2) = 8.390$, $p = 0.015$. To examine where the differences actually occur, a Wilcoxon signed-rank test was carried out. Post hoc analysis with Wilcoxon signed-rank tests was conducted with a Bonferroni correction applied, resulting in a significance level set at $p < 0.01$. There were no significant differences between Tasker and Atooma ($Z = -0.333$, $p = 0.739$). However, there was a statistically significant difference in IFTTT vs Atooma ($Z = -2.746$, $p = 0.006$) and Tasker vs IFTTT ($Z = -2.683$, $p = 0.007$). Triangulating the Wilcoxon signed-rank test results with the data reported in Table 6, we can assert that IFTTT was the most accurate tool. A deeper analysis on the success rate for each task accomplished by participants revealed that Task 1 ("Create a rule that deactivates WiFi if the smartphone is not connected to any network") was the most difficult task to be carried out with Tasker. In fact, the success rate for Task 1 was 40%. Instead, participants had difficulty in performing Task 5 ("Deactivate all the created rules without deleting them") with Atooma (see Fig. 5).

Tasker appeared the most time-consuming tool to accomplish the five tasks (M = 216 s, SD = 200), followed by Atooma (M = 192, SD = 175) and IFTTT, which is the faster tool (M = 180, SD = 184). However, no statistically significant difference emerged in terms of time as demonstrated by the repeated measures ANOVA with a Greenhouse-Geisser correction ($F(1.780, 176.205) = 1.462$, $p = 0.235$).

In order to investigate more in depth if the three tools well support people in creating rules, we compared the performance of digital natives vs. digital immigrants for each tool. As shown in Table 7, for each tool chi-square tests were applied on the two dichotomous variables. No significant difference emerged in the three experimental conditions regarding the influence of participant background (digital natives vs. digital immigrants) on the success rate of the created rules ($\chi^2_{Atooma}(1) = 0.219$, $p = 0.639$; $\chi^2_{IFTTT}(1) = 0$, $p = 1$; $\chi^2_{Tasker}(1) = 0$, $p = 1$). In addition, the number of right rules and wrong rules created by digital natives and digital immigrants through IFTTT and Tasker was the same, i.e. 45 right rules and only 5 wrong rules using IFTTT and 39 right rules and 11 wrong rules using Tasker. In the case of Atooma, the number of right rules and wrong rules is similar but, also in this case, no statistically significant difference emerged.

As shown in Table 8, for each tool *t*-tests were applied on the two dichotomous variables with respect to the time spent to create rules; the column "Statistic test" reports *t*-test results (a significant difference exists when p-value < 0.05). Table 8 shows that there was a significant difference regarding the influence of participant background (digital natives vs. digital immigrants) on the time spent to create rules only in the Atooma condition ($t(49) = -3.934$, $p < 0.000$). No significant difference on time emerged in IFTTT and Tasker condition ($t(49) = -0.163$, $p = 0.871$ in IFTTT condition, $t(49) = -1.779$, $p = 0.081$ in Tasker condition).

Fig. 5. Success rate of each task in the three experimented tools.

Table 7
Influence of participant background (digital natives vs. digital immigrants) on the success rate of the created rules.

Tool	Digital natives		Digital immigrants		Statistic test
	Right rules (N)	Wrong rules (N)	Right rules (N)	Wrong rules (N)	
Atooma	37	13	39	11	$\chi^2(1) = 0.219$, $p = 0.639$
IFTTT	45	5	45	5	$\chi^2(1) = 0$, $p = 1$
Tasker	39	11	39	11	$\chi^2(1) = 0$, $p = 1$

Table 8
Influence of participant background (digital natives vs. digital immigrants) on the time spent to create rules. (Tests labeled with ∗ are Statistically Significant).

Tool	Variable	Digital natives		Digital immigrants		Statistic test
		M	SD	M	SD	
Atooma	Time	145	137	240	195	$t(49) = -3.934$, $p < 0.000$ *
IFTTT	Time	179	212	182	151	$t(49) = -0.163$, $p = 0.871$
Tasker	Time	196	190	235	209	$t(49) = -1.779$, $p = 0.081$

Table 9
Mean and standard deviation (SD) of the SUS scores related to the three tools.

Tool	Mean	SD
Atooma	62.75	21.93
IFTTT	52.62	21.86
Tasker	34.62	23.62

User satisfaction with the tool was investigated through two questionnaires: 1) the SUS to estimate tool usability perceived by the user, and 2) the NPS questionnaire to assess the willingness of the user to recommend the tool to others. The description of how to compute the scores of these questionnaires is provided in Appendix B.

Based on the answers to the SUS questionnaire, Table 9 shows descriptive statistics (mean and standard deviation) of the SUS score for each tool.

A repeated measures ANOVA with a Greenhouse-Geisser correction determined that the user satisfaction with the three tools differed statistically significantly, $F(4890.619, 260.559) = 18.770$, $p < 0.0005$. Post hoc tests using the Bonferroni correction revealed that there was a significant difference in satisfaction between Atooma and Tasker ($p < 0.0005$), and between IFTTT and Tasker ($p = 0.008$), and also between Atooma and IFTTT ($p = 0.025$). Combining these results with the descriptive statistics reported in Table 9, it emerged that Atooma was perceived as the most satisfying tool (M = 62.75, SD = 21.93) according to the SUS scale, followed by IFTTT (M = 52.62, SD = 21.86) and Tasker (M = 34.62, SD = 23.62).

Moreover, we compared, for each tool, the satisfaction with the tool of digital natives vs. digital immigrants. As shown in Table 10, the only significant difference regards the SUS Score of Tasker, which is in favor of digital natives.

In the context of software applications, the NPS score is used to measure the willingness of users to recommend an application to others. For all the three tools, the NPS score was below the acceptable threshold equal to 0. In particular, the NPS score of Atooma was −25, of IFTTT was −35 and, finally of Tasker was −60. In order to analyze in depth the NPS score, descriptive statistics (mean and standard deviation) for each tool of the NPS score given by each participant was considered (see Table 11).

Table 10
User preferences results: digital natives vs. digital immigrants (Tests labelled with ∗ are Statistically Significant).

Tool	Variable	Digital natives		Digital immigrants		Statistic test
		M	SD	M	SD	
Atooma	SUS Score	71.75	13.02	53.75	25.80	$t(18) = 1.003$, $p = .324$
	NPS Score	6.90	1.52	7.20	1.62	U-value = 42
IFTTT	SUS Score	63.75	17.53	41.50	20.65	$t(18) = 1.62$, $p = 0.116$
	NPS Score	6.00	2.62	6.40	2.67	U-value = 42
Tasker	SUS Score	47.00	18.85	22.25	22.44	$t(18) = 2.83$, $p < 0.05$ *
	NPS Score	4.60	2.21	4.90	4.00	U-value = 40.5

Table 11
Mean and standard deviation of the NPS scores given by participants related to the three tools.

Tool	Mean	SD
Atooma	7.00	1.54
IFTTT	6.20	2.59
Tasker	4.75	3.16

A repeated measures ANOVA with a Greenhouse-Geisser correction determined that the user satisfaction with the three tools differed statistically significantly, $F(28.347, 2.994) = 9.468$, $p = 0.001$. Post hoc tests using the Bonferroni correction revealed that there was a significant difference in satisfaction between Atooma and Tasker ($p = 0.003$) and between IFTTT and Tasker ($p = 0.03$), while no difference emerged between Atooma and IFTTT ($p = 0.31$). This result suggests that users could recommend both Atooma (M = 7.0, SD = 1.54) and IFTTT (M = 6.2, SD = 2.59) to others, but they appeared to be not confident with Tasker (M = 4.75, SD = 3.16). We compared, for each tool, the satisfaction with the tool of digital natives vs. digital immigrants. A Mann-Whitney test was used to analyze the significance of the ordinal assessment for each NPS value. As shown in Table 10, no statistically significant difference was found in the three tools between digital natives and digital immigrants.

4.7.2. Qualitative results

Qualitative results are derived from the analysis of the notes took by observers during the study and from the free comments provided by participants after the study. They allowed us to identify the usability problems participants experienced in accomplishing the 5 tasks.

When using Atooma, one of the major problems emerged during the execution of Task 2, which required to create a rule that sends an SMS to warn a person that you are reaching her when you are close to the meeting point. To complete it, users needed to deeply know the Atooma interface. As a matter of fact, the icons disposed as a wheel needed to be turned to show more options, resulting a new way of interaction for end users. For this reason, most of the participants did not perform the rotation, and often they were stuck and/or completed the task over the time limit (or making some mistakes).

Another typical problem emerged during the task execution was the selection of a meeting point. Participants should have firstly created a point of interest (POI) and then used it in the rule. Most of them believed that, once created, the point of interest was automatically selected in the rule, but it was not so. Some minor mistakes were also performed during the execution of Task 3 (i.e. modification of the rule defined for accomplishing Task 2 to warn a loved person that you are at home). The tool does not allow users to edit the POI, so the users should

Fig. 6. The position of the "+" button circled in red.

Fig. 7. The Tasker "+" button.

create a new one. This was not clear to many participants that tried multiple times to edit the position of the POI. Finally, the last problem users encountered was related to the deletion of a rule (Task 4). To perform this action, users should swipe with the finger the rule to the right, but 60% of the participants did not performed it well.

In using IFTTT, during the tasks' execution, in most of the study sessions, the position of the "+" button was very unclear (see Fig. 6). Due to this problem, participants found this button after several seconds of search. During the execution of Task 1, due to the high number of the icons (almost 170) associated to the actions, participants had to spend a lot of time (almost 7 minutes) to find the right one to complete it. After few tries, however, all the participants managed the triggers and actions in the list view. A minor aspect related to the encountered problems involved the IFTTT map. Compared to the screen, the map size was very little. Finally, a search box was at the top right of the screen but, due to its transparency, it was invisible to each user.

The main problem with Tasker was concerned with the terms used in the tool. The main window allows users to create "Profiles", the second one to create "Tasks" and the third one "Scenes". These concepts are well described at the launch of the application, however only 5 out of 20 participants correctly created the rule from the beginning. The correct order to create a rule is to start from the "Profiles" tab. However, most of participants started from the "Tasks" tab. When asked for the reason, all participants answered that the "Tasks" term suited most on what they wanted to do. Cleared the terms, the "+" button created many troubles (see Fig. 7). In fact, when clicked, the "+" button allows users to select a "context". However, when this button is clicked, a panel with different choices related to an application launch, a day, an hour, etc. pops up. Users felt confused on how to use this panel. To create the rule that activates and deactivates the Wi-Fi, participants had to explore the tool and follow a complex path through it to get the rule done. The same difficulties were found in the rule for sending the SMS.

4.8. Discussion

The comparative study was driven by two guiding questions: RQ1) How different are the three tools in terms of user performance in composing and managing rules? RQ2) How different are the three tools in terms of user satisfaction in composing and managing rules?

As regards to the first question, the comparative study demonstrated that IFTTT was the most accurate tool in terms of correct rules created to accomplish the tasks: the 90% of the created rules were correct. Through the other two tools, i.e. Atooma and Tasker, participants were also able to create rules satisfying the tasks, but in a minor percentage, 76% and 78% respectively. In addition, IFTTT was the tool that allowed the participants to accomplish all the 5 tasks without any particular difficulty, while participants had difficulty in creating a rule to deactivate WiFi with Tasker and to deactivate the rules created with Atooma. To analyze the performance, the variable execution time was also considered. For this analysis, no significant difference emerged among the three tools. However, it is worth noting that the time spent to create rules with IFTTT was the minor while Tasker appeared the most time-consuming tool. A deeper analysis on the technological background of the participants highlighted that it did not have any influence on the creation of right rules with the three tools. A small difference emerged in terms of time spent for creating the rules with Atooma. In fact, it was found that digital natives spent just over half the time of digital immigrants in creating the rules. This showed that Atooma might not be adequate for people who have no experience with such a type of applications.

As said above, IFTTT appears to be the most effective tool in comparison with Atooma and Tasker. This result could be explained by

IFTTT simple interface, which shows all triggers and actions as a list. Even if the participants initially spent more time in finding a specific item, afterwards they perceived easy to interact with a list view. In addition, though on one side the items' categorization in Atooma seemed to facilitate the selection of elements, on the other side the circle view of the triggers and actions hindered participants, mainly digital immigrants, who are more used to interact with list views. Atooma prevented the participants to perform tasks quickly also because it did not consider the data previously stored by the participant during a rule creation, forcing him/her to create a new one each time, as happened during the study when participants had to select a POI.

In the case of Tasker, the main hampering factor was related to the terminology it uses, which is not familiar at all to the participants; this problem affects the whole usability of the tool.

Concerning the satisfaction in interacting with the tool, the analysis of the SUS questionnaire revealed that Atooma was the most satisfying, followed by IFTTT and Tasker, respectively. No significant difference emerged between digital natives and immigrants regarding the perceived satisfaction with the used tools, except for Tasker, which was more appreciated by digital natives. Coherently with the SUS results, Atooma had the highest score of NPS among the three evaluated tools. However, statistical analysis on the NPS score found that participants would prefer both Atooma and IFTTT with respect to Tasker. This result is justified by the participants' appreciation of Atooma user interface that participants found more appealing.

As already said, our main objective in performing the experimental study was not the comparison of the three tools per se; rather we were interested in identifying the interaction mechanisms that better satisfy users in order to provide useful indications for the design of future tools. The derived design implications are presented in Section 6.

5. Threats to validity

This section analyses issues that are potential threats to the validity of the comparative study (Graziano and Raulin, 2012) and presents how they were minimized.

5.1. Internal validity

The internal validity refers to the extent to which the experimenter can be confident that the observed changes in the dependent variable were due to the effects of the independent variable and not to the effects of extraneous variables. The main threats to the internal validity of our study were: 1) the sequence effect, 2) the subject effect, 3) the experimenter effect, 4) the diffusion of the treatment, and 5) understandability of the documents.

The *sequence effect* is a typical threat of the within-subjects design study, where each participant is exposed to more than one experimental condition, as in this case, where all the 20 participants used the three tools. The sequence effect was controlled by varying the order of presentation of the tools to be used according to a Latin Square design. However, we are planning to perform a between-subject design study involving more participants so that we can use other statistics and provide further evidence to the study results.

The *subject effect* refers to any changes in the behavior of participants that were due to being in the study, rather than to the variables under study. When people know they are being observed, they may behave differently than they normally would. Participants are often sensitive to cues from the experimenter. In our case, the observer was particularly careful in retaining his emotions, both positive and negative, in order to mask any type of signal.

The *experimenter effect* is concerned with any biasing effects in a study that are due to actions of the researcher. The researcher attempts to carry out the study procedure as objectively and as accurately as possible. We eliminated this threat by involving the same person that acted as observer. In this way, we avoided any variability in the study procedure from the initial training to the way in which users were observed.

The *diffusion of the treatment* is related to the possibility that participants, who are in close proximity, are able to communicate with each other. To compensate this problem, since the researchers knew all the participants, those ones that were familiar were observed in consecutive test sessions. Moreover, the participants were asked to return all the material at the end of each session.

Finally, the *understandability of the material* was alleviated by performing a pilot study, whose results were instrumental for clearing up all the misunderstandings.

5.2. External validity

The external validity refers to the degree to which researchers are able to generalize the results of a study to other participants, conditions, times, and places. The main threats to the external validity of our study were: 1) the participants' selection, 2) the tools' selection and, finally 3) the task complexity.

The *participants' selection* was carefully carried out in order to involve in the study a representative sample of the target users. We involved people that were part of a family with different roles, i.e. children, parents and grandparents. In addition, we intentionally recruited people without experience with IoT technology as well as task-automation applications, since we wanted to analyze the usability of such tools from the point of view of inexperienced people. We are aware that this a limitation to the generalizability of the results, since end users of smart environments could be expert in the configuration of home-automation systems. Thus, we can safely accept the study results for not-skilled people, but further studies have to be performed including expert people.

The *tools' selection* was based on the results of the literature review, whose aim was to identify the most flexible and powerful end-user oriented tools for IoT and smart home control. Thus, once we identified those tools that appeared to be the most usable, we performed the study involving real users to test whether such tools can be actually used by people not skilled in computer programming, like EUD prescribes.

The *task complexity* was managed by considering a set of tasks defined in a previous study (Cabitza et al., 2017), whose aim was to identify tasks that household members could be interested in performing to adapt the behavior of a smart home to their needs. It is worth noticing that the definition of the rules to accomplish the defined tasks involved the whole available functionality provided by each tool, i.e., creation, editing, deletion, activation and deactivation of a rule. In addition, we identified tasks that could be accomplished by all the three tools.

5.3. Construct validity

The construct validity refers to how well the study results support the theory behind the research study. The construct validity of a study might have been influenced by the measures that were applied in the quantitative analysis and the reliability of the questionnaire. We alleviated the first threat by considering metrics, such as the success rate and the time spent to carry out a task, which are commonly used in user studies (Preece et al., 2015). Regarding the questionnaires, we used the SUS and the NPS questionnaires, which are well-known and thoroughly validated models for evaluating usability of software applications.

5.4. Statistical validity

Statistical validity refers to the question of whether these statistical conclusions are reasonable, i.e., the accuracy of the *p*-value on which a statistical decision is based. In our study this was addressed by applying the most common tests employed in the empirical software engineering field (Graziano and Raulin, 2012) and by considering a p-value below 0.05.

6. Design implications

Inspired by successful design solutions implemented in the tools considered in our study and by taking into account the problems participants encountered during the interaction with the three tools, we elaborated a list of ten implications for the design of future tools supporting the creation of ECA rules to control a smart home. The first eight are specifically oriented to single-user interaction with a tool for smart home control and management, since our comparative study mainly investigated this kind of interaction. In addition, from both the observation of user behavior during the study and the design principles discussed in (Davidoff et al., 2006), we also recognized the importance of supporting multi-user interaction with a smart home, a situation that obviously occurs in a house were different inhabitants live. Thus, we completed the list with two further design implications that consider this issue.

The ten implications are listed in the following:

1. *Facilitate trigger and action retrieval*
   Triggers and actions should be well categorized in order to facilitate retrieval and inclusion in the rules being composed (as done in Atooma), in order to overcome, for example, the difficulties some participants experienced with Tasker and IFTTT in identifying and including them in composing rules.
2. *Facilitate trigger and action selection*
   A visualization of triggers and actions based on the list view appears as an intuitive and fast solution for item selection (like in IFTTT). The study revealed that selection of triggers and actions in Atooma requires a cumbersome interaction, as well as Tasker makes this activity too much articulated.
3. *Make rule creation flexible*
   The creation of trigger and action parts of a rule should be allowed in any order. This is not realized in most of existing tools: for instance, Atooma and IFTTT force the user to define first the trigger part, whilst Tasker requires that the action part of a rule is previously created and then used in the rule. In addition, Tasker provides another interesting form of flexibility: it allows creating sequences of actions (scenes) to be activated through a single command without any trigger. Finally, the creation of rules with multiple events (and conditions) and multiple actions should be supported, as happens in Atooma and Tasker, which thus offer a higher flexibility than IFTTT.
4. *Support re-use*
   Re-use of already created rules is a fundamental feature to allow users to define new rules in an easier and faster way, and to possibly share them with other users. The online galleries made available by Atooma and IFTTT answer to this request.
5. *Speak the user's language*
   Even if this is a well-known guideline (e.g. see Nielsen heuristics (Nielsen, 1993)), literature highlighted that user interfaces often lacked terminology familiar to the users thus affecting the whole usability of the application (see for example, (Russo et al., 2017)). More than in other software applications, the adopted terminology must be simple and concrete and never recall programming language constructs, as it often occurs in Tasker. Adopting user-oriented terms and voice-based interaction is probably the future of interfaces for smart home control.
6. *Keep track of interaction history*
   To enable an efficient usage, the system must keep track of the last user operations, to avoid repeated and boring tasks (see for example the problem of POI registration in Atooma).
7. *Be coherent with device interaction metaphor*
   The interaction mechanism adopted by the tool should be the same used by the user's device. For example, the swipe direction to delete a rule should be the same of the swipe direction adopted by the operating system for deleting e-mails and call logs, and not different as it happens in Atooma.
8. *Provide different levels of complexity*
   Different users may have different attitudes towards this kind of tools. Some users may prefer to be supported by a simple and fixed mechanism for rule creation, while others would like to address the task in a more flexible manner. For example, in our comparative study it was observed that, notwithstanding the low usability of Tasker, digital natives were significantly more satisfied by its interface than digital immigrants. Consequently, in order to accommodate different users' skills and attitudes, the same tool should offer different solutions for both simple automation tasks and more complex ones, through different levels of complexity, and thus realizing the so-called "gentle slope of complexity" (Spahn et al., 2008).
9. *Support management of conflicting rules*
   When different users participate in the creation of rules managing a shared space, as a house, it may happen that such rules are in conflict. Conflicts could be detected and solved at rule definition time, by means of notifications that advise the user of an already existing rule that could be in conflict with that one under definition, and thus starting a negotiation that might depend on priorities and permissions assigned to users on specific devices. Other conflicts may be recognized at use time, since more than one rules could simultaneously match in given conditions; also in this case, suitable policies should be defined to deal with such situations.
10. *Support collaborative rule creation*
    Collaborative tasks are usually carried out among home inhabitants; thus, supporting collaboration is an important aspect that is rarely addressed in current literature and commercial tools. Furthermore, some users might want to create complex rules, without possessing the necessary skills to do that, and other more tech-savvy inhabitants might be willing to help them. Mutual help mechanisms supporting the collaborative creation of rules should be integrated in future tools for smart home control. This aspect deserves further investigation, as well as it calls for adequate techniques that motivate people to participate. Recent papers of some of the authors of this article explore a gamified approach to address this problem (Benzi et al., 2015; Caivano et al., 2017; Fogli et al., 2016b).

7. Conclusions

This paper presented the results of a systematic and rigorous research aimed at deriving design implications that may help in developing future tools for smart home control and management. Thus, the results of a literature review in the field of smart home, as a specific application of AmI and IoT, are reported. The review was specifically oriented to identify the most suitable tools to manage and configure smart home behavior by end users, i.e. non-technical people that are neither expert in computer programming, nor willing to be.

The search and selection strategies adopted in the review allowed us to identify 48 publications out of 1057, which referred to a total of 12 different tools. A preliminary analysis highlighted that all tools implemented the rule-based paradigm that provides end users with visual interfaces allowing them to compose events and/or conditions with actions, using structures like ‘if-condition(s)-then-action(s)’ or ‘when-event(s)-then-action(s)’. Then, a high-level comparison of such tools analyzing their characteristics and the satisfaction of the design principles for the smart home, proposed in (Davidoff et al., 2006), discovered that the majority of tools are affected by important weaknesses that make them far from a successful adoption in real settings.

Three tools - IFTTT, Atooma and Tasker - emerged as the most promising for deepening our investigation aimed at identifying which aspects users appreciate most of these tools, and what are the problems that should be solved in the next years to make them acceptable by end users. These tools were tested in a study with end users to compare them in terms of user performance and user satisfaction in composing and managing rules. The empirical study has been conducted according to the meta-model reported in (Ferreira et al., 2018).

On the basis of the results of the literature review and the comparative study, ten design guidelines for creating software tools that allow end users to easily control and manage a smart home were derived. They provide operative indications for the development of tools implementing a rule-based approach to smart home control that will be used by users without programming experience.

As to future work, we are taking into account these design implications in the development of a new tool for smart home control and to evaluate it with real end users, possibly in family settings. A first prototype is already available, called We@Home: it is a solution integrating gamification elements to address the problem of collaboration and multi-user interaction with a smart home (Caivano et al., 2017). In addition, we will continue the exploration of alternative paradigms that could favour a natural interaction with smart homes and of interaction metaphors offering a better user experience such as those proposed in (Benzi et al., 2017; Fogli et al., 2017) also through replication of empirical studies (Baldassarre et al., 2014; Carver et al., 2014).

Acknowledgments

This work is partially supported by the Italian Ministry of University and Research (MIUR) under grants PON04a2_B "EDOC@WORK3.0″, and PON03PE_00136_1 "DSE" and “COSMOS”.

A special thank to Maria Francesca Costabile for the stimulating discussions.

Appendix A: Relevant papers resulting from the literature review

P1. Ballesteros, F.J., Soriano, E., Guardiola, G.: Octopus: An Upperware based system for building personal pervasive environments. *Journal of Systems and Software*, 85(7), 1637–1649 (2012). doi: 10.1016/j.jss.2012.02.011.

P2. Barricelli, B.R., Valtolina, S.: Designing for End-User Development in the Internet of Things. In: Díaz, P., Pipek, V., Ardito, C., Jensen, C., Aedo, I., Boden, A. (eds.) End-User Development, vol. 9083, pp. 9–24. Springer International Publishing (2015).

P3. Bergesio, L., Bernardos, A.M., Marquínez, Í., Besada, J.A., Casar, J.R.: PERSEO: A system to Personalize smart Environments through Smartphones, sEnsors and media Objects. *Journal of Ambient Intelligence and Smart Environments*, 7(6), 777–804 (2015). doi: 10.3233/AIS-150,345.

P4. Cabitza, F., Fogli, D., Lanzilotti, R., Piccinno, A.: Rule-based tools for the configuration of ambient intelligence systems: a comparative user study. *Multimedia Tools and Applications*, 1–21 (2016). doi: 10.1007/s11042-016-3511-2.

P5. Carlson, D., Altakrouri, B., Schrader, A.: Reinventing the share button for physical spaces. In Proc. of: IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops), pp. 318–320. IEEE (2013).

P6. Choi, H.S., Lee, J.Y., Yang, N.R., Rhee, W.S.: User-centric service environment for context aware service mash-up. In Proc., pp. 388–393. IEEE Computer Society (2014).

P7. Cook, D.J., Das, S.K.: Pervasive computing at scale: Transforming the state of the art. *Pervasive and Mobile Computing*, 8(1), 22–35 (2012). doi: 10.1016/j.pmcj.2011.10.004.

P8. Coutaz, J., Crowley, J.L.: Learning about End-User Development for Smart Homes by “Eating Our Own Dog Food”. In Proc. of: CHI 2015 - Workshop on End User Development in the Internet of Things Era, pp. 67–74. (2015).

P9. Funk, M., Korn, O., Schmidt, A.: Enabling end users to program for smart environments. In Proc. of: CHI 2015 - Workshop on End User Development in the Internet of Things Era, pp. 9–14. (2015).

P10. Ghose, A., Sinha, P., Bhaumik, C., Sinha, A., Agrawal, A., Choudhury, A.D.: UbiHeld - Ubiquitous healthcare monitoring system for elderly and chronic patients. In Proc., pp. 1255–1264. (2013).

P11. Gil, M., Giner, P., Pelechano, V.: Personalization for unobtrusive service interaction. *Personal and Ubiquitous Computing*, 16(5), 543–561 (2012). doi: 10.1007/s00779-011-0414-0.

P12. Giordano, S., Puccinelli, D.: When sensing goes pervasive. *Pervasive and Mobile Computing*, 17(Part B), 175–183 (2015). doi: 10.1016/j.pmcj.2014.09.008.

P13. Gong, Q., Li, G., Pang, Y.: Design and implementation of smart home system based on ZigBee technology. *International Journal of Smart Home*, 8(6), 143–156 (2014). doi: 10.14257/ijsh.2014.8.6.14.

P14. Gupta, A., Murarka, P.: A ubiquitous sensor network for domestic energy monitoring and energy consumption optimization. In Proc. of: IEEE International Conference on Green Computing and Communications (GreenCom), pp. 58–65. IEEE (2012).

P15. Henze, M., Hermerschmidt, L., Kerpen, D., Haussling, R., Rumpe, B., Wehrle, K.: User-driven privacy enforcement for cloud-based services in the internet of things. In Proc. of: International Conference on Future Internet of Things and Cloud (FiCloud), pp. 191–196. IEEE (2014).

P16. Ho, J.C.F.: Configuring Devices as End-User Programming in the Era of Internet of Things. In Proc. of: CHI 2015 - Workshop on End User Development in the Internet of Things Era, pp. 45–48. (2015).

P17. Huang, J., Cakmak, M.: Supporting mental model accuracy in trigger-action programming. In Proc. of: ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp), pp. 215–225. ACM, New York, NY, USA (2015).

P18. Jacobsson, A., Boldt, M., Carlsson, B.: On the risk exposure of smart home automation systems. In Proc. of: International Conference on Future Internet of Things and Cloud (FiCloud), pp. 183–190. IEEE (2014).

P19. Jinhui, H., ZiYu, X., Xu, X., Siyang, T.: Zigbee-based intelligent home furnishing. *International Journal of Smart Home*, 9(1), 61–68 (2015). doi: 10.14257/ijsh.2015.9.1.07.

P20. Kamilaris, A., Pitsillides, A., Trifa, V.: The Smart Home meets the Web of Things. *International Journal of Ad Hoc and Ubiquitous Computing*, 7(3), 145–154 (2011). doi: 10.1504/IJAHUC.2011.040115.

P21. Kamilaris, A., Pitsillides, A., Yiallouros, M.: Building energy-aware smart homes using web technologies. *Journal of Ambient Intelligence and Smart Environments*, 5(2), 161–186 (2013). doi: 10.3233/AIS-130,201.

P22. Khan, V.J., Markopoulos, P., Eggen, B., Metaxas, G.: Evaluation of a pervasive awareness system designed for busy parents. *Pervasive and Mobile Computing*, 6(5), 537–558 (2010). doi: 10.1016/j.pmcj.2010.03.003.

P23. Kim, Y., Lee, S., Chong, I.: Orchestration in Distributed Web-of-Objects for Creation of User-Centered IoT Service Capability. *Wireless Personal Communications*, 78(4), 1965–1980 (2014). doi: 10.1007/s11277-014-2056-9.

P24. Kouicem, A., Chibani, A., Tari, A., Amirat, Y., Tari, Z.: Dynamic services selection approach for the composition of complex services in the

web of objects. In Proc. of: IEEE World Forum on Internet of Things (WF-IoT), pp. 298–303. IEEE (2014).

P25. Kubitza, T.: Towards a Toolkit for the Rapid Creation and Programming of Smart Environments. In Proc. of: CHI 2015 - Workshop on End User Development in the Internet of Things Era, pp. 61–66. (2015).

P26. Lee, C.T., Yang, C.H., Chang, C.M., Kao, C.Y., Tseng, H.M., Hsu, H., Chou, P.H.: A smart energy system with distributed access control. In Proc. of: IEEE Internet of Things (iThings), 2014 IEEE International Conference on, and Green Computing and Communications (GreenCom), IEEE and Cyber, Physical and Social Computing(CPSCom), pp. 53–60. IEEE (2014).

P27. Lee, J., Garduño, L., Walker, E., Burleson, W.: A tangible programming tool for creation of context-aware applications. In Proc. of: ACM international joint conference on Pervasive and ubiquitous computing (UbiComp), pp. 391–400. ACM (2013).

P28. Lee, N.K., Lee, H.W., Ryu, W.: Considerations for web of object service architecture on IoT environment. International *Journal of Smart Home*, 9(1), 195–202 (2015). doi: 10.14257/ijsh.2015.9.1.21.

P29. Lin, H.T.: Implementing smart homes with open source solutions. *International Journal of Smart Home*, 7(4), 289–296 (2013).

P30. Lucci, G., Paternò, F.: Analysing How Users Prefer to Model Contextual Event-Action Behaviours in Their Smartphones. In: Díaz, P., Pipek, V., Ardito, C., Jensen, C., Aedo, I., Boden, A. (eds.) End-User Development: 5th International Symposium, IS-EUD 2015, Madrid, Spain, May 26–29, 2015. Proceedings, pp. 186–191. Springer International Publishing, Cham (2015).

P31. Martin, D., Lamsfus, C., Alzua-Sorzabal, A., Torres-Manzanera, E.: Empowering End-Users to Develop Context-Aware Mobile Applications Using a Web Platform. In Proc. of: International Conference on Future Internet of Things and Cloud (FiCloud), pp. 139–145. IEEE (2014).

P32. Miclaus, A., Riedel, T., Beigl, M.: End-user installation of heterogeneous home automation systems using pen and paper interfaces and dynamically generated documentation. In Proc. of: International Conference on the Internet of Things (IOT) pp. 19–24. IEEE (2014).

P33. Moreno, M.V., Ramos, J.L.H., Skarmeta, A.F.: User role in IoT-based systems. In Proc., pp. 141–146. IEEE Computer Society (2014).

P34. Nakagawa, M., Tsukada, K., Siio, I.: Smart makeup system: Supporting makeup using lifelog sharing. In Proc. of: 13th international conference on Ubiquitous computing (UbiComp), pp. 483–484. ACM, New York, NY, USA (2011).

P35. Palviainen, M., Kuusijärvi, J., Ovaska, E.: Framework for end-user programming of cross-smart space applications. Sensors (Switzerland), 12(11), 14,442–14,466 (2012). doi: 10.3390/s121114442.

P36. Palviainen, M., Kuusijärvi, J., Ovaska, E.: A semi-automatic end-user programming approach for smart space application development. *Pervasive and Mobile Computing*, 12, 17–36 (2014). doi: 10.1016/j.pmcj.2013.04.002.

P37. Reitberger, W., Spreicer, W., Fitzpatrick, G.: Situated and mobile displays for reflection on shopping and nutritional choices. *Personal and Ubiquitous Computing*, 18(7), 1721–1735 (2014). doi: 10.1007/s00779-014-0781-4.

P38. Stavropoulos, T.G., Kontopoulos, E., Bassiliades, N., Argyriou, J., Bikakis, A., Vrakas, D., Vlahavas, I.: Rule-based approaches for energy savings in an ambient intelligence environment. *Pervasive and Mobile Computing*, 19, 1–23 (2015). doi: 10.1016/j.pmcj.2014.05.001.

P39. Surie, D., Janlert, L.E., Pederson, T., Roy, D.: Egocentric interaction as a tool for designing ambient ecologies - The case of the easy ADL ecology. *Pervasive and Mobile Computing*, 8(4), 597–613 (2012). doi: 10.1016/j.pmcj.2011.12.004.

P40. Surie, D., Pederson, T., Janlert, L.E.: The easy ADL home: A physical-virtual approach to domestic living. *Journal of Ambient Intelligence and Smart Environments*, 2(3), 287–310 (2010). doi: 10.3233/AIS-2010-0074.

P41. Turner, K.J.: Flexible management of smart homes. *Journal of Ambient Intelligence and Smart Environments*, 3(2), 83–109 (2011). doi: 10.3233/AIS-2011-0100.

P42. Turner, K.J.: Managing home care with loose goals and policies. *Journal of Ambient Intelligence and Smart Environments*, 6(5), 489–506 (2014). doi: 10.3233/AIS-140,277.

P43. Ur, B., McManus, E., Pak Yong Ho, M., Littman, M.L.: Practical trigger-action programming in the smart home. In Proc. of: SIGCHI Conference on Human Factors in Computing Systems, pp. 803–812. ACM, New York, NY, USA (2014).

P44. Valtolina, S., Barricelli, B.R.: End-user Empowerment in Lifelogging Activities. In Proc. of: CHI 2015 - Workshop on End User Development in the Internet of Things Era, pp. 27–32. (2015).

P45. Woo, J.-b., Lim, Y.-k.: User experience in do-it-yourself-style smart homes. In Proc. of: ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp), pp. 779–790. ACM, New York, NY, USA (2015).

P46. Yan, H., Cousin, B., Gueguen, C., Vuichard, J.P.: Refined overlay power management in the home environment. In Proc. of: IEEE Internet of Things (iThings), 2014 IEEE International Conference on, and Green Computing and Communications (GreenCom), IEEE and Cyber, Physical and Social Computing(CPSCom), pp. 546–553. IEEE (2014).

P47. You, C.W., Kao, H.L., Ho, B.J., Chen, N.C., Hsieh, Y.H., Huang, P., Chu, H.H.: ThermalProbe: Exploring the use of thermal identification for per-user energy metering. In Proc. of: IEEE Internet of Things (iThings), 2014 IEEE International Conference on, and Green Computing and Communications (GreenCom), IEEE and Cyber, Physical and Social Computing(CPSCom), pp. 554–561. IEEE (2014).

P48. Zhong, Y., Suo, Y., Xu, W., Yu, C., Guo, X., Zhao, Y., Shi, Y.: Smart home on smart phone. In Proc. of: 13th international conference on Ubiquitous computing (UbiComp), pp. 467–468. ACM, New York, NY, USA (2011).

Appendix B: System Usability Score (SUS) and Net Promoter Score (NPS)

This appendix reports the SUS and NPS questionnaires administered to the participants in the comparative study.

The *System Usability Score* (*SUS*) gives an overview of the user's subjective usability evaluation of a given system. It is a closed-ended questionnaire encompassing 10 statements on an ordinal 5-point Likert scale from "strongly disagree" to "strongly agree", as shown in

This questionnaire was chosen for its reliability, brevity and wide adoption (Borsci et al., 2009).

For each returned questionnaire, SUS computes a single score ranging from 0 to 100 for the tested system. A SUS score above a 68 would be considered above average and anything below 68 is below average. In (Bangor et al., 2008), a conventional threshold of 70 is proposed to consider acceptable the overall usability of the evaluated system. This score is calculated as follows:

- For odd items (1,3,5,7,9): subtract one from the user response.
- For even items (2,4,6,8,10): subtract the user responses from 5.
- This scales all values from 0 to 4 (with 4 being the most positive response).
- Add up the converted responses for each user and multiply that total by 2.5. This converts the range of possible values from 0–40 to 0–100.

Table 12
The SUS questionnaire.

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently.	1	2	3	4	5
2. I found the system unnecessarily complex.	1	2	3	4	5
3. I thought the system was easy to use.	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system.	1	2	3	4	5
5. I found the various functions in this system were well integrated.	1	2	3	4	5
6. I thought there was too much inconsistency in this system.	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly.	1	2	3	4	5
8. I found the system very cumbersome to use.	1	2	3	4	5
9. I felt very confident using the system.	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system.	1	2	3	4	5

Table 13
The NPS questionnaire.

How likely is it that you would recommend this application to a friend or colleague?										
0	1	2	3	4	5	6	7	8	9	10
Not at all likely				Neutral						Likely

The final score is obtained by averaging the score of each questionnaires and indicates the overall participants' satisfaction of the application.

The *Net Promoter Score* (*NPS*) is a single closed-ended question asking a person to rate on an 11-point scale the likelihood of recommending the tool to a friend or colleague, see Table 13.

Based on their rating, respondents are then classified in 3 categories: detractors, passives and promoters. 'Detractors' give a score lower than or equal to 6; they are not particularly thrilled by the product or the service. 'Passives' give a score of 7 or 8; they are somewhat satisfied. Finally, 'promoters' answer 9 or 10; they love the products. The NPS is then calculated by subtracting the percentage of detractors from the percentage of promoters. Thus, if all of the respondents are detractors an NPS of -100 is obtained; instead, if all of the respondents are promoters, the resulting NPS is 100. A positive NPS is felt to be good, but an NPS $>$ 50 is excellent. Originally, NPS was proposed to measure the willingness of a customer to recommend a product or service to others, i.e. as a proxy for gauging the customer's overall satisfaction with a company's product or service and the customer's loyalty to the brand (Reichheld, 2003).

References

Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., Ayyash, M., 2015. Internet of things: a survey on enabling technologies, protocols, and applications. IEEE Commun. Surv. Tutorials 17 (4), 2347–2376. http://dx.doi.org/10.1109/COMST.2015.2444095.

Ardito, C., Buono, P., Costabile, M.F., Lanzilotti, R., Piccinno, A., 2012. End users as co-designers of their own tools and products. J. Visual Lang. Comput. 23 (2), 78–90. http://dx.doi.org/10.1016/j.jvlc.2011.11.005.

Ardito, C., Costabile, M.F., Desolda, G., Latzina, M., Matera, M., 2015. Making mashups actionable through elastic design principles. In: Díaz, P., Pipek, V., Ardito, C., Jensen, C., Aedo, I., Boden, A. (Eds.), End-User Development: 5th International Symposium, IS-EUD 2015, Madrid, Spain, May 26-29, 2015. Proceedings. Cham: Springer International Publishing, pp. 236–241. http://dx.doi.org/10.1007/978-3-319-18425-8_22.

Atzori, L., Iera, A., Morabito, G., 2010. The internet of things: a survey. Comput. Netw. 54 (15), 2787–2805. http://dx.doi.org/10.1016/j.comnet.2010.05.010.

Bafandeh Mayvan, B., Rasoolzadegan, A., Ghavidel Yazdi, Z., 2017. The state of the art on design patterns: a systematic mapping of the literature. J. Syst. Softw. 125 93–118. http://dx.doi.org/10.1016/j.jss.2016.11.030.

Baldassarre, M.T., Carver, J., Dieste, O., Juristo, N., 2014. Replication types: towards a shared taxonomy. In: Proc. of 18th International Conference on Evaluation and Assessment in Software Engineering, London, England, United Kingdom, pp. 1–4. May 13–14. doi:10.1145/2601248.2601299.

Bangor, A., Kortum, P.T., Miller, J.T., 2008. An empirical evaluation of the system usability scale. Int. J. Human Comput. Interact. 24 (6), 574–594. http://dx.doi.org/10.1080/10447310802205776.

Barricelli, B.R., Valtolina, S., 2015. Designing for end-user development in the internet of things. In: Díaz, P., Pipek, V., Ardito, C., Jensen, C., Aedo, I., Boden, A. (Eds.), End-User Development 9083. Springer International Publishing, pp. 9–24. http://dx.doi.org/10.1007/978-3-319-18425-8_2.

Benzi, F., Cabitza, F., Fogli, D., Lanzilotti, R., Piccinno, A., 2015. Gamification techniques for rule management in ambient intelligence. In: De Ruyter, B., Kameas, A., Chatzimisios, P., Mavrommati, I. (Eds.), Ambient Intelligence 9425. Springer International Publishing, pp. 353–356. http://dx.doi.org/10.1007/978-3-319-26005-1_25.

Benzi, F., Fogli, D., Guida, G., 2017. Towards natural interaction with smart homes. In: Proc. of 12th Biannual Conference on Italian SIGCHI Chapter, Cagliari, Italy, pp. 1–6. http://dx.doi.org/10.1145/3125571.3125581.

Bianchi, A., Caivano, D., Visaggio, G., 2000. Method and process for iterative reengineering of data in a legacy system. In: Proc. of Proceedings Seventh Working Conference on Reverse Engineering. 2000. pp. 86–96. http://dx.doi.org/10.1109/WCRE.2000.891456.

Borsci, S., Federici, S., Lauriola, M., 2009. On the dimensionality of the system usability scale: a test of alternative measurement models. Cognit. Process. 10 (3), 193–197. http://dx.doi.org/10.1007/s10339-009-0268-9.

Brich, J., Walch, M., Rietzler, M., Weber, M., Schaub, F., 2017. Exploring end user programming needs in home automation. ACM Trans. Comput. Human Interact. 24 (2). http://dx.doi.org/10.1145/3057858.

Brooke, J., Jordan, P.W., Weerdmeester, B., Thomas, A., McLelland, I.L., 1996. SUS: a quick and dirty usability scale. In: Jordan, P.W., Thomas, B., Weerdmeester, B.A., McClelland, A.L. (Eds.), Usability Evaluation in Industry. Taylor and Francis.

Cabitza, F., Fogli, D., Lanzilotti, R., Piccinno, A., 2017. Rule-based tools for the configuration of ambient intelligence systems: a comparative user study. Multimedia Tools Appl. 76 (4), 5221–5241. http://dx.doi.org/10.1007/s11042-016-3511-2.

Cabitza, F., Fogli, D., Piccinno, A., 2014. Fostering participation and co-evolution in sentient multimedia systems. J. Visual Lang. Comput. 25 (6), 684–694. http://dx.doi.org/10.1016/j.jvlc.2014.10.014.

Caivano, D., 2005. Continuous software process improvement through statistical process control. In: Proc. of Ninth European Conference on Software Maintenance and Reengineering, 21–23 March 2005, pp. 288–293. http://dx.doi.org/10.1109/CSMR.2005.20.

Caivano, D., Cassano, F., Fogli, D., Lanzilotti, R., Piccinno, A., 2017. We@Home: a gamified application for collaboratively managing a smart home. In: De Paz, J.F., Julián, V., Villarrubia, G., Marreiros, G., Novais, P. (Eds.), Ambient Intelligence– Software and Applications – 8th International Symposium on Ambient Intelligence (ISAmI 2017). Cham: Springer International Publishing, pp. 79–86. http://dx.doi.org/10.1007/978-3-319-61118-1_11.

Caivano, D., Lanubile, F., Visaggio, G., 2001. Software renewal process comprehension using dynamic effort estimation. In: Proc. of Proceedings IEEE International Conference on Software Maintenance. ICSM 2001, 2001, pp. 209–218. http://dx.doi.org/10.1109/ICSM.2001.972734.

Carver, J.C., Juristo, N., Baldassarre, M.T., Vegas, S., 2014. Replications of software engineering experiments. Empirical Softw. Eng. 19 (2), 267–276. http://dx.doi.org/10.1007/s10664-013-9290-8.

Cook, D.J., Augusto, J.C., Jakkula, V.R., 2009. Ambient intelligence: Technologies, applications, and opportunities. Pervasive Mob. Comput. 5 (4), 277–298. http://dx.doi.org/10.1016/j.pmcj.2009.04.001.

Costabile, M.F., Fogli, D., Fresta, G., Mussio, P., Piccinno, A., 2003. Building

environments for end-user development and tailoring. In: Proc. of IEEE Symposium on Human Centric Computing Languages and Environments, Auckland, New Zealand, pp. 31–38. http://dx.doi.org/10.1109/HCC.2003.1260199.
Costabile, M.F., Fogli, D., Mussio, P., Piccinno, A., 2006. End-user development: the software shaping workshop approach. In: Lieberman, H., Paternò, F., Wulf, V. (Eds.), End User Development 9. Springer, Dordrecht, The Netherlands, pp. 183–205. http://dx.doi.org/10.1007/1-4020-5386-X_9.
Costabile, M.F., Fogli, D., Mussio, P., Piccinno, A., 2007. Visual interactive systems for end-user development: a model-based design methodology. IEEE Trans. Syst. Man Cybern. Part A Syst. Humans 37 (6), 1029–1046. http://dx.doi.org/10.1109/TSMCA.2007.904776.
Coutaz, J., Crowley, J.L., 2016. A first-person experience with end-user development for smart homes. IEEE Pervasive Comput. 15 (2), 26–39. http://dx.doi.org/10.1109/MPRV.2016.24.
Cypher, A., 1993. Watch What I do: Programming By Demonstration. MIT Press.
Dahl, Y., Svendsen, R.-M., 2011. End-user composition interfaces for smart environments: a preliminary study of usability factors. In: Marcus, A. (Ed.), Design, User Experience, and Usability. Theory, Methods, Tools and Practice 6770. Springer, Berlin Heidelberg, pp. 118–127. http://dx.doi.org/10.1007/978-3-642-21708-1_14.
Davidoff, S., Lee, M.K., Yiu, C., Zimmerman, J., Dey, A.K., 2006. Principles of smart home control. In: Dourish, P., Friday, A. (Eds.), UbiComp 2006: Ubiquitous Computing 4206. Springer, Berlin Heidelberg, pp. 19–34. http://dx.doi.org/10.1007/11853565_2.
Desolda, G., Ardito, C., Matera, M., 2016. EFESTO: a platform for the end-user development of interactive workspaces for data exploration. In: Daniel, F., Pautasso, C. (Eds.), Rapid Mashup Development Tools: First International Rapid Mashup Challenge, RMC 2015, Rotterdam, The Netherlands, June 23, 2015, Revised Selected Papers. Cham: Springer International Publishing, pp. 63–81. http://dx.doi.org/10.1007/978-3-319-28727-0_5.
Desolda, G., Ardito, C., Matera, M., 2017. Empowering end users to customize their smart environments: model, composition paradigms, and domain-specific tools. ACM Trans. Comput. Human Interact. 24 (2), 1–52. http://dx.doi.org/10.1145/3057859.
Dey, A.K., Sohn, T., Streng, S., Kodama, J., 2006. iCAP: interactive prototyping of context-aware applications. In: Fishkin, K.P., Schiele, B., Nixon, P., Quigley, A. (Eds.), Pervasive Computing 3968. Springer, Berlin Heidelberg, pp. 254–271. http://dx.doi.org/10.1007/11748625_16.
Ferreira, W., Baldassarre, M.T., Soares, S., 2018. Codex: A metamodel ontology to guide the execution of coding experiments. Comput. Stand. Interfaces 59 35–44. https://doi.org/10.1016/j.csi.2018.02.003.
Fischer, G., Fogli, D., Piccinno, A., 2017. Revisiting and broadening the meta-design framework for end-user development. In: Paternò, F., Wulf, V. (Eds.), New Perspectives in End-User Development. Cham: Springer International Publishing, pp. 61–97. http://dx.doi.org/10.1007/978-3-319-60291-2_4.
Fogli, D., Lanzilotti, R., Piccinno, A., 2016a. End-user development tools for the smart home: a systematic literature review. In: Streitz, N., Markopoulos, P. (Eds.), Distributed, Ambient and Pervasive Interactions: 4th International Conference, DAPI 2016, Held as Part of HCI International 2016, Toronto, ON, Canada. Cham: Springer International Publishing, pp. 69–79. July 17–22, 2016, Proceedings. doi:10.1007/978-3-319-39862-4_7.
Fogli, D., Lanzilotti, R., Piccinno, A., Tosi, P., 2016b. AmI@Home: a game-based collaborative system for smart home configuration. In: Proc. of International Working Conference on Advanced Visual Interfaces (AVI), Bari, Italy, pp. 308–309. http://dx.doi.org/10.1145/2909132.2926083.
Fogli, D., Peroni, M., Stefini, C., 2017. ImAtHome: making trigger-action programming easy and fun. J. Visual Lang. Comput. 42, 60–75. Supplement C. doi:10.1016/j.jvlc.2017.08.003.
Ghiani, G., Manca, M., Paternò, F., Santoro, C., 2017. Personalization of context-dependent applications through trigger-action rules. ACM Trans. Comput. Human Interact. 24 (2), 1–33. http://dx.doi.org/10.1145/3057861.
Graziano, A.M., Raulin, M.L., 2012. Research Methods: A Process of Inquiry. Pearson (8th Edition).
Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M., 2013. Internet of Things (IoT): a vision, architectural elements, and future directions. Future Gener. Comput. Syst. 29 (7), 1645–1660. https://doi.org/10.1016/j.future.2013.01.010.
Lucci, G., Paternò, F., 2014. Understanding end-user development of context-dependent applications in smartphones. In: Sauer, S., Bogdan, C., Forbrig, P., Bernhaupt, R., Winckler, M. (Eds.), Human-Centered Software Engineering 8742. Springer, Berlin Heidelberg, pp. 182–198. http://dx.doi.org/10.1007/978-3-662-44811-3_11.
Nardi, B.A., Engeström, Y., 1999. A web on the wind: the structure of invisible work. Comput. Supported Cooperative Work (CSCW) 8 (1), 1–8. http://dx.doi.org/10.1023/A:1008694621289.
Nielsen, J., 1993. Usability Engineering. Academic Press, San Diego.
Pane, J.F., Ratanamahatana, C.A., Myers, B.A., 2001. Studying the language and structure in non-programmers' solutions to programming problems. Int. J. Hum. Comput. Stud. 54 (2), 237–264. https://doi.org/10.1006/ijhc.2000.0410.
Preece, J., Sharp, H., Rogers, Y., 2015. Interaction Design: Beyond Human-Computer Interaction, 4th Edition. John Wiley & Sons Ltd.
Reichheld, F.F., 2003. The one number you need to grow. Harv. Bus. Rev. 81 (12), 46–55.
Russo, P., Lanzilotti, R., Costabile, M.F., Pettit, C.J., 2017. Adoption and use of software in land use planning practice: a multiple-country study. Internat. J. Human Comput. Interact. 1–16. http://dx.doi.org/10.1080/10447318.2017.1327213.
Sadri, F., 2011. Ambient intelligence: a survey. ACM Comput. Surveys 43 (4), 1–66. http://dx.doi.org/10.1145/1978802.1978815.
Sas, C., Neustaedter, C., 2017. Exploring DIY practices of complex home technologies. ACM Trans. Comput. Human Interact. 24 (2). http://dx.doi.org/10.1145/3057863.
Spahn, M., Doerner, C., Wulf, V., 2008. End user development: approaches towards a flexible software design. In: Proc. of 16th European Conference on Information Systems, Galway, Ireland, June 9–11.
Tetteroo, D., Markopoulos, P., Valtolina, S., Paternò, F., Pipek, V., Burnett, M., 2015. End-user development in the internet of things era. In: Proc. of ACM Conference on Human Factors in Computing Systems - Extended Abstracts (CHI), Seoul, Republic of Korea, pp. 2405–2408. April 18–23. doi:10.1145/2702613.2702643.
Ur, B., McManus, E., Pak Yong Ho, M., Littman, M.L., 2014. Practical trigger-action programming in the smart home. In: Proc. of SIGCHI Conference on Human Factors in Computing Systems, Toronto, Ontario, Canada, pp. 803–812. http://dx.doi.org/10.1145/2556288.2557420.

Danilo Caivano is currently assistant professor at the Computer Science Department of the University of Bari “Aldo Moro”. He received the PhD in Computer Science from the University of Bari. He carries out his research in the Software Engineering Laboratory at the Computer Science Department. Since 2007 he is Chief Executive Officer of SER&Practices (www.serandp.com), a Spin Off company of the University of Bari that he has contributed to start up. He has managed several large and complex projects, many of which are focused on research and development in partnership with Universities, Research Centers, as well as national and international companies. He is actively involved in the Project Management Institute - Southern Italy Chapter (www.pmi.org) and in the International Software Engineering Research Network (isern.iese.de). His research and teaching activities focus on topics related to Software Engineering with emphasis on Project and Process Management in collocated and distributed contexts and on software development, maintenance and testing. He is member of the editorial board of several prestigious international journals and conferences.

Daniela Fogli is currently associate professor at the Department of Information Engineering of the University of Brescia, Italy. She received the Laurea degree in Computer Science from the University of Bologna, Italy, in 1994 and the PhD degree in Information Engineering from the University of Brescia, in 1998. From 1998 to 2000, she was a post-doc grant holder at the Joint Research Centre of the European Commission. Since 2000 to April 2015, she has been assistant professor at the University of Brescia. Her research interests include meta-design, end-user development, web usability and accessibility, decision support systems. She is member of ACM (Association of Computing Machinery) and SIGCHI Italy (the Italian Chapter of ACM's Special Interest Group on Computer-Human Interaction).

Rosa Lanzilotti is currently associate professor at the Computer Science Department of the University of Bari "Aldo Moro". She got her PhD in Computer Science from the University of Bari on May 2006. Her research interests are on Human-Computer Interaction and, in particular, usability and user experience, accessibility, technology-enhanced learning, statistical methods for data analysis, end-user development, UX practices in industry and public institutions. She is member of ACM (Association of Computing Machinery), ACM SIGCHI (ACM's Special Interest Group on Computer-Human Interaction) and SIGCHI Italy (the Italian Chapter of ACM SIGCHI).

Antonio Piccinno is currently assistant professor at the Computer Science Department of the University of Bari "Aldo Moro". He received a Laurea Degree with full marks and honors in Computer Science from the University of Bari. In 2005 he got the PhD in Computer Science from the University of Bari. His research interests focus on End-User Development, visual language theories, human-computer interaction, multimedia and multimodal interaction, web interfaces. He is in the Steering Committee of the International Symposium on End-User Development and served as Program Co-Chair in the third edition (IS-EUD 2011) held in Torre Canne (Brindisi), Italy in June 2011. He is member of ACM (Association of Computing Machinery), ACM SIGCHI (ACM's Special Interest Group on Computer-Human Interaction) and SIGCHI Italy (the Italian Chapter of ACM SIGCHI).

Fabio Cassano is currently a PhD Student at the Computer Science Department of the University of Bari "Aldo Moro" and member of the Interaction, Visualization, Usability & UX (IVU) Lab. He got at the Polytechnic University of Bari the Bachelor Degree in Computer Engineering in 2013 and the Master Science Degree in Computer Engineering with honors, in 2015, with a thesis in Human-Machine Interaction. His research interests focus on End-User Development, Ambient Intelligence, Smart Home, Internet of Things, Home Assistance, Active Aging. He is member of SIGCHI Italy (the Italian Chapter of ACM SIGCHI) and EUSSET (EUropean Society for Socially-Embedded Technologies).