Supporting Participation in Smart Home Control: Beyond Trigger-Action Programming

Daniela Fogli

Dipartimento di Ingegneria dell'Informazione
Università degli Studi di Brescia, Brescia, Italy
daniela.fogli@unibs.it

Abstract. The user interfaces proposed so far for smart home control are usually based on event-condition-action rules, which are created by means of "if-then" instructions. This is a form of trigger-action programming that software developers consider suitable to pure end users. However, it has been observed that end users who are not knowledgeable in computer programming find this approach unnatural and difficult to learn. Therefore, even though they want to participate in smart home configuration and tailoring, they do not actually participate. This paper discusses this problem and presents a mobile app that aims at fostering participation of all family members in smart home control. Further issues related to user participation in this domain are finally explored.

Keywords: end-user development, smart home, trigger-action programming

1 Introduction

The control and configuration of a smart home is often regarded as a difficult activity that should be carried out by some expert in software and hardware technology [3]. Demeure and colleagues discussed this problem through a field study they conducted with the participation of ten households using different home automation systems for a long period of time. In all households, there was always only one member of the family (called "guru" in the study) in charge of installing, configuring and managing the smart home. This family member was always a male adult, knowledgeable in computer programming [3]. Other family members would like to participate in the creation of new home behaviors, but they had to delegate this task to the guru. Indeed, given the user interfaces and the computer-oriented languages available in the home automation systems, they were not capable of participating at all.

This issue has been investigated in literature works and commercial products through some analyses of user interfaces of systems for smart home control [1][5]. From these analyses, it emerged that the event-condition-action (ECA) rule-based paradigm is the most used in such user interfaces. These interfaces allow users to carry out a form of *trigger-action* ("if, then") *programming* [10], where users are driven through the interface in setting up the "if" and "then" parts of a rule. The "if" part usually includes an event triggering the rule and some (optional) conditions;

Proc. of Fourth International Workshop on Cultures of Participation in the Digital Age - CoPDA 2016
Gothenburg (Sweden), October 23, 2016 (published at http://ceur-ws.org).

whilst, the "then" part encompasses a list of actions to be activated on some devices available in the house.

In this position paper, a different user interface for the creation of ECA rules is discussed. It is aimed at supporting users to perform this activity in an unwitting manner [2], actually going beyond trigger-action programming. The first version of an iOS app, called ImAtHome, developed over the Apple HomeKit framework, is presented in the following to show this user interface.

2 From "want to" to "being able to" participate

The user interfaces proposed so far for smart home configuration and adaptation to the inhabitants' needs pay attention to facilitating the creation of "if-then" instructions by end users, according to a computer-oriented approach to End-User Programming (EUP) [9] and End-User Development (EUD) [8]. However, this perspective forces users to think in terms of "if-then" constructs and, often, also of AND/OR logic propositions, whenever complex antecedent and consequent parts must be created. As a consequence, end users who do not completely understand these technical aspects may consider rule creation as too difficult for their knowledge and skills, and do not participate in the shaping of their home behavior, even when they would like to.

In general, EUP and EUD often encompass special-purpose languages to carry out the programming activity and cope with software engineering issues [7]. In this way, research scholars in EUP and EUD neglect that, nowadays, most end users would like to carry out their usual activities in an easier and more efficient manner, possibly with the help of software tools, but without the need of acquiring any new skills to use them, which requires time and learning efforts.

There are situations where end users are interested in participating in system development, but they are not able to participate. This is not the case of content creation in wikis, app installation on personal smartphones, and sharing videos on dedicated platforms. All these activities have been made natural to users by means of user interfaces and interaction metaphors that are well integrated both with existing software environments and with users' expertise.

Following these considerations, the interaction metaphor here proposed for the creation of smart home behaviors is centered on two tasks, namely the definition of *scenes* followed by the definition of *rules* starting from available scenes. Scenes are sets of device actions, while rules make the house able of activating itself some given scenes. The metaphor also encompasses a guided but easy way of defining events and conditions for triggering rules. These ideas have been implemented in ImAtHome, an iOS mobile application. The iOS guidelines for application development have been followed by making all activities similar to the usual activities users can perform in iPhone apps, and terms such as "if", "then" or "do" never appear in ImAtHome. The goal is making all users able to participate in smart home configuration and control by unwittingly perform trigger-action programming.

Proc. of Fourth International Workshop on Cultures of Participation in the Digital Age - CoPDA 2016
Gothenburg (Sweden), October 23, 2016 (published at http://ceur-ws.org).

3 ImAtHome: Beyond Trigger-Action Programming

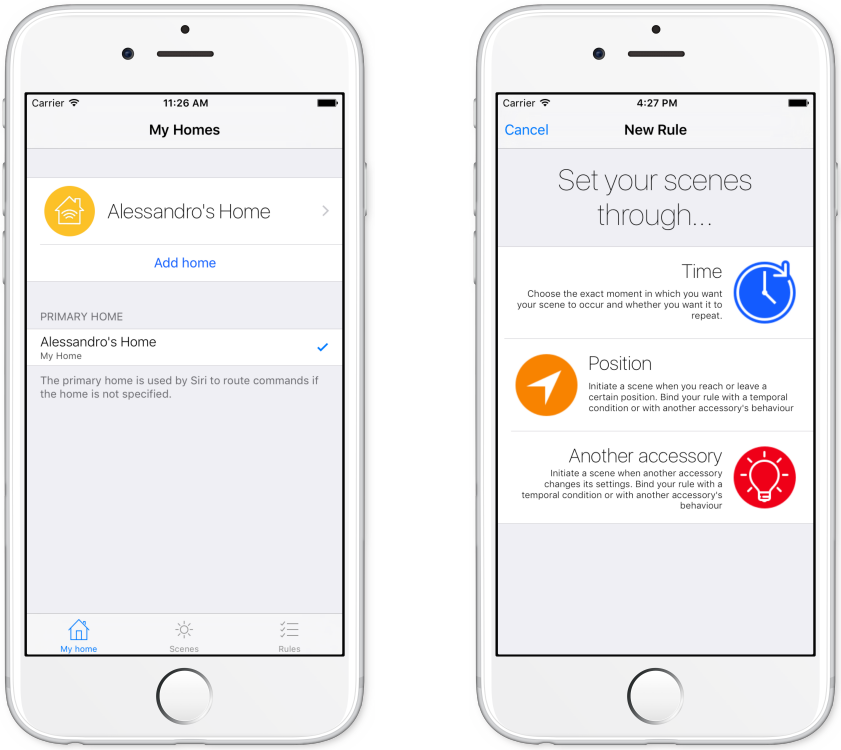
HomeKit is a framework made available in iOS for communicating with and controlling connected home automation accessories. It is mainly a communication protocol that supports the integration and interoperability of different kinds of accessories. Several companies are developing accessories (conditioners, thermostats, light bulbs, etc.) compatible with HomeKit. As a consequence, several dedicated apps are being developed to control such devices. The advantages given by the compatibility with HomeKit are that accessories can be controlled through Siri and may be included in the creation of a scene. However, the main drawback is that each accessory keeps on being controlled only by its corresponding app. ImAtHome exploits the common communication protocol for interacting with many kinds of accessories or combinations of them, by proposing itself as a hub for controlling one's own smart home: all compatible accessories are made available on the smartphone in a unique app, with the same interaction style and the possibility to work in combination one another.

The app includes three sections, as one can observe in the bottom tab bar of Figure 1(a). In the first section, entitled "My home", the user can access or define a home and its rooms, as well as all the accessories associated with each room. Accessories are automatically recognized by the app through HomeKit, and the user may bind them to rooms by means of OCR technology.

The second section, "Scenes", presents a default list of void scenes available in the HomeKit database, that is "Good morning", "I'm leaving", "I'm home" and "Good night". The user may complete them or create a new one. To create a new scene, the user gives it a name and defines a sequence of actions, by setting the characteristics of the accessories involved in each action. When the new scene is saved, the user can manually activate it with a tap.

The third section ("Rules") is that devoted to the creation of ECA rules. When the user decides to create a new rule, the screenshot in Figure 1(b) is shown. Here the message at the top tells the user to choose among three options to trigger his/her scenes by "Time", "Position" or "Another accessory". Then a screen appears where the user may define the details of an event related to the chosen option, defines additional conditions, and selects one or more (pre-defined or user-defined) scenes that must be activated. Note that the three options mentioned above correspond to the three conditional relationships for triggering scenes, which are supported in HomeKit. Selecting one of them allows defining the "event" part of an ECA rule. However, differently from the interaction with other tools, here the user does not need to know that the "if" part of an "if-then" construct is under creation. Then, by defining the additional conditions and selecting the scenes to activate, the users actually creates the condition part and action part of an ECA rule respectively. The user can do it without being aware that he/she is actually creating an ECA rule or, in other terms, that he/she is performing trigger-action programming. Moreover, differently from other user interfaces for ECA rule definition, ImAtHome requires to define action sets (scenes) first, and then relate them to events and conditions. This allows users to activate scenes manually if needed and use them in several different rules.

Proc. of Fourth International Workshop on Cultures of Participation in the Digital Age - CoPDA 2016
Gothenburg (Sweden), October 23, 2016 (published at http://ceur-ws.org).



(a) (b)

Fig. 1. The app ImAtHome: a) "My Home" page and (b) Rule settings.

4 Discussion and Conclusion

The app ImAtHome has been presented to illustrate an example of EUP/EUD technique for smart home control that is as much as possible integrated with iOS interaction style. A user test was carried out to evaluate the usability of the application. We involved 14 participants (6 females) of different ages, education degrees and professional backgrounds, and asked them to perform five tasks of increasing complexity using the iPhone simulator and the HomeKit Accessory Simulator. All users, without any previous training, were able to use ImAtHome efficiently and effectively (the reader could find in [6] some details about the results of this experimentation).

However, the tradeoff existing between usability and powerfulness of this app should be evaluated and addressed. Other applications, such as Tasker (http://tasker.dinglisch.net/), support more complex configuration activities, which are however possible only by software programmers and not by pure end users. A "rich ecology of participation" [4] should be conceived for smart home control, by allowing users with different levels of expertise to contribute in different ways and at their own pace. This issue is not addressed in existing applications yet; in general, the multiple

Proc. of Fourth International Workshop on Cultures of Participation in the Digital Age - CoPDA 2016
Gothenburg (Sweden), October 23, 2016 (published at http://ceur-ws.org).

user control of a smart home is not supported in current solutions. Indeed, this would open up other problems to be solved. For instance, a variety of social mechanisms, from collaboration to competition, from delegation to reciprocity, should be implemented to stimulate participation. Furthermore, the simultaneous intervention in scene activation or rule creation would require studying mechanisms for solving incoherencies and conflicts.

Acknowledgments. The author wishes to thank Matteo Peroni and Claudia Stefini for the development of the app presented in the paper.

References

1. 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 (2016) DOI: 10.1007/s11042-016-3511-2
2. Costabile, M. F., Mussio, P., Parasiliti Provenza, L., Piccinno, A.: End users as unwitting software developers. In: Proceedings of the 4th international workshop on End-user software engineering (WEUSE '08). ACM, New York, NY, USA, 6-10 (2008)
3. Demeure, A., Caffiau, S., Elias, E., Roux, C.: Building and Using Home Automation Systems: A Field Study. In: Díaz, P., Pipek, V., Ardito, C., Jensen, C., Aedo, I., Boden, A. (eds.) End-User Development. LNCS, vol. 9083, Springer International Publishing, 125-140 (2015)
4. Fischer, G.: Understanding, fostering, and supporting cultures of participation. Interactions, 18(3), 42-53 (2011)
5. Fogli, D., Lanzilotti, R., Piccinno, A.: End-User Development Tools for the Smart Home: A Systematic Literature Review. In: N. Streitz and P. Markopoulos (Eds.): DAPI 2016, LNCS 9749, Springer International Publishing Switzerland, 69–79 (2016)
6. Fogli, D., Peroni, M., Stefini, C.: Smart Home Control through Unwitting Trigger-Action Programming. In: Proceedings of Distributed Multimedia Systems (DMS 2016), Salerno, Italy (2016)
7. Ko, A.J., Abraham, R., Beckwith, L., Blackwell, A., Burnett, M., Erwig, M., Scaffidi, C., Lawrance, J., Lieberman, H., Myers, B., Rosson, M.B., Rothermel, G., Shaw, M., Wiedenbeck, S.: The State of the Art in End-User Software Engineering. ACM Comput. Surv. 43(3), 1-44 (2011)
8. Lieberman, H., Paternò, F., Wulf, V. (eds.): End-User Development. Springer, Dordrecht, The Netherlands (2006)
9. Myers, B., Ko, A.: The Past, Present and Future of Programming in HCI. Institute for Software Research, Paper 782. http://repository.cmu.edu/isr/782 (2009)
10. Ur, B., McManus, E., Pak Yong Ho, M., Littman, M.L.: Practical trigger-action programming in the smart home. In: SIGCHI Conference on Human Factors in Computing Systems, ACM, New York, NY, USA, 803-812 (2014)

Proc. of Fourth International Workshop on Cultures of Participation in the Digital Age - CoPDA 2016
Gothenburg (Sweden), October 23, 2016 (published at http://ceur-ws.org).