A Homemade Pill Dispenser Prototype Supporting Elderly

Paolo Buono, Fabio Cassano, Alessandra Legretto, Antonio Piccinno

Dipartimento di Informatica, Università di Bari “Aldo Moro”, Bari, Italy
{paolo.buono, fabio.cassano1,
alessandra.legretto, antonio.piccinno}@uniba.it

Abstract. People, and mainly elderly people, need a continuous support for different reasons. Recent technologies are offering many possibilities that was not possible to conceive in the past. In particular, the proliferation of IoT devices raise the need to standardize protocols and interaction languages. The aim of this work is to create a device for the management of pills according to the user’s therapy, with Internet of things (IoT) devices and by allowing users to manage the pill dispenser by themselves. The work falls into two main areas of current research: the End-user development (EUD) and the Internet of things (IoT). The main issue we cope with such device is to allow the different therapies for each person and for each drug. We propose the EUDroid system, which provides the end user with the possibility to easily activate LEDs and buzzer related to pills from the users’ smartphone. The user chooses the type of pill to be associated to each LED, the day and time of activation and some other property. A formal language to configure the device has been adopted in order to allow users to build complex conditions for remind to follow the therapy.

Keywords: Internet of Thing, End-User Development, Elderly, Pill Dispenser.

1 Introduction

Computer users have rapidly increased in both number and diversity [1]. They include managers, accountants, engineers, teachers, scientists, health care workers, etc. Many of these people work on tasks that rapidly vary on a yearly, monthly, or even daily basis. Consequently, their software needs are different, complex, and frequently changing. Professional software developers cannot directly meet all of such needs because of their limited domain knowledge and slow development processes [2].

End-user Development (EUD) helps to solve this problem. EUD is defined as "a set of methods, techniques and tools that allow users of software systems, who are acting as non-professional software developers, at some point to create, modify, or extend a software artifact" [3], enabling end users to design or customize the user interface and functionality of a software. End users know both their own context and needs, better more than anybody else. Moreover, they often have real-time awareness of shifts in their respective domains. Through EUD, end users can adapt the software they are using to fit their requirements. A recent view of EUD extends the context of software systems to techniques, methodologies, situations, and socio-technical environments that allow end users to act as professionals in those domains in which they are not professionals, including IoT [4].

Our work address different situations in which a user might want to change the behavior of a software/hardware artifact. In this demo we address the need of many people to take one or more pills daily for a given period. We designed and developed an IoT device that allows the management and distributions of pills for the end users, who are often elderly, and by end users with no IT expertise.

The project EUDdroid has been developed into three parts: the first involves a smartphone app, the second is the pill dispenser that has the role of the "Internet of Things" (IoT) device and the third is a web server that manages the user's requests. IoT devices are objects to be sensed or controlled remotely across an existing network infrastructure, creating opportunities for more direct integration of the physical world into computer-based systems, and resulting in improved efficiency, accuracy and economic benefit in addition to reduced human intervention [5].

The main purpose of the project is to realize an Android application that allows the composition and customization of elementary events, such as the management of a pill for diabetes or cholesterol. The idea to use electronic devices to help elderly to take pills is well known in the literature. Many of the existing works, however, do not allow users to customize the behavior of the machine. As a matter of fact, commercial solutions allow users only to define what pill should be dropped according to the calendar. The final user is not allowed to deeply edit the parameters of the pills dispenser and, to the best of our knowledge, not any formal language has ever been defined to allow multiple users to customize the machine [6]. Moreover, it is not always possible for the end user, to control and edit the pills dispenser through the smartphone with an app. A similar solution like our pill dispenser is proposed in [7].

2 EUDroid formal language

A formal specification can describe unambiguously the behavior of a user interface and may help the designer to easily extend the functionalities of the app, making sure that such functionalities are not in contrast with the previous behavior. Once defined the behavior of the system (in terms of syntax and semantics), new widgets or features can be introduced, adding their semantics. In the following, is reported a formal description of the basic components and the logic of the system [4].

An elementary component (c) is an electronic device that receives an input or gives an output. The elementary components are divided into sensors and actuators. The first are electronic devices that mostly collect and analyze information; the latter are electronic devices that perform operations or "carry out a behavior".

The project includes five actuators: a) red LED (LR), b) blue LED (LB), c) green LED (LG), d) yellow LED (LY) and e) Buzzer (B). All of them can assume the state of active (high level) or inactive (low level). The behavior of the buzzer is the opposite, the low level activates the buzzer, while the high level disables the buzzer. An Ethernet shield let the system communicate through the Internet. A button is the user's input.

When the button is pressed, its state is low and all the actuators are turned off, when it is not pressed, its state is high. Another elementary component is the message (M). It belongs to the category of the actuators because it is an output component. In this work, the message is a text sent to a mobile phone. The goal is to build an app that composes elementary events in order to get composite events.

An elementary event (e) is composed of an action and a trigger. An Action (A) consists in the execution of operations, i.e. the turn on/off elementary components. A Trigger (TR) is the logical condition on one or more actions. In this demo, the trigger is the Calendar (CL) item. A calendar is composed by a start date (SD) and time (T). A Composite Event (CE) is the set of elementary events, so it is composed by more Trigger-Action relations. An elementary component can be any actuator of those provided in the project, listed above, that can carry information such as Delay (DY) and Duration (DU). AT defines the whole actuator's set. Given an actuator $AT = \{LR, LB, LG, LY, B, M\}$, an elementary component c is defined as:

$$c =< x, DY, DY > \ with\ x \in AT, DY \in N, DU \in N \tag{1}$$

For example, $c =< LR, 120{,}60 >$ means that the red LED will be on after 2 minutes (120") compared to the activation of a given trigger, and will have a duration of 1 minute (60"). We defined a sequence of elementary components form action as:

$$A =< c_1, c_2, \dots, c_n > \ where\ c_i \in C, 1 \leq i \leq n \tag{2}$$

C is the set of elementary components. Therefore, as the elementary event is composed of the pair trigger-action we will write:

$$e =< CL, A >, where\ CL =< SD, T > \tag{3}$$

Consequently, given the set of elementary events E, a Composite Event *(CE)* is:
$CE =< e_1, e_2, \dots, e_n > \ where\ e_i \in E, 1 \leq i \leq n.$

The web server hosts a database with the pill dispenser location and the available commands for the specific device. Commands can be expressed in the form: $Cm =< p, s, DY, DU >$. Where p defines the pin position on the Arduino board, s defines the state (high or low) of the elementary component. In the demo the device, based on Arduino, has three tasks: a) checks the Web Server if there is a command associated to its code; b) if there is one it reads the event and run it according to the delay and duration; c) if the user press the button it reset the status of the pill dispenser. As shown in Fig. 1, the physical prototype has currently four compartments for pills. At the bottom right there is the button to reset the current event. At the top right there is a small hole for the buzzer (Fig. 1).

3 Conclusions

This demo is intended to be an example of how to compose rules and apply them to a physical IoT device. In this demo we addressed a few elementary components. The final goal is to build a complete formal language that covers various user needs. Such language could be mapped to existing visual tools that allow users to compose rules, but lack of advanced operators, such as the composition operator. In the future, we will enhance the physical device allowing to perform EUD activities not only through a mobile device, but also manipulating physical devices [8] and allowing remote control from more generic automation platform, such as [9, 10].

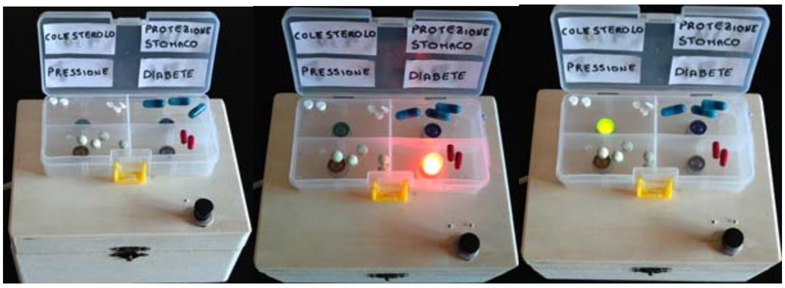

Fig. 1. Pill Dispenser prototype in the status: off, diabetes pills, and pressure pills

References

1. Burnett, M., Scaffidi, C.: End-User Development. In: Soegaard, M., Dam, R.F. (eds.) The Encyclopedia of Human-Computer Interaction, 2nd. The Interaction Design Foundation, Aarhus, Denmark (2013)
2. Scaffidi, C., Shaw, M., Myers, B.: Estimating the Numbers of End Users and End User Programmers. (2005). Proc. VL/HCC 2005. IEEE Computer Society, 207-214.
3. Lieberman, H., Paternò, F., Klann, M., Wulf, V.: End-User Development: An Emerging Paradigm. In: Lieberman, H., Paternò, F., Wulf, V. (eds.) End-User Development, vol. 9, pp. 1-8. Springer, Dordrecht, The Netherlands (2006)
4. Fischer, G., Fogli, D., Piccinno, A.: Revisiting and Broadening the Meta-Design Framework for End-User Development. In: Paternò, F., Wulf, V. (eds.) New Perspectives in End User Development. Springer Int.
5. M. Bartolomeo, Internet of Things: Science fiction or business fact? (2004), Harvard Business Review.
6. Crema, C., Depari, A., Flammini, A., Lavarini, M., Sisinni, E., Vezzoli, A.: A smartphone-enhanced pill-dispenser providing patient identification and in-take recognition. (2015) Proc. MeMeA 2015, pp. 484-489.
7. Yang, G., Xie, L., Mäntysalo, M., Zhou, X., Pang, Z., Xu, L.D., Kao-Walter, S., Chen, Q., Zheng, L.R.: A Health-IoT Platform Based on the Integration of Intelligent Packaging, Unobtrusive Bio-Sensor, and Intelligent Medicine Box. IEEE Transactions on Industrial Informatics 10(4), 2180-2191 (2014)
8. Ardito, C., Buono, P., Costabile, M.F., Lanzilotti, R., Simeone, A.L. Comparing low cost input devices for interacting with 3D Virtual Environments (2009), HSI '09, pp. 292-297.
9. Desolda, G., Ardito, C., Matera, M. (2016). EFESTO: A platform for the end-user development of interactive workspaces for data exploration. Communications in Computer and Information Science, CCIS, vol. 591, pp. 63-81.
10. Desolda, G. (2015). Enhancing workspace composition by exploiting linked open data as a polymorphic data source. Smart Innovation, Systems and Technologies, 40, pp. 97-108.