Recommending Trigger-Action Rules for the IoT

CANDIDATE: Alessia Mantovani
SUPERVISORS: Fulvio Corno, Luigi De Russis and Monge Roffarello Alberto

The Internet of Things (IoT) is the Internet extended into to the physical world. Its function is to collect data from objects and transform it into useful information. These objects can be standard devices, such as smartphones or tablets, but also online services or any kind of Internet-enabled physical objects. For empowering end users to program their IoT devices, the End User Development (EUD) approach has demonstrated to be effective and easily understandable. EUD enables end users (i.e., people who are not professional developers) to customize the functionality of a specific software. This is valuable because end users know their own context and needs better than anybody else. One of the most used paradigms in EUD is Trigger-Action Programming. By employing such a paradigm, user can manually define compose through dedicated interfaces by means of "if-then" constructs, to create event-action associations (i.e., a trigger-action *rule*). An example of a trigger-action rule is "**IF** I go to the cinema **THEN** change my phone profile to Silent". Its elements are:

- Trigger (IF): it is an event that, when detected, causes the execution of an operation;
- Action (THEN): it is an operation that is executed as a consequence of a trigger.

Some works have tried to improve models and interfaces for EUD in the IoT and one of them is EUPont, an ontology that considers the entities of the IoT ecosystem and allows the definition of abstract trigger-action rules that can be automatically adapted to different contextual situations. In this work, EUPont was used for saving data about supported features and services, contextual information, and the trigger-action rules.

Goal

By exploiting Trigger-Action Programming, the goal of the thesis is to design, develop, and evaluate a mobile-based application that suggests and actually executes trigger-action rules based on contextual information and user preferences, with the aim of assisting end users in customizing their smartphone functionalities and online services. For simplicity, this work focus on mobile and web services (i.e., the involved IoT objects were smartphones and online services), thus also allowing an easiest "in-the-wild" evaluation with end users. With this application, the user's commitment regarding the composition of the rules can be reduced, as well as the infinite possibilities of interconnection of the features that the objects of interest can provide to them. In particular, this application can facilitate end users in this scenario, by recommending rules in according to their smartphone usage habits, which they can evaluate and subsequently decide whether to activate or not. While this application is the core of the thesis work, it cannot fully reach the mentioned objective without being part of a more articulated system able to manage, elaborate and store additional information, briefly described hereafter.

Analysis

The first step in this work was to create a list of all the operations that a user can perform with her smartphone, making differentiation between smartphone features (i.e., supported Android functionalities) and online services (e.g., those provided by Facebook, Instagram or Twitter). Some of the most important monitored features are:

- Battery level and device under charge or nots;
- Opening, closing, installing and uninstalling applications;
- Enabling, disabling, connecting and disconnecting Wi-Fi, Bluetooth, mobile data and airplane mode functionalities, by considering the name of the network/Bluetooth device to which connecting or from which disconnecting;
- All events about the screen, such as brightness, rotation, night mode, screen lock/unlock;

- Jack plugged/unplugged, audio/music play/stop, image/video/photo taken;
- Notification profile mode and all the received notifications with the contact from which they come and their content;
- Walking, running, and moving on a vehicle/on a bicycle;
- Places (i.e, when entering into a place and/or when exiting from another one).

Another important category considered is the one which monitors the repetition of all the just mentioned (and others) events over the time (e.g. every day at 10).

To create this list, a thoughtful search on the web was conducted, by considering whether a feature could be a trigger and/or an action for a rule. Since the process that the application uses to intercept those features has to be automatic, and users do not have to manually choose a trigger and pair it with an action, the application has to detect every activity that users perform on their devices and store the related information. Furthermore, the application has to analyze the collected data to create recommendations (if any) consistent with the detected feature.

Design and Implementation

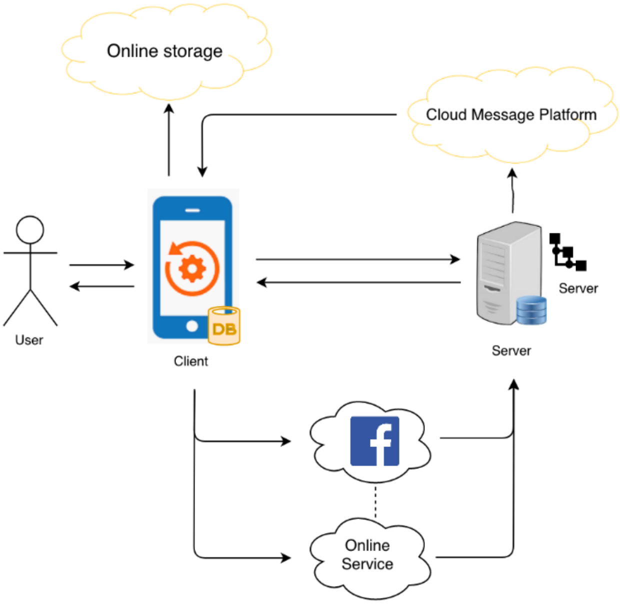

Figure 1: System general architecture.

The mobile application realized in this thesis fully cannot reach its goal without additional information coming from either the cloud or a custom-made server. Therefore, it was inserted in a system that follows a client-server architecture and it is composed by: a server, a mobile application as a client, one or more online services (e.g., Facebook), an online storage, and a cloud messaging platform. As shown in Figure 1, the server, in particular, is a RESTful one, it hosts the EUPont ontology and a remote database while the mobile application is an Android application that hosts an additional local database. After a user performs an operation on her smartphone, the application collects information and when is able to create an association between a Trigger and an Action, it asks the user if the association has to be saved or ignored. In the first case the application sends a request to the server and stores the information in the remote database. In the second case, instead, nothing happens, but information on both saved or ignored rules (and also deleted ones, since the user can eventually delete rules after saving them) are stored on an online database for analysis purpose (i.e., Firebase). Regarding online services, instead, the work has been done for Facebook only (for a first version of the application it has been decided to focus on one service). The creation process of a rule which has a trigger or an action that belongs from Facebook, involves also Firebase cloud messaging platform. Whenever an activity is performed on Facebook, this platform receives requests from the server (through the implementation of a Webhook) and sends a notification to the application. This notification allow the application to collect information about operation on online services.

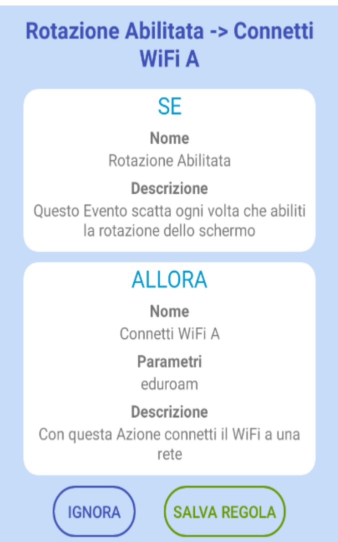

Figure 2: details of a rule

The user interface available in the app allows a user to insert her credentials for logging in the application, to navigate the saved rules, to save new rules and to set preferences, e.g., the places to be monitored. It is very simple because the application largely runs in background. There are four views: login, home (for rule list visualization), details for a selected rule (Figure 2) and settings.

To suggest rules starting from the monitored features, the application uses the following algorithm. Every time a user performs an operation on her device (and this operation is part of the supported features), the application stores the information in the local database and, whenever an event is detected, a search within a fixed time interval is performed. The algorithm stores the correlation with the first event found that does not belong to the same category of the detected one. If the occurrences of this correlation are lower than a specific threshold, nothing happens, otherwise a new rule is suggested to the user. Upon reaching a pre-set threshold (it counts the repetition of the same rule and is different according to the type of events), the user receives a new suggested rule and, according to the perceived utility of this rule, she can ignore it or save it (Figure 2). The application also provides a view in which the user can find previously saved rules, so that she can simply consult the list and decide to disable and/or delete them at any moment.

Evaluation and Results

End users have been involved into an “in-the-wild” evaluation of the application and they have been part of the improvement process. In the final step of the work, thanks to their feedback, it has been possible to refine the application. It has been also possible to explore which rules are mostly created, which rules do not raise interest in users and if they are well disposed towards the usage of this kind of mobile applications. The evaluation involved 6 participants that installed the application for 2 weeks. Before starting the evaluation, participants received an introductive document to explain the objective of the evaluation, an initial questionnaire, the application, and a final questionnaire. From the data collected at the end of the evaluation, the most important considerations which emerged were:

- no suggested rule has been deleted (neither when they were just suggested, nor in a second moment) and around half of the rules were salved and half ignored;
- the number of rules per user are consistent with users’ answers to “smartphone usage level” question (made in the initial questionnaire) for example a user who answered that her phone usage is high has received many rule suggestions; while a user who said not to use smartphone frequently has received few suggestions;
- the type of suggested rules changes in relation to user habits, and there are no equally recurring rules for each user, but Wi-Fi and GPS rules were the most common.

The final opinion of participants was that this application has a great potential. The idea behind this application has been largely appreciated, despite the variety of suggested rules was not very wide because of the choices of supported features and online service. Users were enthusiast to try it and other people that were not enrolled in the study but have heard about it, were interested in possible future releases. The main elicited drawback, instead, is that since the application needs to monitor all activities that users perform on their device, it requires many services running in the background and this causes an intense battery consumption.

Conclusion

In this thesis, a mobile application to suggest rules has been realized and evaluated with end users. From the evaluation, the composition of the rules could be further improved inserting new supported functionalities and/or online services. Starting from the work done during this thesis it will be also possible to integrate the developed application with the RecRules recommender system. This integration will allow to compute more complex suggestions based on trigger-action rules previously activated by the user (content-based information) and activated by other users (collaborative information).