Programming IoT-spaces: A User-Survey on Home Automation Rules

Danny Soares[1], João Pedro Dias[1,2], André Restivo[1,3], and Hugo Sereno Ferreira[1,2]

[1] DEI, Faculty of Engineering, University of Porto, Porto, Portugal
[2] INESC TEC, Porto, Portugal
[3] LIACC, Porto, Portugal
[4] {up201505509,jpmdias,arestivo,hugo.sereno}@fe.up.pt

Abstract. The Internet-of-Things (IoT) has transformed everyday manual tasks into digital and automatable ones, giving way to the birth of several end-user development solutions that attempt to ease the task of configuring and automating IoT systems without requiring prior technical knowledge. While some studies reflect on the automation rules that end-users choose to program into their spaces, they are limited by the number of devices and possible rules that the tool under study supports. There is a lack of systematic research on (1) the automation rules that users wish to configure on their homes, (2) the different ways users state their intents, and (3) the complexity of the rules themselves — without the limitations imposed by specific IoT devices systems and end-user development tools. This paper surveyed twenty participants about home automation rules given a standard house model and device's list, without limiting their creativity and resulting automation complexity. We analyzed and systematized the collected 177 scenarios into seven different interaction categories, representing the most common smart home interactions.

Keywords: Internet-of-Things · Home Automation · Trigger-action programming · End-user Development

1 Introduction

The Internet-of-Things (IoT) has been converting traditionally analog interactions with digital-based ones, opening doors to an increase of automation in IoT-enabled spaces, leveraging their sensing and actuating capabilities.

Smart homes are a primary example of an IoT-space, although its adoption has been slower than expected [28]. The installation of sensors and actuators in houses can improve the lives of the people who live in them in several ways, including providing more comfort and reducing costs. For example, having heaters, fans, or A/C devices monitored and controlled by an IoT system can improve temperature management efficiency, making houses more comfortable for residents and saving energy [30, 28].

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77970-2_39

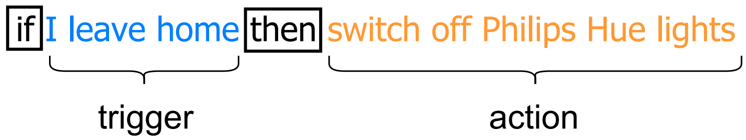

Fig. 1: Trigger-action programming (TAP) example, using the nomenclature typically used in the literature [32]. Using the IFTTT online service nomenclature [17], in a *if A then B* rule, *A* is a trigger, *B* is an action, and the entire rule is an applet.

Automating IoT systems, including smart homes, is not without its challenges, especially when most end-users do not have any technical background [16]. The heterogeneity and a large number of devices, platforms, and services used in IoT, together with the need for end-users to be able to configure and automate them, requires a different approach. While traditional programming (using code editors and integrated development environments) has been the go-to solution for developers and other technical individuals, as the number of IoT application scenarios, environments, and non-technical users increased, it became necessary to build abstractions of sensors, actuators, and whole devices, with additional supporting solutions as a way to reduce the complexity of developing and managing them [3, 16]. This lead to the (re)birth of several low-code programming strategies for end-user development (EUD), which include trigger-action programming (TAP — see Fig. 1) [24], programming by demonstration [20], visual programming [25], and domain-specific languages [15]. Most of the programming solutions that use these strategies also differ on the means of programming, leveraging visual notations [25, 9, 26, 31, 18], natural language processing tools, and voice assistants [1, 19] as a way for users to configure (*viz.*, program) their systems.

While several authors [18, 1] state that these low-code solutions for end-user development still have considerable limitations, they also point out their growth in the variety and number of users. Thus, it becomes of paramount importance to understand what end-users wish to automate, how they state their intents, and grasp into the users' programming mental models. Knowing this can provide valuable information to future research, allowing researchers and industry-alike to model their own systems, and making it more easy find their limitations. This also provides ground for the development of test scenarios, using both testbeds and simulations [10, 8]. As far as we could find, there is a lack on the literature of a systematic study on the concrete rules that users would define for smart home automation given a base set of devices and a minimal but realistic definition of a home (*i.e.*, akin to a house floor plan).

Some studies already reflect on the automation rules that end-users program into their spaces [32, 1, 22]. However, these studies are limited by the number of devices and ways of interaction that the development tool under study supports (which, in most cases, is limited to the IFTTT online service [17] due to the easy

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77970-2_39

access to the applet dataset). Earlier works [6, 4] attempted to survey automation rules and their complexity, but due to the continuous and rapid evolution of the IoT ecosystem, they fall short to represent the current spectrum of automation possibilities.

We surveyed 20 participants for home automation rules given a standard house model and devices in this work. The study intended to gather as many and as varied home automation scenarios as possible from individuals with different backgrounds and technical know-how while maintaining a certain level of similarity with real-world scenarios and not limiting their creativity and resulting automation complexity.

We proceeded to split the gathered scenarios into categories according to similarities in their structure and types of conditional statements. This survey also added knowledge on how users typically describe their home automation scenarios using text, allowing us to understand if different individuals use different phrases to describe the same scenarios.

The remainder of this paper is structured as follows: Section 2 presents an overview of the relevant literature. Section 3 presents the devised model of a smart home, the methodology of our survey, and its results, which are further analyzed and discussed in Section 5. Finally, Section 6 presents an overview of our study's threats, and Section 7 gives some final remarks.

2 Related Work

Dey *et al.* [6], circa 2006, gathered 371 automation scenario descriptions from 20 participants. Almost all of the participants (95%) stated their automation rules in a *if-then* fashion, and around 23.5% of the rules used explicit Boolean logic (*e.g.*, use of `and` or `or` statements). They also categorize the rules accordingly to their complexity: 78.6% fell into the *simple if-then* category, 7% mentioned temporal constraints, 7% mentioned spatial constraints, 6.5% mentioned personal relationships, and, less than 1% focused on environmental personalization (which depends on knowing the inhabitants preferences). Around 14% of the rules mentioned some kind of pre-defined user preference (*e.g.*, preferred ambient temperature).

Brush *et al.* [4], circa 2010, conducted an *in-situ* study of the home automation, by doing semi-structured home visits to 14 households and carried both an analyse of the in-place system and several interviews with the inhabitants. They found out that most of the automation's were of two levels of automation: *user controlled* — where the household explicitly performs an action that triggers one or more actuators — and *rule-based* — where actions happen based on events or at certain times (TAP-like).

Ur *et al.* [32] carried three studies to understand how users use TAP on smart home scenarios. In the first study, they asked 318 workers on Amazon's Mechanical Turk (MTurk) to list five things that they would want a smart home system to do, concluding that most of them fit into four categories, namely: (1) programming, *e.g.*, "automatically turning on the lights when it is dark

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77970-2_39

outside"; (2) self-regulation, *e.g.*, "adjust the house to my preferred temperature at all times"; (3) remote control, "hitting a button on my phone to turn on the lights"; and (4) specialized functionality, *e.g.*, "a breakfast-making machine". To further check the ability to model the workers' intents, they tried to fit them into the TAP model, finding that 62.6% of the submitted answers fit the model. In a second study [32], the authors downloaded a dataset of 67169 recipes (TAP rules) from IFTTT [17], focusing only on the recipes related to smart home automation, which corresponded to a total of 1107 (2,1%), concluding that 513 recipes (0.8%) use physical devices as triggers and 594 (1.3%) use physical devices as actions. A third-study required that a sample of 226 MTurk workers complete pre-defined automation tasks using IFTTT, concluding that 80% or more of the participants successfully implement the presented automation cases. However, the authors' study on the diversity and complexity of the rules that end-users want to configure is too open and vague. There is no common base of devices to automate nor sample building schematic. Also, there is no dataset provided with the first study's collected cases (which is very relevant for this work).

Ammari *et al.* [1] study on how people use voice assistants, which included 82 Amazon Alexa and 88 Google Home users, concluded that IoT-related commands are one of the three most uses of these assistants. They conclude that 85% of the Amazon Alexa and Google Assistant IoT commands involved switching lights on and off. In the case of Alexa, about 10% involved adjustments in light's color and temperature, and a small percentage involved adjusting the temperature of different parts of the house. For Google Assistant, around 10% of the queries were related to changing light colors, dimming lights, and changing fan speeds. By their work, only five study participants created trigger-action rules using Alexa, and the authors do not discuss the intricacies of these rules.

Mi *et al.* [22] also carried a survey on IFTTT, including an analysis of over 408 services (third-party services such as Amazon Alexa), 1490 triggers, 957 actions, 320000 applets. Although their work is not IoT-focused, they carry an analysis on the IoT-related subset of the dataset. In their study, they conclude that the majority of the entries by users are trigger-action ones (*e.g.*, "turn on the light"); thus, the resulting applets are, in their majority, relatively simple, mostly due to the limited and simple interfaces exposed by most IoT devices. The authors also add that this is due to "the fact that most tasks (in the smart home context) we want to automate are indeed simple". While we agree that the limitations posed by the devices limit what end-users can program them to do and that the majority of the rules are indeed simple by nature, as the number of inhabitants and devices increases, the resulting operational context can be complex to model and reason about [21].

3 Survey

A survey was envisioned as the most effective way to gather as many automation scenarios as possible in a timely fashion. The methodology was based on the one presented by Molléri et al. [23].

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77970-2_39

3.1 Smart Home Model

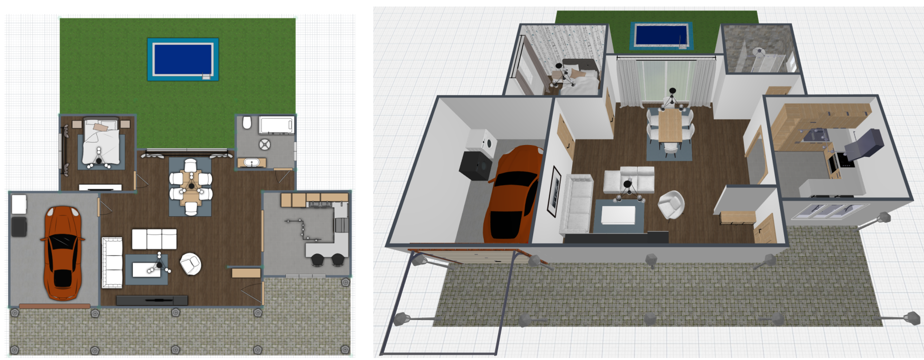

Fig. 2: 2D and 3D floor plan of the smart house used for the survey.

To have a pre-defined, common foundation from where the participants could base themselves to draft their own automation scenarios, we designed a house floor plan and 2D/3D models of it, as shown in Figure 2. The house has a total of 8 *spaces*: (a) a *garage*, (b) a *front patio*, (c) a *pool*, (d) a *garden*, (e) a *living room*, (f) a *kitchen*, (g) one *bedroom*, and (h) a *bathroom*.

Along with the home model, we provide a list of smart devices containing various types of sensors and actuators for the participants to use. Namely, across all home divisions, there are the following IoT devices: (1) motion, temperature, humidity, smoke, and air quality sensors, (2) security cameras, (3) controllable lights, (4) controllable windows and blinds, (5) A/C system, (6) robot vacuum cleaner, and (7) sound system.

The (a) *garage* has (a.1) controllable outside and inside doors, (a.2) washing machine, and (a.3) a dryer machine. The (b) *front patio* has only a (b.1) controllable entry door. The (c) *pool* has a (c.1) automated pool cover, (c.2) cleaning system, (c.3) water temperature sensor, and (c.4) water heating system. The (d) *garden* has a (d.1) water sprinkler system, (d.2) soil moisture sensor, and (d.3) robot lawnmower. The (e) *living room* has a (e.1) smart TV. The (f) kitchen has a (f.1) stove, (f.2) oven, (f.3) exhaust hood, (f.4) dishwasher, and (f.5) coffee machine. The (g) *bedroom* has (g.1) a smart TV, and (g.2) controllable bedside lamps. Lastly, the (h) *bathroom* has a (h.1) heated towel rack.

We also allowed and instigated participants to include other devices in their home automation scenarios as long they were available as off-the-shelf IoT solutions. No limitations on the interoperability of the IoT system parts nor in the end-user programming interface were defined nor presented.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77970-2_39

3.2 Methodology

An online form[5] was picked as a data collection method, given the study's motivation to gather as many automation scenarios as possible while attempting to reducing any bias on the respondent population. The form presented the smart home model, namely, the house floor plan and the list of available devices. Users had only one question where they could insert as many scenarios as they wished to, without any limitation in size or form.

The survey was then disseminated among 20 participants from different educational fields and ages. All the answers were collected in a spreadsheet, anonymized, and the individual scenarios identified, allowing further analysis.

4 Results

The survey resulted in 177 scenarios grouped into categories according to their structure and format similarity. This allowed us to filter duplicated entries, keeping only the mostly-unique and representative ones. The list of categories and a brief description and a sample of three (when available) representative scenarios extracted from the dataset are given in the following paragraphs.

Sensors and actuators Scenarios that only use sensors and actuators, where the sensors trigger the actuators, *e.g.*:
- "When there is movement in the garage, turn on the garage lights";
- "Adjust pool water temperature according to the outside temperature if someone is using it";
- "When smoke detectors are activated, alert through the sound system, alert the house owner via SMS, and warn the fire department".

Actuators on schedule Scenarios where actuators are triggered on a fixed schedule, *e.g.*:
- "When time is 7:30 am, turn on the coffee machine, the hot water system, and the kitchen lights";
- "Every Saturday at 3 pm, turn on the robot vacuum cleaner, and the robot lawn mower";
- "When time is 7:30 am, turn on the coffee machine, the hot water system and the kitchen lights (if they were turned off)".

Actuators on time interval, with sensors Scenarios that combine sensors information and time intervals to trigger actuators, *e.g.*:
- "During the night, when there is motion in one room, light that room at 200 brightness";
- "When a TV series starts, and I am home, turn on the TV on that channel and prepare popcorn. If I am not home, record it";
- "When turning on the dishwasher (or washing machine) after 20h, wait for 00h to start".

Sensors with timers Scenarios that combine sensors information and time intervals to trigger actuators, *e.g.*:

[5] Google Forms, https://forms.google.com.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77970-2_39

- "During the night, when there is motion in one room, light that room at 200 brightness";
- " When a TV series starts, and I am home, turn on the TV on that channel and prepare popcorn. If I am not home, record it";
- "When turning on the dishwasher (or washing machine) after 20h, wait for 00h to start".

Actuators with timers Scenarios where the actuators are triggered when the status of the sensor does not change for some time, *e.g.*:
- "When it is 23:00, turn on the garden watering system for 10 minutes";

External services Scenarios that depend on external services to trigger the actuators, *e.g.*:
- "When the sun is expected during the following hours, turn off the heating system";
- "If it will rain in the next hour, inform that drying outside is not the best plan";
- "With a solar panel; and weather information; schedule a machine to run sometime during the day (*e.g.*, washing machine). It is expected that the system automatically schedules it to the time of day that is most likely to be sunny for power saving."

One-time actions Scenarios that are meant to happen only once, instead of being recurring, *e.g.*:
- "Shut all unnecessary devices";
- "When it is 7:00 today, turn on the bedroom lights".

The full dataset and the model house floor plan and device list are available on Zenodo [29] to ease the study's replication and allow further analysis.

5 Analysis

In this section, an analysis of the submitted 177 scenarios is done, attempting to extract insights from the different rules submitted given the suggested smart home model. We considered all the submitted entries valid smart home automation scenarios, and we were able to categorize all the submitted scenarios in one of the seven defined categories. The scenarios differ (1) in the granularity of application (*e.g.*, with some of them being specific to a house part or domestic appliance), (2) in complexity (rules range from direct triggers to one device to triggers of multiple devices depending on several conditional statements) and (3) in writing fashion (with most of them being close to a conditional programming logic).

Table 1 details the distribution of scenarios through the categories, showing the absolute and relative frequency of submissions per category as identified in Section 4.

Responses such as *"Intensity of lights based on the available natural light"*, *"Blinds inclination system based on outside light"*, *"On schedule turn on the coffee machine"* would need modifications to be closer to a programming-like

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77970-2_39

Table 1: Categories of submitted scenarios, along with the absolute and relative frequency of submissions for each category.

Category	**Absolute Frequency**	**Relative Frequency**
Sensors and actuators	103	0.58
Actuators on schedule	42	0.24
Actuators on time interval, with sensors	15	0.08
Sensors with timers	9	0.05
External services	5	0.03
One-time actions	2	0.01
Actuators with timers	1	0.01
Total	**177**	**1.00**

format to be possible to implement them in usual end-user programming solutions. For example, these should look more like *"When the luminosity in the living room is below $value, then increase the light's intensity by $increment"*, or *"When time is 7:00, then turn on the coffee machine"*. These responses (and respective scenarios) are still valid because the information portrayed is enough to understand their meaning and to which category they belong.

With the information collected, the house plan, and devices provided to the participants, we created a **resulting ecosystem** that represents the house with all the devices that the participants used. This ecosystem is represented in Figure 3, showing the house plan with all the devices used by the participants. Some respondents also mentioned using an external weather forecast API and wearables (which are not represented in the isometric visualization).

The most common way of specifying scenarios is by using the structure "when *condition*, then *action*" or "*action* when *condition*" (with the recurrent use of "if" instead of "when"). This is close to the representation commonly used by TAP approaches. However, some scenarios are depicted differently, mainly for scheduled actions, such as "*action* at *time*", or "everyday at *time*, *action*".

Looking at the dataset, we can see that there are home areas more frequently identified. In contrast, others are almost unseen, as depicted in the chart of Fig. 4. We consider direct mentions all the mentions to specific rooms in the submitted scenarios, *e.g.*, garage or bedroom. All the indirect mentions consist of remarks about certain things that are, typically, only present in certain rooms, namely: washing machine mentions are part of the garage (by the given device list); entrance is considered front patio; lawnmower references considered as part of the garden; kitchen includes mentions to the dishwasher, coffee, and oven; alarm clock, waking up, sleep are all related to the bedroom; and all mentions to shower and toilet are considered part of the bathroom.

Users also tend to specify similar (or equal) scenarios using different expressions, granularity and forms. This was expected since the participants' background (*e.g.*, educational level, previous experience with IoT, age, and the way of expressing ideas) was homogeneous. As an example, "Turn the heating system

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77970-2_39

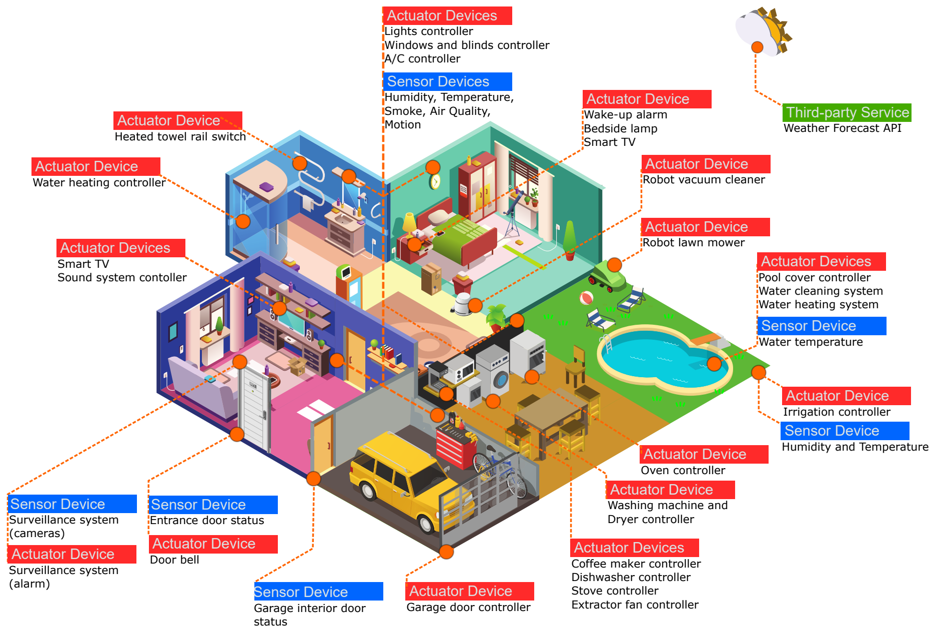

Fig. 3: Isometric visualization of the resulting ecosystem based on the survey responses.

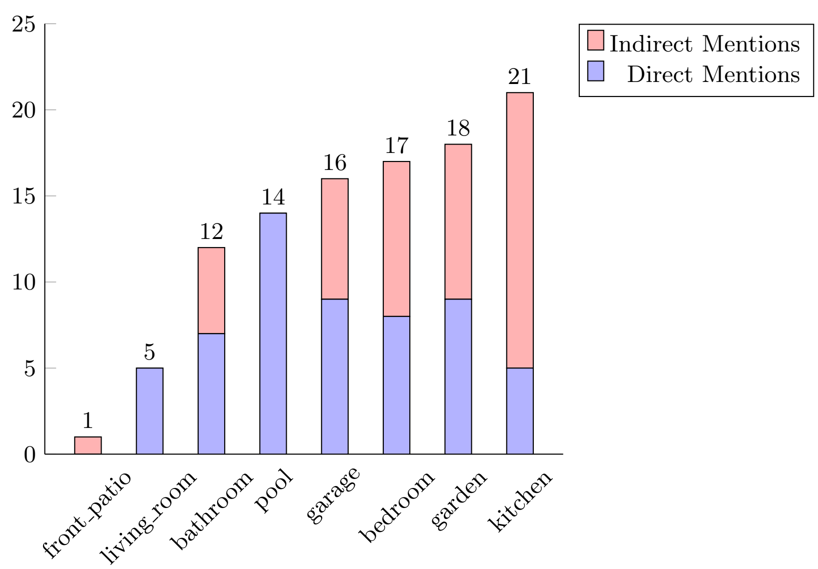

Fig. 4: The number of mentions to specific home parts in the submitted scenarios. Direct mentions consider situations where the surveyed participants directly mentioned a given house part and the indirect mentions consider references to certain things that are, typically, only present in certain rooms.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77970-2_39

on, set to the preferred temperature, on schedule.", "Maintain room temperature in between a specified permissible range.", and "Turn on the AC when the temperature is higher than a given value." transmit the same rule — adjusting the house temperature according to a preference value — in different fashions, either in format or precision.

Approx. 28% of the scenarios mention "turn on" actions, and there are 28 direct mentions to *lights*, 23 to *water*, 21 to *blinds*, and 20 to *temperature*.

It is noted the use of pre-conditionals in some scenarios, more specifically, defining some condition that should be met before enforcing the rule, *e.g.*, "With a solar thermal collector (for heating water); when the sun is expected during the following hours, turn off traditional water heating system". The use of macros that aggregate a set of tasks and sub-rules is also visible, *e.g.*, "*Holiday mode*: when any device is used/triggered notify the owner" and "*Garden automation*: stable soil moisture level, temperature stabilization in adverse weather".

Although integration with external services is only mentioned once, there are several rules that, when implemented, would depend on some information provider. For example, we can consider "If the hot water system is based on electricity, heat when electricity is cheaper" would depend on knowing the market prices of electricity. Further, voice control, *e.g.*, "Voice control over coffee machine/blinds/lights/etc.", is typically accomplished by integrating with a third-party voice assistant such as Amazon Alexa, Apple Siri, or Google Assistant [1].

Finally, it is also noticeable that some rules are too generic, *e.g.*, "Shut down all unnecessary devices", which would require that the IoT system had some degree of contextual awareness to be able to execute them. There is also some concern about the failure of the system parts and the use of IoT to detect them, *e.g.*, "Send SMS alert if faulty freezer/fridge".

6 Threats to Validity

For this survey, we have identified some threats that may affect the validity of the results attained.

We asked participants for home automation scenarios and did not give them any **structure for the phrases**, to understand how they would write the scenarios, which resulted in many scenarios being just a brief description and not specific enough. Perhaps, having requested the participants to provide more detailed scenarios would have resulted in more concrete scenarios. However, constraining participants to use a specific format or having certain degrees of detail for the scenario descriptions would not have allowed us to evaluate whether users tend to follow a pattern or typical structure.

Although we let participants use any off-the-shelf device, we provided them with an initial list of devices to pick from. This, together with the specified house plant, might have introduced a bias into the chosen scenarios.

The **sample size** for this survey was not very large since we were only capable of gathering 20 participants. Having a larger sample could have resulted in more varied scenarios. This could enrich our analysis and provide more insights into

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77970-2_39

the typical way that users express their automation rules and what these rules typically consist of.

The **level of expertise** of the participants could impact the scenarios provided by them. For example, participants with more experience with IoT should provide more complex and realistic scenarios than participants with no experience in that field. To tackle this threat, we attempted to choose participants with different levels of expertise with low-coding programming solutions and IoT. This resulted in having scenarios from participants whose experience ranged from never having thought of a home automation scenario to participants that had already implemented IoT systems and worked with Node-RED extensively.

Even though the participants had different levels of expertise in home automation, and some even had a lot of experience, the results show **little variety in categories** for the scenarios. After collecting 177 scenarios, we only identified seven categories, and 82% of the submissions belonged to two categories (*i.e.*, 58% to Sensors and Actuators category and 24% to Actuators on schedule category). Increasing the number of participants in the survey would probably, have resulted in more varied scenarios and more categories.

7 Conclusions

This paper presented a survey conducted with 20 participants to collect home automation scenarios, which resulted in 177 scenarios to be categorized and analyzed. We consider that all the scenarios fit in one of the seven defined categories, representing different types of automation.

The most common pattern used by users to define their automation scenarios shares a similar structure close to conditional programming — "when *condition*, then *action*", or "*action*, when *condition*" — which is compatible with the trigger-action programming model used by several market solutions including IFTTT. This shows that it is intuitive for regular users to describe home automation scenarios in a mostly-structured fashion, easily transposed to a conditional programming fashion. Besides, the users tend to use (or mention) macros and/or aliases that represent more than one device (*e.g.*, a group of lights) or more than one action (*e.g.*, garden control).

Taking into account the available solutions in the market for end-user programming and their programming strategies, we can consider — taking into account relevant literature such as the one presented in Section 1 — that while most of the scenarios could be easily mapped into TAP rules, the rules that do not follow such model appear as a challenge which is mostly ignored by existing solutions, especially the ones that focus users with little to none technical knowledge. In this case, voice assistants can become of utmost importance, allowing users to create automation rules in a conversation, adding complexity by steps instead of specifying everything in one statement or by a diagram [1, 13, 19].

Contextual awareness also adds value to low-coding solutions [16], since with such contextualization, they can use information about the system and their parts, the system surroundings, and the current defined behaviors and rules to

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77970-2_39

provide user information, insights, and alerts when new rules (or changes) create conflicts with already defined rules, can lead to malfunctions, or nefarious effects — *e.g.*, avoiding turning off the CO2 sensors by mistake by alerting the user to the changes that will occur.

Finally, as the number of devices being Internet-connected increases rapidly, depending on end-users to control and manage all of them appears to be too complex of a challenge. Strategies that hinder this complexity by the end-users seem to be the *natural direction* for IoT systems by enabling devices to manage themselves in terms of software and hardware configurations [2], optimizations in resource utilization (*e.g.*, energy) [27] and usage requirements (continuous adaptation by learning from the environment) [5], and in prevention and recovery of failures or other issues [14, 12, 11].

The data collected in this study[6], including the definition of typical automation categories, can be used as a foundation for follow-up studies, ranging from human-computer interaction (*e.g.*, low-code and end-user programming) to IoT systems design and implementation (*e.g.*, what are the current sensing and actuating needs that are not meet by the existent systems which are of the end-user interest). This data allows the definition of system prototypes by researcher and industry communities to meet current user needs, and be validated against a known dataset of user-based interaction and automation scenarios. In previous work, we already used this data partially for defining research directions and build validation testbeds and scenarios (*e.g.*, definition of a *SmartLab*[7]) [14, 19].

Acknowledgement. This work is financed by National Funds through the Portuguese funding agency, FCT - Fundação para a Ciência e a Tecnologia, within project UIDB/50014/2020.

References

1. Ammari, T., Kaye, J., Tsai, J.Y., Bentley, F.: Music, search, and iot: How people (really) use voice assistants. ACM Transactions in Computer-Human Interaction **26**(3) (Apr 2019)
2. Athreya, A.P., DeBruhl, B., Tague, P.: Designing for self-configuration and self-adaptation in the Internet of Things. Proceedings of the 9th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing, COLLABORATECOM 2013 pp. 585–592 (2013). https://doi.org/10.4108/icst.collaboratecom.2013.254091
3. Baresi, L., Ghezzi, C.: The disappearing boundary between development-time and run-time. In: Proceedings of the FSE/SDP Workshop on Future of Software Engineering Research. p. 17–22. FoSER '10, Association for Computing Machinery, New York, NY, USA (2010)
4. Brush, A.B., Lee, B., Mahajan, R., Agarwal, S., Saroiu, S., Dixon, C.: Home automation in the wild: challenges and opportunities. In: proceedings of the SIGCHI Conference on Human Factors in Computing Systems. pp. 2115–2124 (2011)

[6] The replication package of the user study can be found on Zenodo [29].
[7] The replication package for the *SmartLab* can be found on Zenodo [7].

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77970-2_39

5. Chatzigiannakis, I., Hasemann, H., Karnstedt, M., Kleine, O., Kröller, A., Leggieri, M., Pfisterer, D., Römer, K., Truong, C.: True self-configuration for the IoT. Proceedings of 2012 International Conference on the Internet of Things, IOT 2012 pp. 9–15 (2012). https://doi.org/10.1109/IOT.2012.6402298
6. Dey, A.K., Sohn, T., Streng, S., Kodama, J.: icap: Interactive prototyping of context-aware applications. In: Fishkin, K.P., Schiele, B., Nixon, P., Quigley, A. (eds.) Pervasive Computing. pp. 254–271. Springer Berlin Heidelberg, Berlin, Heidelberg (2006)
7. Dias, J.P.: jpdias/smartlab: Replication package for smartlab (Apr 2021). https://doi.org/10.5281/zenodo.4657647
8. Dias, J.P., Couto, F., Paiva, A.C.R., Ferreira, H.S.: A brief overview of existing tools for testing the internet-of-things. In: 2018 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW). pp. 104–109 (Apr 2018). https://doi.org/10.1109/ICSTW.2018.00035
9. Dias, J.P., Faria, J.P., Ferreira, H.S.: A reactive and model-based approach for developing internet-of-things systems. In: 2018 11th International Conference on the Quality of Information and Communications Technology (QUATIC). pp. 276–281 (Sep 2018). https://doi.org/10.1109/QUATIC.2018.00049
10. Dias, J.P., Ferreira, H.S., Sousa, T.B.: Testing and deployment patterns for the internet-of-things. In: Proceedings of the 24th European Conference on Pattern Languages of Programs. EuroPLoP '19, Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3361149.3361165
11. Dias, J.P., Restivo, A., Ferreira, H.S.: Empowering visual internet-of-things mashups with self-healing capabilities. In: 2021 IEEE/ACM 2nd International Workshop on Software Engineering Research Practices for the Internet of Things (SERP4IoT) (2021)
12. Dias, J.P., Sousa, T.B., Restivo, A., Ferreira, H.S.: A pattern-language for self-healing internet-of-things systems. In: Proceedings of the 25th European Conference on Pattern Languages of Programs. EuroPLop '20, Association for Computing Machinery, New York, NY, USA (2020). https://doi.org/10.1145/3361149.3361165
13. Dias, J.P., Lago, A., Ferreira, H.S.: Conversational interface for managing non-trivial internet-of-things systems. In: Proceedings of the 20th International Conference on Computational Science. pp. 27–36. Springer (2020)
14. Dias, J.P., Lima, B., Faria, J.P., Restivo, A., Ferreira, H.S.: Visual self-healing modelling for reliable internet-of-things systems. In: Proceedings of the 20th International Conference on Computational Science (ICCS). pp. 27–36. Springer (2020). https://doi.org/10.1007/978-3-030-50426-7_27
15. Einarsson, A.F., Patreksson, P., Hamdaqa, M., Hamou-Lhadj, A.: Smarthomeml: Towards a domain-specific modeling language for creating smart home applications. In: 2017 IEEE International Congress on Internet of Things (ICIOT). pp. 82–88 (June 2017)
16. Ghiani, G., Manca, M., Paternò, F., Santoro, C.: Personalization of Context-Dependent Applications Through Trigger-Action Rules. ACM Transactions on Computer-Human Interaction **24**(2), 1–33 (may 2017). https://doi.org/10.1145/3057861
17. "IFTTT": Ifttt helps your apps and devices work together (2019), https://ifttt.com
18. Ihirwe, F., Di Ruscio, D., Mazzini, S., Pierini, P., Pierantonio, A.: Low-code engineering for internet of things: A state of research. In: Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings. MODELS '20, Association for Computing Machinery, New York, NY, USA (2020)

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77970-2_39

19. Lago, A.S., Dias, J.P., Ferreira, H.S.: Managing non-trivial internet-of-things systems with conversational assistants: A prototype and a feasibility experiment. Journal of Computational Science **51**, 101324 (2021)
20. Li, T.J.J., Li, Y., Chen, F., Myers, B.A.: Programming IoT Devices by Demonstration Using Mobile Apps. In: Barbosa, S., Markopoulos, P., Paternò, F., Stumpf, S., Valtolina, S. (eds.) End-User Development. pp. 3–17. Springer International Publishing, Cham (2017)
21. Manca, M., Fabio, Paternò, Santoro, C., Corcella, L.: Supporting end-user debugging of trigger-action rules for iot applications. International Journal of Human-Computer Studies **123**, 56–69 (2019)
22. Mi, X., Qian, F., Zhang, Y., Wang, X.: An empirical characterization of ifttt: ecosystem, usage, and performance. In: Proceedings of the 2017 Internet Measurement Conference. pp. 398–404 (2017)
23. Molléri, J.S., Petersen, K., Mendes, E.: An empirically evaluated checklist for surveys in software engineering. Information and Software Technology **119**, 106240 (2020)
24. Rahmati, A., Fernandes, E., Jung, J., Prakash, A.: Ifttt vs. zapier: A comparative study of trigger-action programming frameworks. ArXiv **abs/1709.02788** (2017)
25. Ray, P.P.: A Survey on Visual Programming Languages in Internet of Things. Scientific Programming **2017**, 1–6 (2017)
26. Reiss, S.P.: Iot end user programming models. In: 2019 IEEE/ACM 1st International Workshop on Software Engineering Research & Practices for the Internet of Things (SERP4IoT). pp. 1–8. IEEE (2019)
27. Seo, J., Kim, W.H., Baek, W., Nam, B., Noh, S.H.: Optimally Self-Healing IoT Choreographies. International Conference on Architectural Support for Programming Languages and Operating Systems - ASPLOS **Part F1271**(1), 91–104 (2017)
28. Shin, J., Park, Y., Lee, D.: Who will be smart home users? an analysis of adoption and diffusion of smart homes. Technological Forecasting and Social Change **134**, 246–253 (2018)
29. Soares, D., Dias, J.P., Restivo, A., Ferreira, H.S.: Smart home automation survey (Feb 2021). https://doi.org/10.5281/zenodo.4531395
30. Sundmaeker, H., Guillemin, P., Friess, P., Woelfflé, S., for the Information Society, E.C.D.G., Media: Vision and Challenges for Realising the Internet of Things. Publications Office of the European Union (2010)
31. Torres, D., Dias, J.P., Restivo, A., Ferreira, H.S.: Real-time feedback in node-red for iot development: An empirical study. In: 2020 IEEE/ACM 24th International Symposium on Distributed Simulation and Real Time Applications (DS-RT). pp. 1–8 (2020)
32. Ur, B., McManus, E., Pak Yong Ho, M., Littman, M.L.: Practical trigger-action programming in the smart home. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. p. 803–812. CHI '14, Association for Computing Machinery, New York, NY, USA (2014)

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77970-2_39