Enhancing Sustainability in Smart Home Management with Automation Simulations and Green Suggestions

Davide Guizzardi
Barbara Rita Barricelli
Daniela Fogli
{davide.guizzardi,barbara.barricelli,danielafogli}@unibs.it
University of Brescia
Brescia, Italy

Abstract

This paper presents a digital twin for green smart homes that can simulate the impact of user-created automations on energy consumption. Data collected about the states of appliances and estimated power requirements of automations are exploited by the digital twin to estimate the overall energy consumption of the smart home. Consequently, the digital twin can alert users about potential energy overloads and provide suggestions for sustainable actions.

CCS Concepts

• **Human-centered computing** → **Empirical studies in interaction design**; **Ambient intelligence**.

Keywords

Smart home, Digital twin, Energy management, Automation

ACM Reference Format:

Davide Guizzardi, Barbara Rita Barricelli, and Daniela Fogli. 2025. Enhancing Sustainability in Smart Home Management with Automation Simulations and Green Suggestions. In *30th International Conference on Intelligent User Interfaces Companion (IUI Companion '25), March 24–27, 2025, Cagliari, Italy.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3708557.3716345

1 Introduction

A smart home is a home-like environment endowed with ambient intelligence and automatic control, which allows the home to respond to the behavior of its occupants and provide them with several facilities [10]. A common approach to building smart homes is to create an Internet of Things (IoT) ecosystem [3] that allows residents to tailor home behavior by creating *automations*. Automations are usually conceived as event-condition-action (ECA) rules that users may compose exploiting end-user development techniques implemented as graphic interfaces [4, 13, 22], conversational tools [7, 8, 12, 15], or multi-modal applications [1]. In particular, automation creation can contribute to smart home energy management, helping inhabitants reduce energy consumption and, consequently, their environmental impact and energy bills. However, when several automations are defined to manage smart homes, conflicts may occur during their execution, especially when different users are in charge of automation creation to control shared appliances. Several approaches have been proposed in the literature to detect conflicts between ECA rules [6, 18] and help users solve them [5, 16, 17, 20, 21]. In this paper, we focus on the interaction between automations that may yield other types of conflicts and issues in smart homes related to sustainable energy consumption. Based on the energy consumption estimation of each available appliance, our approach allows the simulation of the impact that the automation under creation could have on the overall energy consumption of the house. Such a simulation is used to notify the user of critical situations (e.g., the quantity of energy required in a given time interval exceeds the maximum load capacity) or provide the user with suggestions for more sustainable behaviors (e.g., by changing the start time of the automation). To implement this approach, a Digital Twin (DT) of a smart home has been developed, which can monitor and estimate appliances' energy consumption based on data collected about their activation. Using a DT allows for real-time synchronization of data from the actual home and performing 'what-if' analyses of energy consumption whenever the user defines a new automation. This idea was originally presented in a preliminary work [2], while the DT architecture was delineated in [11]. Then, in [9], we presented the machine learning algorithms adopted to estimate the consumption of appliances available in a hypothetical smart home using public datasets. The present paper extends the previous work by considering a real smart home and describing how the simulation of user-created automations has been developed.

2 Background

In our previous work [9], we explored the use of clustering techniques to extract the operation modes of appliances from publicly available smart home datasets. In particular, we built a machine learning pipeline that was able to take energy consumption data from GREEND [19] and UK-DALE [14] datasets, convert those time series into a latent state that summarized the entire activation, and group each latent state hence identifying the appliances' modes of operation. In this way, we have created a table that specifies the average power demand and duration for each category of device and each mode of operation of said category. The complete table is available in [9]; as an example, the appliances belonging to the *Television* category can be operated in two modes, *Stand-by* or *On*, with power consumption respectively of 18 Watt and 110 Watt and with an infinite duration for the first mode and an estimated duration of 1833 seconds for the second one.

IUI Companion '25, March 24–27, 2025, Cagliari, Italy

ACM ISBN 979-8-4007-1409-2/25/03
https://doi.org/10.1145/3708557.3716345

These data could be used to estimate the power requirement and the overall duration of the actions within an automation. For each action, it is possible to identify the device involved and the state the action wants to set. Given the tuple <*device type, state*>, the next step is to look for that combination in the table obtained from GREEND and UK-DALE and retrieve the average consumption and duration. At the end of the process, an estimate of the impact of the automation, both in terms of instantaneous power demand and energy consumption, will be available.

Although good in theory and useful for solving the cold start problem, this approach does not consider the inhabitants' lifestyles and the characteristics of their smart devices. Different people use their devices in unique ways, and smart devices often have varying consumption patterns depending on their maker, model, and usage context; hence, the estimation based on public datasets could heavily differ from the actual scenario. To solve this problem, we propose a system that evaluates the impact of automations using real usage data collected from the house where the system is installed. The platform chosen to manage automations, control devices, and monitor state changes is Home Assistant[1]. Home Assistant (HA) is an open-source home automation platform that runs on local hardware and lets users control the devices connected to the network. HA provides an automation engine and constantly keeps track of any changes in the devices' states.

3 Simulation of automations' impact on energy consumption

Periodically, the system queries HA's API to have the history of the devices. For each device, HA will return the list of all the state changes that happened in the time range requested, specifying, for each record, the new state assumed and the timestamp of the change. Since HA does not provide data with a constant granularity, leading to additional challenges to the estimation process, a function that converts HA's history into a list of fixed duration blocks (one minute each) was first implemented. The converted history can then be analyzed to compute the average power demand and duration of the appliances' modes. The category to which a device belongs influences how the power value is obtained: some devices, such as smart plugs, can directly provide their real-time power usage, and in that case, the average power of a state is obtained by summing the power values and dividing the result by the duration of the state (in minutes). Other devices, like smart speakers, do not report their real-time consumption. For this category of devices, a manual mapping is needed where each state of the device is associated with a static power value, usually obtained from the device's user manual. The average duration of each state is obtained directly by computing the overall time of the state and dividing it by the number of times that device was set to that particular state. Once the usage data of a device is computed, the method compares it with existing data in the database. If the database already contains information for a specific device and state, new moving averages for power and duration are defined using Formula (1). If no prior data exists, the newly calculated averages are inserted into the database without any further operations.

$$\text{new average} = \frac{(\text{old average} * \#\text{past samples}) + \sum \text{new values}}{\#\text{ old samples} + \#\text{new samples}} \tag{1}$$

Given the house-specific information saved in the database, it is possible to estimate the impact of new automations on energy consumption more accurately. In particular, two objectives can be achieved: 1) to evaluate whether different automations affect each other and if potential conflicts arise from their coexistence, particularly regarding the maximum amount of energy the home's electrical system can handle; 2) to provide users with suggestions on how to optimize their automations to use less energy and save money.

Listing 1 presents the pseudocode of the function used to detect potential sustainability conflicts. The function takes the automation under creation as input, combines it with the saved automations, and generates a structure that models the state of the devices throughout each day of the week, known as State Matrix. The State Matrix is a $7 \times 1440 \times$ #devices data structure that captures the state of each device for every minute of the week. It is built by initializing an empty matrix and filling it out using saved automations and historical usage data. For each automation, processed in chronological order, the system determines the days the automation applies to and the devices that are subject to changes. According to the devices' desired states, it extracts from the database the duration of each action and updates the matrix cells for the affected minutes, days, and devices with the states defined by the automation's actions. Based on the State Matrix, the system identifies conflicts, i.e. any time periods when the power consumption exceeds a predefined threshold (typically 3 kW for domestic use in Italian households).

To suggest alternative scheduling for an automation, with the aim to optimize energy costs, a further function computes the original cost of an automation and explores alternative start times that could reduce it. The final output of the function is a list of suggestions for possible time optimization that could save the user some money without raising conflicts with the currently saved automations.

4 A smart home digital twin for automation simulations

The presented functions are implemented in a web-based application representing a Digital Twin of a green smart home. Both the DT and the HA Operating System (OS) run on a Raspberry Pi 5 computer connected to the local network.

The system front-end, written in React, shows automations' details and provides the users with information about the sustainability conflicts and suggestions on how to optimize the selected automation. For example, in Figure 1, the list of all the automations saved in HA is presented on the left. By clicking on the name of an automation, a card with its details will appear on the right. Among the details, the 'Green suggestions' section presents ways to improve the sustainability of the automation.

The system back-end is composed of a set of API endpoints, written in Python, that furnish the front-end with the data it needs. It is responsible for the retrieval of automations' details from HA and

[1] https://www.home-assistant.io

Listing 1: Identify Sustainability Conflicts function

```
FUNCTION identify_sustainability_conflicts(state_matrix, device_list, power_threshold):
    FOR day FROM 0 TO 6:  # Monday = 0, Sunday = 6
        cumulative_power_array = ARRAY[1440] initialized to 0
        FOR minute FROM 0 TO 1439:
            cumulative_power_array[minute] = SUM(get_average_power(device, state_matrix[day][minute][device]) FOR device
                in device_list)
        cumulative_power_arrays[day] = cumulative_power_array
    conflicts = EMPTY_LIST
    FOR day FROM 0 TO 6:
        current_array = cumulative_power_arrays[day]
        conflict_start = NULL
        FOR minute FROM 0 TO 1439:
            IF current_array[minute] > power_threshold:
                IF conflict_start IS NULL:
                    conflict_start = minute
            ELSE IF conflict_start IS NOT NULL:
                conflicts.APPEND({day: day, start: conflict_start, end: minute - 1})
                conflict_start = NULL
    RETURN conflicts
```

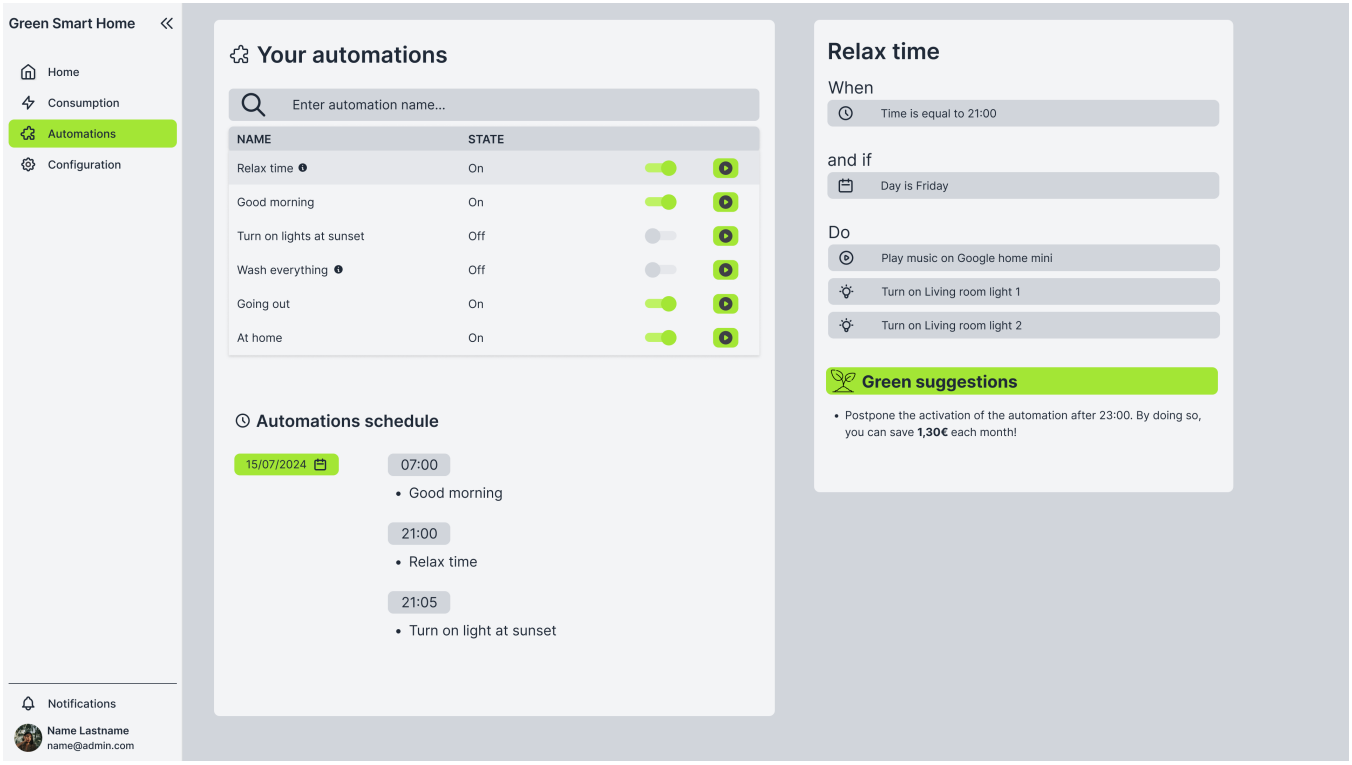

Figure 1: A screenshot of the Digital Twin web application. The "Relax Time" automation details include the green suggestions.

the simulation of their impact to identify conflicts and possible optimizations. Additionally, it periodically extracts HA's device history and computes the moving average of states' power demand and duration that are needed for the simulation feature. HA is integrated into the system thanks to the OS and its API. It autonomously manages automations and devices, collecting data on appliance state changes and saving them into its internal database.

5 Conclusion

This work presented a feature of our Digital Twin of a green smart home to help users optimize energy consumption and avoid overcoming the energy meter load capacity. We plan to integrate machine learning to predict energy consumption in subsequent minutes, hours, or days, and thus provide further green suggestions. Customization of suggestions based on user preferences will also be implemented. The deployment and experimentation of the system in real homes will finally be carried out.

Acknowledgments

This work has been supported by the Italian MUR PRIN 2022 PNRR Project P2022YR9B7, End-User Development of Automations for Explainable Green Smart Homes, funded by European Union - NextGenerationEU.

References

[1] Barbara Rita Barricelli, Alessandro Bondioli, Daniela Fogli, Letizia Iemmolo, and Angela Locoro. 2024. Creating Routines for IoT Ecosystems through Conversation with Smart Speakers. *International Journal of Human–Computer Interaction* 40, 20 (2024), 6109–6127. https://doi.org/10.1080/10447318.2023.2247845

[2] Barbara Rita Barricelli, Luca Cotti, Daniela Fogli, Davide Guizzardi, and Matteo Pigoli. 2024. A Digital Twin to Enhance Energy Consumption Awareness in a Smart Home. In *Proceedings of the 2024 International Conference on Advanced Visual Interfaces* (Arenzano, Genoa, Italy) *(AVI '24)*. Association for Computing Machinery, New York, NY, USA, Article 60, 3 pages. https://doi.org/10.1145/3656650.3656708

[3] Barbara Rita Barricelli and Stefano Valtolina. 2015. Designing for End-User Development in the Internet of Things. In *End-User Development*, Paloma Díaz, Volkmar Pipek, Carmelo Ardito, Carlos Jensen, Ignacio Aedo, and Alexander Boden (Eds.). Springer International Publishing, Cham, 9–24. https://doi.org/10.1007/978-3-319-18425-8_2

[4] Danilo Caivano, Daniela Fogli, Rosa Lanzilotti, Antonio Piccinno, and Fabio Cassano. 2018. Supporting end users to control their smart home: design implications from a literature review and an empirical investigation. *Journal of Systems and Software* 144 (2018), 295–313. https://doi.org/10.1016/j.jss.2018.06.035

[5] Rui Camacho, Paulo Carreira, Inês Lynce, and Sílvia Resendes. 2014. An ontology-based approach to conflict resolution in Home and Building Automation Systems. *Expert Systems with Applications* 41, 14 (2014), 6161–6173. https://doi.org/10.1016/j.eswa.2014.04.017

[6] Fulvio Corno, Luigi De Russis, and Alberto Monge Roffarello. 2019. Empowering End Users in Debugging Trigger-Action Rules. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3290605.3300618

[7] Fulvio Corno, Luigi De Russis, and Alberto Monge Roffarello. 2020. HeyTAP: Bridging the Gaps Between Users' Needs and Technology in IF-THEN Rules via Conversation. In *Proceedings of the International Conference on Advanced Visual Interfaces* (Salerno, Italy) *(AVI '20)*. Association for Computing Machinery, New York, NY, USA, Article 23, 9 pages. https://doi.org/10.1145/3399715.3399905

[8] Fulvio Corno, Luigi De Russis, and Alberto Monge Roffarello. 2021. From Users' Intentions to IF-THEN Rules in the Internet of Things. *ACM Trans. Inf. Syst.* 39, 4, Article 53 (aug 2021), 33 pages. https://doi.org/10.1145/3447264

[9] Luca Cotti, Davide Guizzardi, Barbara Rita Barricelli, and Daniela Fogli. 2024. Enabling End-User Development in Smart Homes: A Machine Learning-Powered Digital Twin for Energy Efficient Management. *Future Internet* 16, 6 (2024), 16 pages. https://doi.org/10.3390/fi16060208

[10] Liyanage C. De Silva, Chamin Morikawa, and Iskandar M. Petra. 2012. State of the Art of Smart Homes. *Engineering Applications of Artificial Intelligence* 25, 7 (2012), 1313–1321. https://doi.org/10.1016/j.engappai.2012.05.002

[11] Simone Gallo, Andrea Mattioli, Fabio Paterno, Barbara Rita Barricelli, Daniela Fogli, and Davide Guizzardi. 2024. An Architecture for Green Smart Homes Controlled by End Users. In *Proceedings of the 2024 International Conference on Advanced Visual Interfaces* (Arenzano, Genoa, Italy) *(AVI '24)*. Association for Computing Machinery, New York, NY, USA, Article 62, 3 pages. https://doi.org/10.1145/3656650.3656710

[12] Simone Gallo and Fabio Paterno. 2022. A Conversational Agent for Creating Flexible Daily Automation. In *Proceedings of the 2022 International Conference on Advanced Visual Interfaces* (Frascati, Rome, Italy) *(AVI 2022)*. Association for Computing Machinery, New York, NY, USA, Article 38, 8 pages. https://doi.org/10.1145/3531073.3531090

[13] Giuseppe Ghiani, Marco Manca, Fabio Paternò, and Carmen Santoro. 2017. Personalization of Context-Dependent Applications Through Trigger-Action Rules. *ACM Trans. Comput.-Hum. Interact.* 24, 2 (2017), 33 pages. https://doi.org/10.1145/3057861

[14] Jack Kelly and William Knottenbelt. 2015. The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes. *Scientific Data* 2, 1 (2015), 150007. https://doi.org/10.1038/sdata.2015.7

[15] André Sousa Lago, João Pedro Dias, and Hugo Sereno Ferreira. 2021. Managing non-trivial internet-of-things systems with conversational assistants: A prototype and a feasibility experiment. *Journal of Computational Science* 51 (2021), 1–12. https://doi.org/10.1016/j.jocs.2021.101324

[16] Ya-Hua Lee and Fuchun Joseph Lin. 2019. Situation Awareness and Conflict Resolution in Smart Home with Multiple Users. In *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*. IEEE, New York, 852–857. https://doi.org/10.1109/WF-IoT.2019.8767178

[17] Sara Maenza, Andrea Mattioli, and Fabio Paternò. 2024. An Approach to Explainable Automations in Daily Environments. *Intelligent Human Systems Integration (IHSI 2024): Integrating People and Intelligent Systems* 119, 119 (2024), 507–513.

[18] Marco Manca, Fabio Paternò, Carmen Santoro, and Luca Corcella. 2019. Supporting end-user debugging of trigger-action rules for IoT applications. *International Journal of Human-Computer Studies* 123 (2019), 56–69. https://doi.org/10.1016/j.ijhcs.2018.11.005

[19] Andrea Monacchi, Dominik Egarter, Wilfried Elmenreich, Salvatore D'Alessandro, and Andrea M. Tonello. 2014. GREEND: An energy consumption dataset of households in Italy and Austria. In *2014 IEEE International Conference on Smart Grid Communications (SmartGridComm)*. IEEE, New York, 511–516. https://doi.org/10.1109/SmartGridComm.2014.7007698

[20] Kenzhegali Nurgaliyev, Dario Di Mauro, Nawaz Khan, and Juan Carlos Augusto. 2017. Improved Multi-user Interaction in a Smart Environment Through a Preference-Based Conflict Resolution Virtual Assistant. In *2017 International Conference on Intelligent Environments (IE)*. IEEE, New York, 100–107. https://doi.org/10.1109/IE.2017.21

[21] Bauyrzhan Ospan, Nawaz Khan, Juan Augusto, Mario Quinde, and Kenzhegali Nurgaliyev. 2018. Context Aware Virtual Assistant with Case-Based Conflict Resolution in Multi-User Smart Home Environment. In *2018 International Conference on Computing and Network Communications (CoCoNet)*. IEEE, New York, 36–44. https://doi.org/10.1109/CoCoNet.2018.8476898

[22] Blase Ur, Elyse McManus, Melwyn Pak Yong Ho, and Michael L. Littman. 2014. Practical Trigger-Action Programming in the Smart Home. In *Proc. of the SIGCHI Conference on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI '14)*. ACM, New York, NY, USA, 803–812. https://doi.org/10.1145/2556288.2557420