Taylor & Francis
Taylor & Francis Group

Behaviour & Information Technology

ISSN: 0144-929X (Print) 1362-3001 (Online) Journal homepage: www.tandfonline.com/journals/tbit20

A smart home platform integrating an energy-aware digital twin and an LLM-enabled conversational agent

Barbara Rita Barricelli, Daniela Fogli, Simone Gallo, Davide Guizzardi, Sara Maenza, Andrea Mattioli & Fabio Paternò

To cite this article: Barbara Rita Barricelli, Daniela Fogli, Simone Gallo, Davide Guizzardi, Sara Maenza, Andrea Mattioli & Fabio Paternò (16 May 2026): A smart home platform integrating an energy-aware digital twin and an LLM-enabled conversational agent, Behaviour & Information Technology, DOI: 10.1080/0144929X.2026.2672579

To link to this article: https://doi.org/10.1080/0144929X.2026.2672579

Published online: 16 May 2026.

BEHAVIOUR & INFORMATION TECHNOLOGY
https://doi.org/10.1080/0144929X.2026.2672579

A smart home platform integrating an energy-aware digital twin and an LLM-enabled conversational agent

Barbara Rita Barricelli [a], Daniela Fogli [a], Simone Gallo [b], Davide Guizzardi [a], Sara Maenza [b], Andrea Mattioli [b] and Fabio Paternò [b]

[a]Department of information engineering, University of Brescia, Brescia, Italy; [b]Institute of Information Science and Technologies "Alessandro Faedo" (ISTI), National Research Council (CNR), Pisa, Italy

ABSTRACT
This paper presents the design, implementation, and evaluation of the Green Smart Home (GSH) platform, a novel end-user development solution that integrates an energy-aware digital twin with an LLM-enabled conversational agent. Motivated by the growing complexity of IoT ecosystems and the need for sustainability-oriented tools accessible to non-experts, the system enables users to create automations through natural language while receiving real-time feedback on conflicts, activation chains, and energy-related issues. The GSH introduces a unified platform that combines conflict detection, consumption forecasting, contextual explanations, and an intuitive visual representation of the home. An autoethnographic study of the system in six of the authors' homes provided insights into long-term use, highlighting increased user awareness of energy consumption, active experimentation with automations, and the value of conversational interaction, while also revealing limitations. Findings confirm the GSH's potential to enhance sustainability, usability, and user empowerment in smart home environments.

ARTICLE HISTORY
Received 12 December 2025
Accepted 26 April 2026

KEYWORDS
Smart home; automation; conflict resolution; sustainability; digital twins; chatbots

1. Introduction

In recent years, digital technologies have undergone a rapid evolution driven by the convergence of artificial intelligence, the Internet of Things (IoT), and advanced data-driven modelling techniques. Among these developments, Digital Twins (DTs) and Large Language Models (LLMs) have emerged as two particularly influential paradigms. Digital twins enable the creation of dynamic digital representations of physical environments, devices, or systems that continuously reflect their real-world counterparts through sensor data and computational models (Barricelli, Casiraghi, and Fogli 2019; Grieves 2015; VanDerHorn and Mahadevan 2021). Originally adopted in industrial and engineering contexts, digital twins are increasingly being explored in domains such as smart homes, healthcare, urban management, and energy systems, where they can support simulation, monitoring, prediction, and optimisation of complex environments (Liu et al. 2024; Nadeem et al. 2025; Yao et al. 2023).

At the same time, the recent explosion of Large Language Models in everyday applications has dramatically expanded the ways in which people interact with digital systems. LLM-based conversational agents are now embedded in a wide range of tools, including productivity applications, home assistants, customer support systems, and educational platforms (Liang and Tong 2025; Xu et al. 2024). Their ability to interpret natural language, generate content, and assist with complex tasks has lowered the barrier to accessing advanced computational capabilities, enabling non-experts to interact with digital systems in more intuitive ways (Cheng and Zhang 2025).

However, these technological developments also bring new challenges, particularly regarding the growing presence of automated behaviours in everyday environments. Modern smart environments increasingly rely on automations that coordinate devices, services, and data streams in order to perform tasks such as energy management, security monitoring, or personalised assistance. These automations are often implemented through mechanisms such as trigger-action rules, machine learning models, or AI-driven agents. While such mechanisms can greatly enhance convenience and efficiency, they also introduce issues related to transparency, predictability, and user control.

CONTACT Davide Guizzardi davide.guizzardi@unibs.it Dipartimento di Ingegneria dell'Informazione, Via Branze 38, 25123, Brescia, Italy

Users frequently struggle to understand why certain automated behaviours occur, how different automations interact, or how to modify them when circumstances change. As smart environments become more complex and incorporate AI-driven decision-making, the risk increases that automations will behave in ways that are difficult for users to anticipate or explain. This lack of transparency may reduce trust, limit adoption, and create barriers to meaningful user control.

In this context, integrating digital twins, conversational AI, and end-user development (EUD) (Lieberman et al. 2006) approaches opens promising research directions. Digital twins can provide a structured and interpretable representation of the environment in which automations operate, enabling simulation and exploration of alternative behaviours. Meanwhile, LLM-based conversational interfaces can allow users to create, explain, and modify automations through natural language interaction. Together, these technologies may support new forms of human-centred automation management, where users are not only recipients of automated behaviour but active participants in understanding, configuring, and evolving the intelligent environments around them.

Such needs are emerging in several application domains, including retail (Chawla and Jain 2024), industry (Manca, Paternò, and Santoro 2022a), and education (Andrao et al. 2025). One domain that is particularly involved in these technological trends and has a wide social impact is the smart home, as common domestic objects are increasingly becoming connected and digitally configurable. In such contexts, one common concern is to limit the energy consumption for economic and environmental reasons.

We have thus designed a solution to meet these needs by adopting two emerging approaches: the use of a digital twin to facilitate awareness of energy consumption and a conversational agent based on a LLM to support the creation and debugging of home automations. Indeed, when combining multiple automations in a smart space, several potential conflicts or security issues can occur (Chen et al. 2022). Some of them have not been considered in previous work on debugging trigger-action rules (Corno, De Russis, and Monge Roffarello 2019).

The result is a novel approach in the area of end-user development, which leverages emerging interactive conversational techniques based on generative artificial intelligence and provides users with support for sustainability in their homes. In particular, based on previous work on conversational agents (Gallo, Paternò, and Malizia 2024), we have developed a novel solution that also addresses home energy-related issues (Costanza, Ramchurn, and Jennings 2012), generating automations following the trigger-action (Ur et al. 2014) programming structure. We present how we have integrated these approaches and how we have validated the developed solution first through a laboratory user test, and then through a deployment in real homes. In the laboratory study, we have considered specific chatbot usability metrics (Borsci et al. 2023). We also aimed to test such a proposal in realistic environments, unlike previous work (Giudici et al. 2024), and considering that previous validations of trigger-action-based approaches in real smart homes (Manca, Paternò, and Santoro 2022b) did not address energy-related issues. For this purpose, we adopted a first-person approach (Coutaz and Crowley 2016) to explore the actual possibilities of the novel platform.

In the paper, after discussing the related work, we introduce the human-centred approach adopted to design the proposed solution, its first version, along with the results of the associated laboratory usability evaluation. Next, we present the final architecture, describing both the backend and frontend design and functionalities. We then report on an autoethnographic study during which the platform was installed in six real homes to validate its scalability and usability. Lastly, we discuss the findings of the study, provide some concluding remarks and outline directions for future work.

2. Related work

Despite the pervasiveness of Internet of Things objects, interfaces to define automated behaviours between them still face critical issues that can impact their successful adoption in everyday settings. Smart home interfaces are primarily designed for single-user situations, overlooking the fact that multiple household tenants may have different automation needs and long-term goals. Moreover, interfaces should support more natural interactions compared to typical visual-based wizard interfaces, such as through conversational agents. Further design challenges include providing debugging and simulation tools, mechanisms for conflict resolution, and balancing a level of control appropriate for users' programming expertise (Brich et al. 2017). An issue that has recently emerged is how to enhance users' awareness of power consumption in smart homes and improve sustainability through automations. In this section, we analyse which relevant work has been put forward that supports users in detecting and solving issues between automations and supports them in behaving sustainably.

2.1. Approaches to detecting problems in automation rules

It is important to examine approaches that aim to improve the management of smart environments by providing tools to address conflicts between automations. Several studies (Chaki and Bouguettaya 2021; Lee and Lin 2019; Nurgaliyev et al. 2017; Ospan et al. 2018) have investigated which strategies to use in the multi-user case. However, these solutions are specific to multi-user cases and may not be suitable for all situations, requiring a more general approach to automations debugging. Chen and colleagues (Chen et al. 2022) categorised problems into three main types: rule preventions, where one rule unintentionally blocks another from triggering; rule collisions, corresponding to a situation when two automations act in an incompatible way on the same object; and unexpected rule chains, where the outcome of one action inadvertently activates another rule. These logical flaws can result in unpredictable smart home behaviours and may cause annoyance and loss of trust in the system, but also pose user risks, including unintentional data leaks or increased vulnerability to external attacks.

One of the first approaches in this regard was presented by Coutaz and Crowley (2016). AppsGate provides several tools for monitoring the smart home, such as a timeline and a dependency graph between entities. The graph shows, for example, whether multiple automations act on the same object, or whether an object is offline, representing a first application of the concept of problem detection.

EUDebug (Corno, De Russis, and Monge Roffarello 2019) is a system aimed at debugging trigger-action rules. It can highlight the problems that can occur in the set of user-defined rules. It also allows for the simulation of the automation activations, which act as an explanation. A graph approach based on the Semantic Colored Petri Net formalism is used to simulate system states and detect problems. However, the system is designed to work with simple single-trigger single-action automations. A different solution is presented in ITAD (Manca et al. 2019). This system is designed to support users in understanding whether the defined rules lead to the desired behaviour and, if not, why they fail. The system implements a Whyline approach for TAP rules. Users can ask 'why' and 'why not' questions to know why a rule verifies or does not. A conflict analysis functionality can highlight potential conflicts between rules that are simultaneously verified. Additionally, a functionality allows users to insert contextual values to simulate whether rules activate in specific situations. This approach can only detect conflicts and provides only basic explanations. FORTNIoT (Coppers, Vanacken, and Luyten 2020) represents an approach designed to improve user awareness of smart environments by anticipating which automations will be activated and what effects they will produce. The system analyses causal relationships between triggers and actions to clarify the logic behind each automation verification. A further development (Coppers, Vanacken, and Luyten 2022) introduces methods to handle unintended behaviours of the system by allowing temporary exceptions to automations. Despite these enhancements, the visual interface becomes difficult to manage as the number of connected devices increases.

Other approaches use formal methods to constrain automation behaviour. MenShen (Bu et al. 2018) employs Linear Hybrid Automata to analyse individual rules, detecting whether undesired system states can occur and suggesting fixes based on user-defined conditions. AutoTap (Zhang et al. 2019) leverages Linear Temporal Logic to enforce user-defined restrictions on environmental states. It can generate new rules or adjust existing ones to meet these constraints and user preferences. These methods depend on users explicitly defining the system's properties.

Huang et al. (Huang et al. 2024) present a framework for detecting interactions between automation rules using transformers to generate TAP rules and their descriptions. The method addresses both explicit and implicit inter-rule vulnerabilities, representing rules as sequences of IoT services. However, no user interface to use the model is presented, and the approach is limited to simple one-trigger–one-action rules.

An initial study (Gallo, Paternò, and Malizia 2024) aims to detect and resolve different types of problems between automations in the IoT domain. The approach analyses from an algorithmic point of view how to identify problems between automations and how automations can be problematic for a goal. However, this solution was only evaluated in a laboratory test, without evaluation in real settings. Furthermore, the energy management aspect was not investigated.

Despite extensive research on smart home automation, several gaps remain unaddressed. Most existing systems focus on either rule management and conflict detection or sustainability, but little attention has been devoted to integrating both aspects within a unified framework. Approaches to debugging automations often rely on formal models or rule simulations designed for simple, one-trigger, one-action rules, which limit their applicability in real-world environments. Moreover, while several studies have explored visual tools, few have investigated how the creation and management of rules, as well as potential issues, can be made accessible through natural and conversational interfaces. There is also a scarcity of studies focusing on longitudinal assessments to investigate the real-world implementations of such platforms, in particular concerning energy management (Houshangi 2025). Other aspects that require more attention include the evolution of user behaviour, the adoption of platforms, and the performance of systems across various household configurations.

2.2. Sustainability in smart home

The possibility of combining an EUD methodology based on a conversational agent for automation creation with a digital twin of the smart home opens up new possibilities to support sustainable behaviours. This approach can enhance users' awareness and control, leading to optimised appliance usage and lower energy consumption (Giudici, Crovari, and Garzotto 2025). However, this possibility has not been particularly considered in the literature. Previous work has focused on other aspects, such as the development of algorithms to optimise the balance between comfort and energy savings (Malek et al. 2022), providing users with visualisations concerning energy consumption – e.g. (Costanza, Ramchurn, and Jennings 2012; Stogia et al. 2024) – or persuading them to behave more sustainably (e.g. Alan et al. 2016; Beheshtian et al. 2020).

A pioneering work in this context was FigureEnergy (Costanza, Ramchurn, and Jennings 2012), a graphical visualisation system that allows users to annotate and manipulate a representation of their energy consumption. The platform was assessed through a longitudinal study, in which twelve participants used it in their homes. The study shows that the manual annotation encouraged users to engage with the data, helping them make sense of it. The system only allows for the annotation and exploration of consumption data, without allowing users to create automated behaviours.

Giudici et al. (2024) introduced GreenIFTTT, a conversational agent powered by GPT-4 designed to help users create eco-friendly smart home automations. The system focuses on promoting energy efficiency and supporting sustainable electricity consumption by translating users' environmental goals into concrete automations. Unlike traditional systems that only visualise energy data or attempt to persuade users to act sustainably, GreenIFTTT actively engages them in managing their home environment to lower energy use and environmental impact. However, the study does not address real-time energy data management, making it unclear how effectively the system evaluates or optimises automation sustainability.

Cotti et al. (2024) propose an approach based on deep learning for recognising appliance activations, with the goal of estimating energy consumption. This information is leveraged to suggest more efficient automation start times or to alert users about potential issues caused by the simultaneous activation of multiple energy-intensive appliances. However, the paper does not discuss how end users can create automations with the proposed system, and issues arising from conflicting device access, rule chains, or loops are not considered.

Overall, it becomes clear that there is a need for a platform to manage IoT environments that enable the creation of automations, combining conflict detection, explanation, and sustainability awareness in a single integrated system. Such a platform should provide intuitive, visual, and conversational interfaces for end users and support dynamic smart home contexts where automation goals and priorities can evolve over time. We designed and developed the Green Smart Home (GSH) system to fill this gap. The platform was assessed through an autoethnography study conducted by six authors, which enables the in-depth exploration of the system's implications in a real-world context of use.

2.3. System evaluation through first-person experience

First-person research methods have been recognised as a significant tool for reflecting on the use of technology in everyday life, relying on firsthand experience as a means of knowing (Desjardins et al. 2021). Among these methods, autoethnography is usually based on describing personal experiences and opinions while living with digital systems (Cecchinato, Cox, and Bird 2017; Lucero 2018) that, in some cases, a researcher has contributed to designing (Coutaz and Crowley 2016; Zhang and Wakkary 2014). Autoethnography (Chang 2016; Ellis 2008) has also been employed in the literature to explore the social implications of smart technology. For instance, in (Aagaard, Christensen, and Gram-Hanssen 2023), one of the authors kept a diary for 20 months detailing how she learned to live with a smart home by interacting with a smart alarm, a lighting management system, a smart heating control, and a smart electric vehicle. The autoethnographic approach was praised for its 'detailed insider perspective', which provides in-depth insights and a day-to-day, longitudinal perspective on living with a smart home. According to (Hine 2020), using autoethnography in one's own home allows one to better grasp how the relation between technology and users develops. The autoethnographic study described in (Turner, Nurse, and Li 2022) analysed smart home cybersecurity practices from the expert perspective over an 80-day period, investigating how cybersecurity was managed and discussed among family members. The study provided interesting insights for encouraging the application of this method in those situations where understanding the adoption of digital technology in daily life is challenging.

In summary, autoethnography allows experts to explore the usefulness of first-person, reflexive research in a specific area. Empathising with non-expert users is a further positive side effect of this method. However, results can be biased by personal thoughts, experiences, and behaviours; thus, eliciting knowledge that could be generalisable might become difficult. To overcome this limitation, this paper presents an evaluation that incorporates data from the diverse experiences of six authors with the system. One further author analysed the data, conducted a thematic analysis of the participants' diaries, and then discussed the findings with the participants in the study to avoid misunderstandings and preserve the personal and reflexive nature of the research (McDonald, Schoenebeck, and Forte 2019).

3. Design approach

A human-centred approach has been adopted to design the GSH system, beginning with user research, proceeding through the creation of static and dynamic prototypes, and culminating in the development of the first interactive prototype, which was evaluated with end users. The results of this first cycle led to gathering several feedback and indications to improve the system. These also provided important hints on how to better integrate the system's main features, namely the conversational approach to automation creation and the DT functionalities for monitoring and managing the smart home. Here, we outline the steps that led to the development and evaluation of the first prototype, while the final system will be presented in Section 4, followed by its evaluation through the autoethnographic study discussed in Section 5. The diagram in Figure 1 provides a comprehensive overview of the phases involved in the adopted human-centred design approach.

3.1. User research

User research began with a semi-structured interview with an expert in the home automation domain, who works for an Italian company that develops and sells domotic solutions. We explored several topics related to our project objectives, including the creation and management of automations, the interoperability of smart devices with those of other companies, and the approaches implemented to achieve sustainability and reduce energy consumption. What emerges from the interview is that commercial domotic solutions are far from being easily personalised by home inhabitants through automations. This activity is usually performed by installers. Therefore, when the habits of households change, an intervention by the installer is usually required, also to prevent end users from creating wrong or dangerous automations. Similarly, only the installers can configure the systems to reduce energy consumption, despite a growing interest from households in monitoring energy consumption and enhancing smart home efficiency.

After this preliminary exploration, we conducted two online surveys among end users, employing two types of

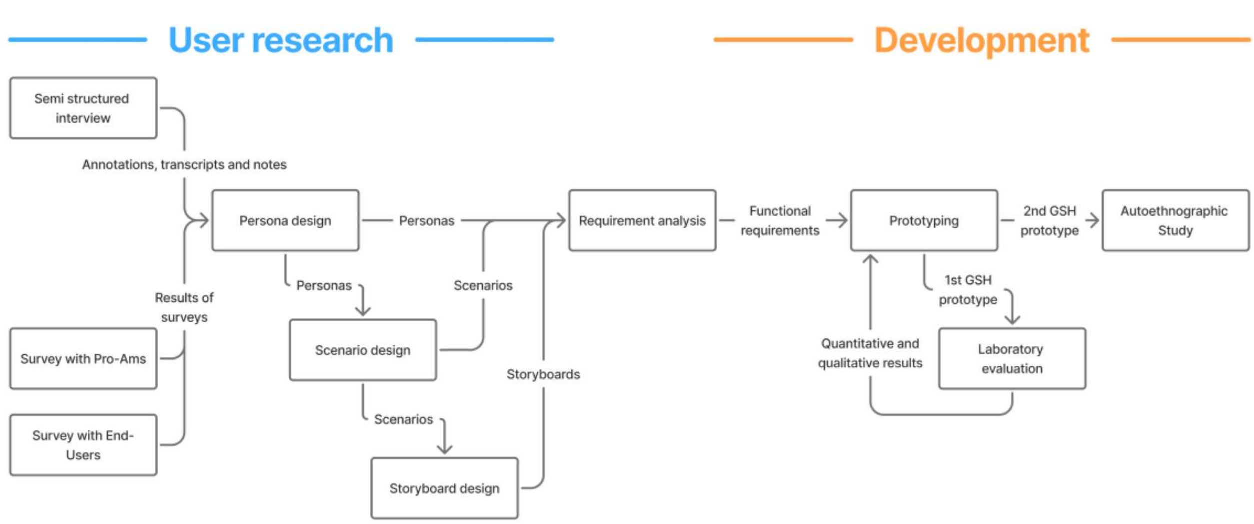

Figure 1. Diagram of the human-centered design process.

questionnaires: the former targeted a general population of potential end users, and the latter a community of end users specifically interested in home automation. From the data acquired through these activities, personas, scenarios, and storyboards have been defined to shed light on the possible functional and non-functional requirements of the GSH system.

3.1.1. Online surveys among end users

A first questionnaire targeted the general population of potential users of a green smart home. The study investigated users' attitudes towards current technologies for smart homes, their opinions on environmental sustainability, and the approaches they adopted for creating and managing automations for smart homes. This online survey involved 111 participants (age range 17-70, M = 38.8, SD = 12.27). Notwithstanding that most participants (73%) held a degree or a PhD, a high percentage (63%) declared no experience or low experience with programming. Among the participants, 63 had at least 1 smart device, whilst only 26 used automations. Most of these 26 users (80%) reported encountering issues with their automation creation tool. The main hurdles they faced were unsupported devices, unavailable actions, and conditions not aligned with the desired scenarios. They also underlined that the user interfaces of such tools exhibited poor usability. Dealing with conflict automations and consequent unexpected behaviours was recognised as the main concern when multiple inhabitants live in the same smart environment. Therefore, participants' interest in a system that can detect conflicts and suggest possible solutions was quite high. From this survey, it became clear that current solutions for smart homes still lack adequate usability and relevant features to meet end users' needs. The possibilities provided by current technologies often remain hidden to most end users, especially when it comes to the personalisation possibilities offered by automations. Thus, better EUD techniques are needed to make automation creation easier and engaging, and artificial intelligence techniques could support users with suggestions for conflict resolution and efficient energy management. Adequate visualisation features may then allow consumption monitoring and improve sustainability awareness.

For the second online survey, 36 participants (age range 32-70, M = 47, SD = 8.86) were recruited through specific social communities among technical users with amateur experience in managing smart homes and creating automations – called *pro-ams* in (Leadbeater and Miller 2004). Participants had varying levels of programming expertise, ranging from no expertise at all or low expertise (18 participants) to moderate expertise with different programming languages (11 participants), to professional expertise (7 participants). All respondents had considerable experience with smart homes (from 6 months to more than 3 years). Participants reported interacting with their smart home devices using instant commands and automations. Automations were used for comfort purposes (e.g. managing temperature), energy saving, environmental and network security management, and organisation-related tasks (e.g. calendar notifications, weather updates, medication reminders, and the like).

From the second survey, it emerged that experienced users are very interested in features for creating and managing automations for energy management; however, these features require user effort and expertise in existing platforms. Concerns were expressed about the lack of intuitive user interfaces for automation creation, observing how inexperienced users might find it difficult to adopt such technologies. The need for protocol standardisation, supporting interoperability, and making systems easier to configure and maintain was another important requirement underlined by the participants.

Further details about the questions included in the surveys and the collected results can be found in (Barricelli et al. 2024).

3.1.2. Personas, scenarios, and storyboards

The results of the online surveys led us to define four personas – Martina, Luca, Anna, and Franco – representing the different characteristics of users that emerged from the survey answers, in terms of age, biography, programming experience, interest in sustainability and/or energy cost reduction, needs regarding a green smart home, frustrations related to existing smart home technologies, and motivations in using a novel smart home solution. For the sake of brevity, we summarise in Table 1 the main characteristics of such personas.

We then created a series of scenarios that described possible interactions between the defined personas and the GSH system. For the sake of space, Table 2 reports a synthetic version of such scenarios.

To complete the user research activity, we created five storyboards that pictorially illustrate the above scenarios. Figure 2 shows the storyboard describing the fifth scenario (*Use of the consumption forecast feature).*

Personas, scenarios, and storyboards synthetise the results of the user research carried out through the interview with the domain expert and the surveys with generic end users and pro-ams. The definition of different personas led to characterise the end users of the GSH system as individuals who are interested in

Table 1. Summary of personas.

Persona	Profile	Context and Technical background	Main goals	Frustrations
Martina (31)	Graphic designer, lives alone in an apartment	Limited technical expertise; strong interest in sustainability	Monitor energy use and carbon footprint; receive personalised recommendations	Limited control over energy usage; lack of personalised guidance
Luca (40)	Software engineer, lives with partner in an IoT-enhanced house	High technical expertise; multiple existing smart devices	Centralised control of devices; advanced automations	Compatibility issues between devices; inconsistencies in devices' performance and functionality
Anna (38)	Housewife, lives with husband and two children	Low–medium technical expertise; interested in cost-reducing alternatives	Reduce energy bills; avoid interruptions due to power overloads	High-cost energy bills; frequent circuit breaker trippings
Franco (56)	Doctor, lives with wife and college-age son	Medium technical expertise; automation enthusiast	Coordinate family automations; maintain household harmony	Conflicts between automations and inhabitants' actions

technology and smart homes but are not necessarily experts in smart device control or automation creation; these users desire to behave sustainably or to exploit technology to reduce energy costs. Scenarios and storyboards allowed the design team to reflect on the different functionalities that could meet the needs and address the frustrations of the target users (such as monitoring energy consumption, having a unique system to control different kinds of devices, avoid energy interruptions, manage the contemporary operation of different automations), also considering their different levels of technical background.

Table 2. Summary of the scenarios.

Name	Summary of scenario description
Creation of a green automation after environmental footprint analysis	Martina reviews her household's energy consumption and carbon report, identifies high heating usage, and creates an 'Out of home' automation that reduces heating and switches off selected smart plugs when she is away.
Activation of different appliances with conflict due to excessive consumption	Luca activates some lights and air conditioning at 18; when Sara attempts to start the oven, one hour later, the system predicts excessive power consumption and notifies her. After discussing, they turn off the air conditioner before using the oven.
Obtaining a suggestion for a better start time for automation	Anna schedules the dishwasher for 20:00; the system suggests postponing it to 23:00 to save money. Anna decides to accept the system's suggestion and adjusts the automation schedule accordingly.
Notification of conflict between automations of different users	Luca creates an automation to dim the lights at 20:00, while Sara creates one to brighten them at the same time. The system detects the conflict and suggests delaying one of the two automations; Sara accepts the adjustment.
Use of the consumption forecast feature	Franco explores the consumption forecasting page. After reviewing tailored green suggestions, he adjusts thermostat settings, schedules appliances during off-peak hours, and decides to replace his refrigerator with a more efficient model.

3.2. Functional requirements

The user research activities led us to identify the functional requirements of the GSH system, which aims to provide a novel approach to the sustainable management of a smart home by integrating various physical and digital resources in an EUD environment, fostering interoperability and usability. The defined requirements for the proposed solution are:

1. *End-user creation and modification of automations*. The system should enable users, even those who are not software developers, to easily create, modify, and monitor automations in their domestic environment. This involves the use of intuitive, no-code interfaces that employ innovative interaction paradigms, such as natural language.
2. *Management of dependencies among multiple automations*. The system should help users detect and understand issues that occur between automations, as well as between the deployed automations and long-term sustainability objectives. Furthermore, possible resolutions for the issues should be suggested.
3. *Automation and recommendation explanations*. The smart home should be able to provide explanations regarding the possible execution and effects of the automations, informing users about the changes that can be made to the automations to maintain or achieve a certain effect. Explanations justifying the system's recommendations for resolving issues that occurred between automations should be provided upon request by the user.
4. *Prediction and simulation capabilities*. The system should utilise smart home historical data, applying machine learning and deep learning techniques to these data to provide energy consumption

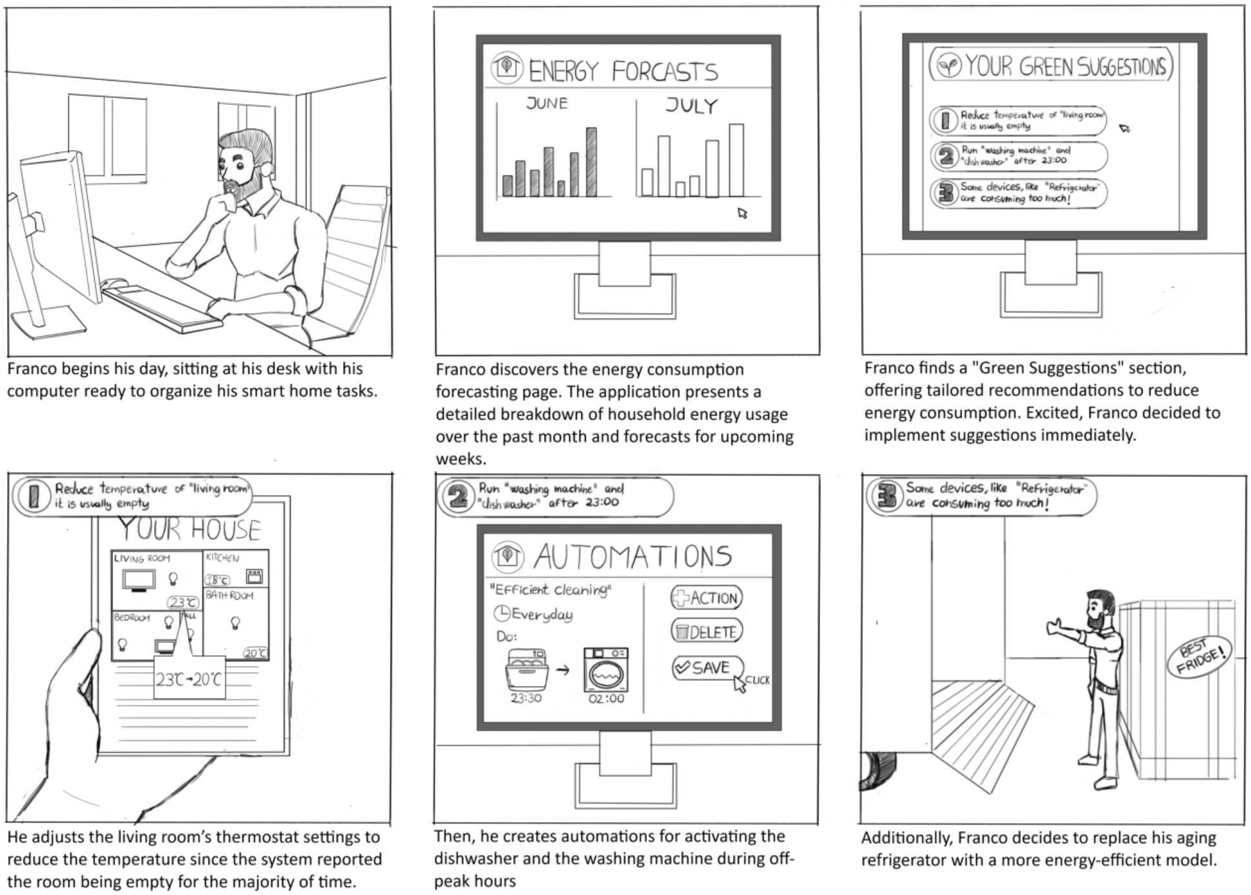

Figure 2. The storyboard associated with the use of the consumption forecast feature scenario.

predictions. Using predictions, the system should be able to provide suggestions for optimising newly created automations in terms of energy consumption.

5. *Descriptions of the state of the smart space*. The system should provide users with information about the state of various connected objects and devices, as well as which automations are active between them. Together with an understanding of what is needed to execute an automation, these state descriptions are essential to ensure that users can form a clear idea of the system's functioning.
6. *Integration of a Digital Twin for enhanced management, visualisation, and analytics*. The users' request for a central access point to control the connected devices, while ensuring device interoperability and offering a usable interface, inspired the idea of developing a digital twin of the smart home. The digital twin should provide a real-time, dynamic representation of the physical space, allowing users to directly activate connected objects and devices, visualise energy consumption, and gain insights into their home's sustainability and the impact of the automations they create.

3.3. First version of the GSH system

A first version of the GSH system satisfying the first four requirements has been presented in (Barricelli et al. 2025). In the following, the main features of this first prototype are summarised, as well as the results of the user study carried out in the laboratory.

3.3.1. First prototype

The developed prototype includes a conversational tool that enables end users to create automations in a natural and engaging manner. Such a tool leverages underlying services to implement various controls on the automation being created, thereby preventing conflicts with other automations and issues related to overall household energy consumption.

The architecture of the prototype encompasses three main components: the Chatbot, the Rule Checker, and the Energy Checker. The Chatbot interacts with the user, leveraging GPT-4o as a dialogue manager. To access data about devices and automations, save the newly generated automations, and enable automation execution, the Chatbot is integrated with Home

Assistant, an open-source home automation platform that integrates smart devices from different manufacturers into a unified ecosystem. The Chatbot can verify if conflicts occur between the automation being created and the existing automations in the database. If this is the case, it is also responsible for suggesting possible modifications to existing automations or the new one to the user. This behaviour is achieved with the help of the Rule Checker, which is called through its application programming interface (API). This component applies two types of control: the former compares the actions of the automation under creation with the actions of all the existing automations to identify if there are pairs of actions that perform conflicting operations (e.g. 'turn on' versus 'turn off') on the same device in the same time interval; the latter examines the automations for potential direct or indirect chains between automations. When the Rule Checker identifies a conflict, it employs a suitable algorithm to suggest potential corrections to the automation under creation by selecting, from a set of candidate automations, the one most closely aligned with the one submitted by the user. Similarly, the Chatbot can verify if energy consumption issues occur due to the creation of a new automation that may cause energy overload and can provide suggestions about energy optimisation. To this purpose, the Chatbot exploits the Energy Checker through API calls. This component utilises historical data collected through Home Assistant to estimate the power consumption and duration of the actions of the automation being created; based on such an estimation, it calculates the cumulative power of all devices for each minute of the day. If the overall energy demand at a given moment overcomes the maximum power available in the smart home, an energy-related issue is detected. In this case, the component may return to the Chatbot different recommendations for the user, such as shifting the start time of the automation, deactivating one or more of the conflicting automations, or splitting the newly created automation into smaller ones, each consuming less power within a given time interval. If no energy issue is detected, the Energy Checker evaluates alternative start times that take advantage of lower energy prices and may return a suggestion for optimising the automation energy costs.

The Chatbot orchestrates the overall behaviour of the system by interacting with the user and the other components of the architecture. In this way, the user can iteratively refine their initial automation idea with the support of the Chatbot. A complete interaction with the Chatbot will be illustrated in Section 4, where the extended version of the system is described.

3.3.2. Laboratory user study

A laboratory usability study has been carried out with the participation of 10 users (aged 23-56, $M = 32$, $SD = 9.54$). The participants had varying levels of programming experience and were required to complete six tasks related to creating automations. The system was prepared in advance with a given state for each device and a set of predefined automations to create conflict situations during task execution, giving users the opportunity to choose how to cope with emerging issues by interacting with the chatbot. As quantitative data, the percentage of user-accepted and rejected suggestions was measured, as well as the number of modifications made by the users to system automations when conflicts were notified. Direct observation of users and the think-aloud method were adopted to collect qualitative data. After completing all six tasks, each participant was asked to fill out a questionnaire that included the Bot Usability Scale (BUS) (Borsci et al. 2023) and the following open questions:

1. 'Do you think the chatbot is useful?' In case of an affirmative answer: 'In what way do you think it is useful?' In case of a negative answer: 'Why do you think it is not useful?'
2. 'Do you have any suggestions on how the chatbot could be improved?'
3. 'List up to three things you liked about the chatbot.'
4. 'List up to three things you did not like about the chatbot.'

All tasks were completed by all the users. From quantitative data, it was interesting to discover that users accepted the system suggestions in about half of the cases, both for solving conflicting situations and energy-related issues; in particular, they usually preferred to modify existing automations rather than the automation under creation, even when the system suggested a slight modification to this one. Notwithstanding this behaviour, users became more aware of the potential issues and were able to identify their personal, yet successful solutions. This prioritisation of current needs with respect to existing automations may also be due to the nature of the tests, which were not conducted in the wild but rather in a simulated environment.

From direct observation, it was also interesting to notice how several participants expressed a desire to maintain control over device states through automations that not only activate devices but also deactivate them when necessary. Most of them demonstrated awareness of possible hazard situations (e.g. an oven is turned on by an automation, but there is uncertainty about the presence of a human at home); in these cases, they were surprised that the chatbot did not provide an appropriate warning. With respect to the chatbot, users also expected it to know simple information, such as the current day; however, after a first complaint, they were always able to resolve the emerging issues.

The BUS questionnaire yielded positive results for all scales, while the additional open-ended questions enabled the collection of further feedback. Specifically, the chatbot was generally perceived as useful, as it facilitated automation creation for users with limited technical knowledge; participants also appreciated its assistance in optimising energy consumption. As negative aspects, they underlined some delays in the chatbot's responses and inconsistencies in confirmation replies. Users expressed a desire to interact with a more comprehensive interface, including a house map that displays the devices and provides the possibility of interacting with them. They also gave several suggestions for improvement, both for the chatbot and the overall system, including access to real-time data to provide more context-aware recommendations. These new features and improvements, which are also in line with functional requirements (5) and (6) – see Section 3.2, have been integrated into the extended version of the GSH that will be presented in Section 4.

4. Resulting final architecture

The second version of the GSH system represents a substantial evolution of the previous prototype, expanding its scope beyond the conversational automation creation capabilities. While the earlier version primarily focused on enabling users to define automations through a natural language-based interface, the current system is a comprehensive smart home management platform. Users can now monitor and control devices, configure their smart home layout, analyse energy consumption, and manage automations, all within a unified interface. Moreover, the conversational automation creation tool has been enhanced and integrated into the new system. The refinement process was guided by the results of the laboratory user study and addressed the main usability issues and functional limitations that had been identified during the experimentation.

The overall architecture of the new GSH system (see Figure 3) is organised into three main layers: the smart home environment, the backend and core logic layer, and the frontend of the proposed platform. At the foundation lies the smart home, composed of several IoT objects, sensors, smart plugs, and appliances. All of them are connected to the Home Assistant software, which manages their states and executes the created automations. Moreover, Home Assistant provides a comprehensive set of RESTful APIs that external applications, such as the GSH system, can use to retrieve data, send commands, and manage automations. Above this layer, the backend acts as the core of the system, handling all logical operations and coordinating data flow between Home Assistant and the GSH frontend. It gathers device and consumption data, processes analytics, and exposes RESTful endpoints that allow the frontend to access all its functionalities.

At the top, the web frontend provides an intuitive interface that allows users to monitor their smart home, manage connected objects and devices, and configure automations.

4.1. Backend functionalities

The backend implements the core logic of the system architecture, providing all the essential operations that

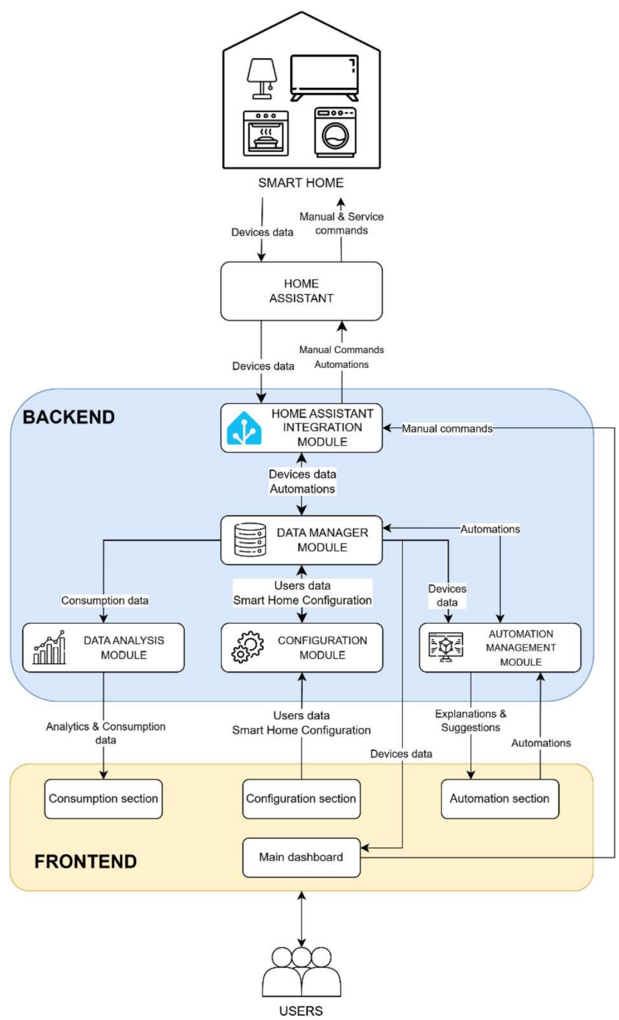

Figure 3. Overall architecture of the integrated system.

enable the smart home functionalities. It follows a modular design, where each module encapsulates a specific set of features and exposes one or more API endpoints, allowing the frontend to interact with the system. The backend comprises five main modules: the Home Assistant Integration Module, the Data Manager Module, the Data Analysis Module, the Configuration Module and the Automation Management Module. These endpoints are mainly implemented using the FastAPI Python framework, which provides efficient request handling for RESTful services. Each endpoint invokes additional libraries to perform specific tasks, such as PyMongo for database management, TensorFlow for deep learning–based energy consumption forecasting, and other specialised packages that support the individual functionalities of each module.

4.1.1. Home assistant integration module

The Home Assistant Integration Module is a crucial component that enables seamless interaction between the system and the smart home environment. It is responsible for managing communication with the Home Assistant (HA) middleware.

The integration module interacts with HA RESTful APIs using the Requests Python library to send queries and commands to HA. Specifically, the module obtains device states, issues commands to smart home devices, and creates and deploys automations on the HA automation engine. Depending on the request, the retrieved information is either returned directly to the calling API endpoint or stored persistently through the features of the Data Manager Module. Hence, the module acts as a middleware layer between the other system modules and the physical smart home.

4.1.2. Data manager module

The Data Manager Module is the component responsible for handling all data utilised within the system. It stores and dispatches various types of information, including the smart home configuration, energy consumption data, device state information, automations, and other related datasets. To accommodate the diverse nature of the data types available in the system, the module implements and orchestrates different data management technologies, optimised for specific requirements, such as user authentication, configuration storage, and time-series data handling. The module seamlessly integrates these heterogeneous storage technologies, internally redirecting each request to the appropriate database based on the endpoint invoked. This approach favours the integration with the other modules, which connects to a consistent data interface.

Moreover, the module interacts directly with the Home Assistant Integration Module to retrieve real-time data from the smart home environment and to deploy automations into the real world. Whenever other modules require access to stored data, they query the Data Manager Module, which provides the requested information from its periodically updated local database. Similarly, any data generated or modified through the frontend that requires persistent storage is routed through this module, ensuring consistent handling, validation, and saving of information within the system.

4.1.3. Configuration module

The Configuration Module consists of a set of functions that enable the frontend to interact with the Data Manager Module for retrieving and storing information related to the system's configuration. This module manages various types of configuration data, including user information and preferences, device attributes such as names, map positions, and operational settings, as well as the household energy plan and its scheduling parameters.

4.1.4. Data Analysis Module

The Data Analysis Module is responsible for processing and analysing the energy consumption data collected and stored by the Data Manager Module. Starting from raw data with a one-minute resolution, the module produces information both at the individual device level and for the overall smart home. Depending on the specific query issued by the frontend, the module retrieves the data from the functions of the Data Manager Module and aggregates them with different levels of granularity, such as hourly, daily, or monthly summaries. In addition to data aggregation, the module integrates deep learning–based forecasting capabilities, which are deployed using the TensorFlow Python library. In particular, it employs pretrained Long Short-Term Memory (LSTM) neural network models to predict future energy consumption (Guizzardi, Barricelli, and Fogli 2025). These models take as input a series of energy values, representing the overall consumption of the previous days, and output a prediction of the expected future consumption for the next 12 hours. Additional factors, such as the time of the day, day of the week, and device activations, are taken into consideration by the model to improve its prediction accuracy.

4.1.5. Automation management module

The Automation Management Module is responsible for verifying the correctness, consistency, and efficiency of the automations before their deployment in the smart home environment. Its primary objective is to detect potential problems such as conflicts between rules, activation chains that may lead to undesired behaviour, and energy-related inefficiencies. This module receives input from the conversational interface, when it is sufficient for the automaton creation, it first generates it in a JSON format compatible with the Home Assistant language syntax, and then invokes two separate procedures, one for the 'conflict check' (Rule Checker) and another for the 'energy check' (Energy Checker). Each procedure returns any identified issues along with possible improvements or corrective actions. Additionally, the chatbot utilises the Data Manager Module to respond to user queries regarding energy consumption.

4.1.5.1. Rule checker. The Rule Checker analyses the set of active automations specified in the Home Assistant format, which is able to specify objects functionalities from different manufacturers, to identify logical issues that may compromise the reliability of the environment. Two types of analyses are performed: conflict detection and activation-chain detection. Conflicts occur when two or more automations simultaneously attempt to control the same device in contradictory ways. The algorithm first inspects the actions associated with each automation and isolates those that operate on the same device. It then verifies whether they perform opposite operations (e.g. one automation switches a device on while another switches it off) or conflicting adjustments (e.g. raising the thermostat temperature while opening a window). The Rule Checker also accounts for cases in which one automation overwrites another without being strictly opposite, such as an automation that turns on all the lights in the house and another that switches off the light in the living room. Once potential conflicts are identified, the system analyses the triggers to ensure that they can indeed occur simultaneously. Automations triggered by opposite events (e.g. 'presence detected' vs. 'no presence detected') are excluded, since they cannot overlap by design. Finally, the conditions associated with the automations are compared: if they are absent, identical, or compatible, the system concludes that a genuine conflict exists and reports it to the user.

The second functionality of the Rule Checker identifies direct and indirect activation chains. Direct chains arise when the action of an automation directly triggers another automation. For instance, a rule that turns on the air conditioner when the temperature exceeds 22 °C may trigger another automation that, when the air conditioner is on, closes the windows. The algorithm can detect such relationships even in the presence of time delays.

Indirect chains occur when the action of an automation alters an environmental parameter that could indirectly trigger another automation. For example, if one automation switches on the heating, it may cause the temperature to rise and inadvertently activate another automation that opens the windows when the temperature exceeds a given threshold. Although such interactions are not guaranteed, the system highlights them to increase user awareness. Loops are treated as a particular case of direct chains, where two or more automations recursively activate each other.

Indirect chains are detected through the *Semantic Component*, which encodes the relationships between devices, their states, and contextual variables (e.g. lighting, temperature, sound pressure, energy consumption). These relationships are stored in JSON files that specify, for each device type and state, the environmental variables that are affected. For example, turning on a light modifies the 'lighting' variable, while activating the heating modifies the 'temperature' variable. A second file links contextual variables to long-term user goals, describing how changes in these variables may support or conflict with those goals under specific fuzzy conditions. By exploiting these semantic descriptions, the Rule Checker can trace the indirect effects of automations on the environment and identify potential activation chains that are not explicitly encoded in the rules.

Finally, when a conflict or an activation chain is detected, an LLM-based Recommendation System (RS) is invoked to generate both an explanation of the problem and possible solutions. The RS receives as input a prompt composed of: (i) the instructions to follow in order to generate solutions to the problem, (ii) the problem expressed as a natural language sentence (e.g. 'The rule {automation1} conflicts with the rule {automation2}'), and (iii) the list of smart objects available in the environment. Two prompt instructions were defined, one tailored to conflict detection and the other to activation chains, ensuring that the system generates context-aware solutions. The output of the RS consists of a structured response that includes: (i) a description of the detected problem, expressed in natural language to ensure clarity for the end-user, and (ii) a set of alternative automations for each problematic automation. These alternatives are designed to preserve the intended functionality of the original automations while eliminating the detected conflict or undesired activation cascade.

4.1.5.2. Energy checker. While the Rule Checker focuses on logical correctness and behavioural consistency, the

Energy Checker addresses the sustainability and feasibility of automations from an energy perspective. Its main purpose is to estimate the energy demand generated by automations, identify potential overloads, and provide users with optimisation strategies aimed at reducing consumption and costs.

The Energy Checker periodically queries Home Assistant's APIs to retrieve the historical activity of devices. For each device, the system receives a record of all state changes within a specified time interval, including the new state and the timestamp of the change. However, HA's historical data is not provided with constant granularity, which complicates the estimation process. To overcome this, the Energy Checker first converts the irregular time series into fixed-duration intervals of one minute. This transformation produces a uniform representation of device behaviour over time, allowing for more consistent energy analysis.

Once the device history has been converted into minute-long blocks, the Energy Checker computes the average power demand and average duration of each device state. The method used to estimate energy usage depends on the type of device:

- Devices with real-time monitoring (e.g. smart plugs): these devices can directly provide instantaneous power values. For each state, the system calculates the average power by summing all recorded values and dividing the result by the duration of the state (in minutes).
- Devices without real-time monitoring (e.g. smart speakers): in this case, the system relies on a manual mapping, where each possible device state is associated with a static power value. These values are typically derived from user manuals or external reference datasets. For this category, the average duration of each state is estimated by computing the total time spent in that state and dividing it by the number of activations.

This dual approach ensures that both monitored and non-monitored devices can be integrated into the analysis.

The consumption data extracted from device histories is continuously integrated with the database to refine long-term estimates. For each device–state pair, if historical data already exists, new averages are computed using a moving average formula. This mechanism incrementally updates both power and duration values by combining past records with new observations: if no prior data exists for a given device–state pair, the newly computed averages are simply inserted into the database. In this way, the system progressively builds house-specific consumption profiles, which increase the accuracy of subsequent predictions.

To estimate the overall energy impact of new automations, the Energy Checker builds a State Matrix, a three-dimensional structure of size $7 \times 1440 \times \#\text{devices}$. This matrix represents, for each day of the week (7), each minute of the day (1440), and each device, the expected state of the device as determined by both historical usage and saved automations. The matrix is populated by processing all automations in chronological order: for each automation, the system identifies the days it applies to, the target devices, and their intended states. It then queries the database for the average duration of each state and updates the corresponding cells in the matrix to reflect the expected device behaviour. The State Matrix thus offers a minute-level projection of the smart home's energy usage across the week.

By analysing the State Matrix, the Energy Checker identifies energy issues, i.e. time windows in which the aggregated power demand of all active devices exceeds a predefined threshold (typically set at 3 kW for domestic households in Italy). When such overloads are detected, the system reports the problematic intervals to the user, highlighting the devices and automations responsible for the excessive demand.

In addition to conflict detection, the Energy Checker provides optimisation strategies aimed at reducing energy costs. Once the original cost of an automation is calculated (based on the time and power demand of its actions), the system explores alternative schedules by shifting the start time of the automation. This optimisation process evaluates different temporal configurations to identify those that would reduce overall energy consumption or exploit cheaper tariff windows, without introducing new conflicts with existing automations. The final output is a ranked list of suggested rescheduling options, each presented to the user along with the estimated savings.

4.2. Frontend design and functionalities

The frontend of the system is a web application accessible through any modern web browser. The application is implemented using the React JavaScript framework, while the Laravel PHP framework is employed on the server side for user management and authentication. Users can log into the application using their email address and password. If the login is successful, the system redirects the user to either the home page or the initial configuration page, depending on whether a smart home setup is already present. If no configuration is available, a wizard is presented, allowing inhabitants to configure the planimetry of the smart home, selecting

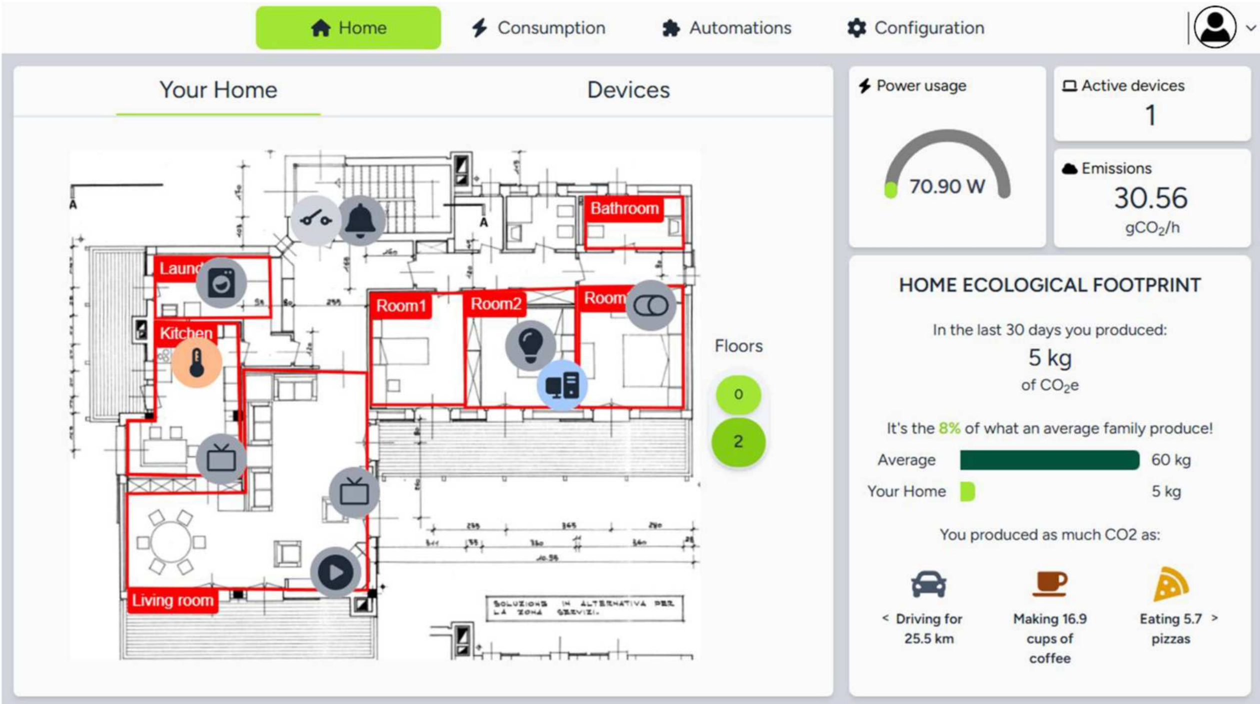

Figure 4. The main dashboard of the GSH platform 22.

the set of smart devices to use and specifying data related to energy cost and its availability.

4.2.1. Main dashboard

The main dashboard (Figure 4), which provides access to the system's functionalities, is divided into two main sections. On the left, a map of the smart home is displayed, illustrating the layout and associated devices. Here, users can select a specific device from the map to access its control panel. From the control panel, it is possible to send actions to the devices, hence changing their current state. The set of available actions varies depending on the device type. For example, a smart plug allows only basic control functions, such as turning it on or off, whereas a smart light provides more advanced options, including brightness and colour adjustments. From the same panel, it is also possible to obtain information about the device's real-time state, its current energy consumption, and other device-specific quantities. On the right side, the home page displays information concerning the overall state of the house and its environmental impact. At the top, three widgets respectively report the current total energy consumption, the number of active devices, and the associated CO_2 emissions. Below, a card titled 'Home Ecological Footprint' presents the household's CO_2 emissions over the past 30 days, comparing them to the average Italian household. Additionally, it translates the emissions into more intuitive metrics, such as the equivalent kilometres driven by a car, the number of smartphone charges, or the time required for a tree to absorb an equivalent amount of CO_2.

4.2.2. Consumption section

Through the Consumption section, shown in Figure 5, users can get insights into the smart home's energy usage patterns. The page is organised into three tabs. The 'Total energy consumption' tab enables users to visualise energy consumption data for the entire household or for individual devices over a specified time range, with hourly, daily, or monthly granularity. The 'Predicted consumption' tab provides a short-term forecast of energy usage for the upcoming hours, based on consumption data from previous days. Finally, the 'Consumption comparison' tab allows users to perform side-by-side analyses of energy consumption across different days or months, choosing either the entire home or a specific device as the object of the comparison. All data are provided using bar charts.

4.2.3. Automation section

The Automation section provides users with tools to manage and create smart home automations. The interface, presented in Figure 6, is divided into two main parts. On the left, a list of all saved automations is displayed, while on the right, the chat-based interface for creating automations appears. Its interface replicates that of the previous prototype described in (Barricelli

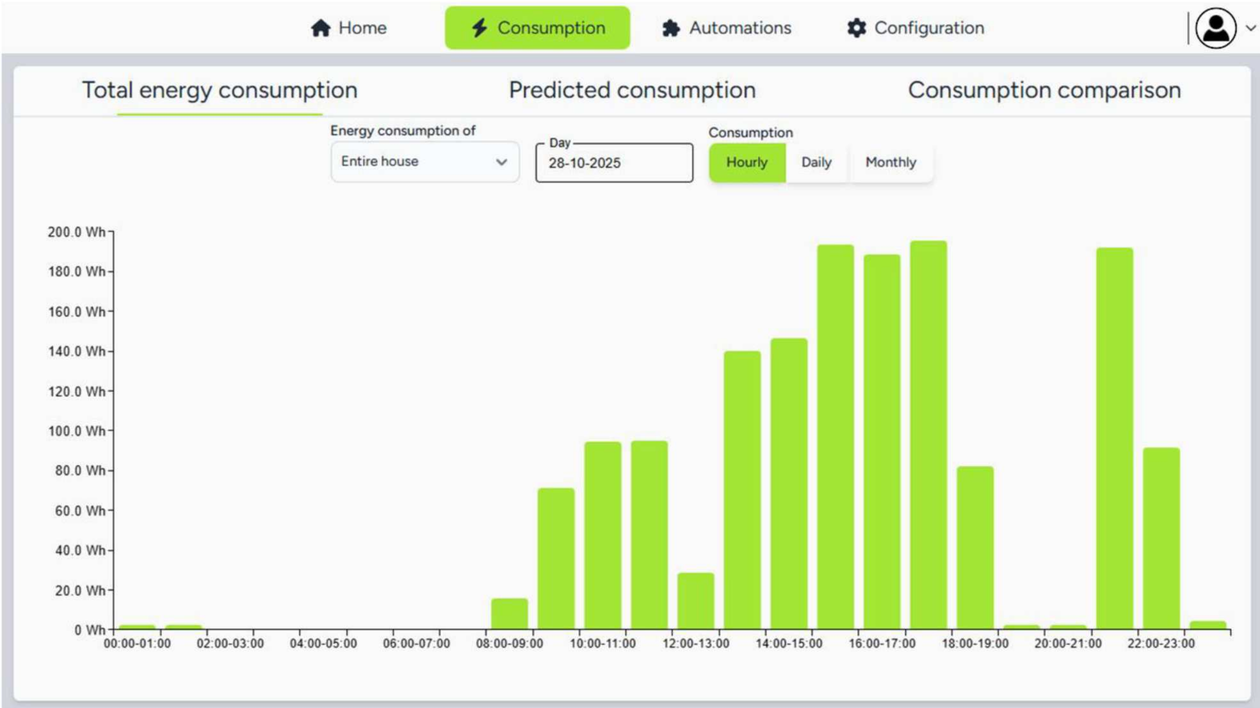

Figure 5. Consumption section of the application. It shows the total energy consumption graph with the hourly view.

et al. 2025). In the example of Figure 6, the user asked for 'an automation that turns off the air conditioner in the living room when it consumes too much', and the chatbot has replied with a suitable automation that triggers when the power of the air conditioner exceeds 1000W for 10 min straight. Before saving the automation, a summary of its elements is presented. Each automation consists of three key components: the event(s) that initiate(s) the automation; an optional set of conditions, which specify the constraints that must be satisfied for the automation to execute; and a set of actions, which determine the operations to be performed once the automation is triggered.

Other than during the creation process, users can access these automation details from a side drawer that opens when an automation is selected from the

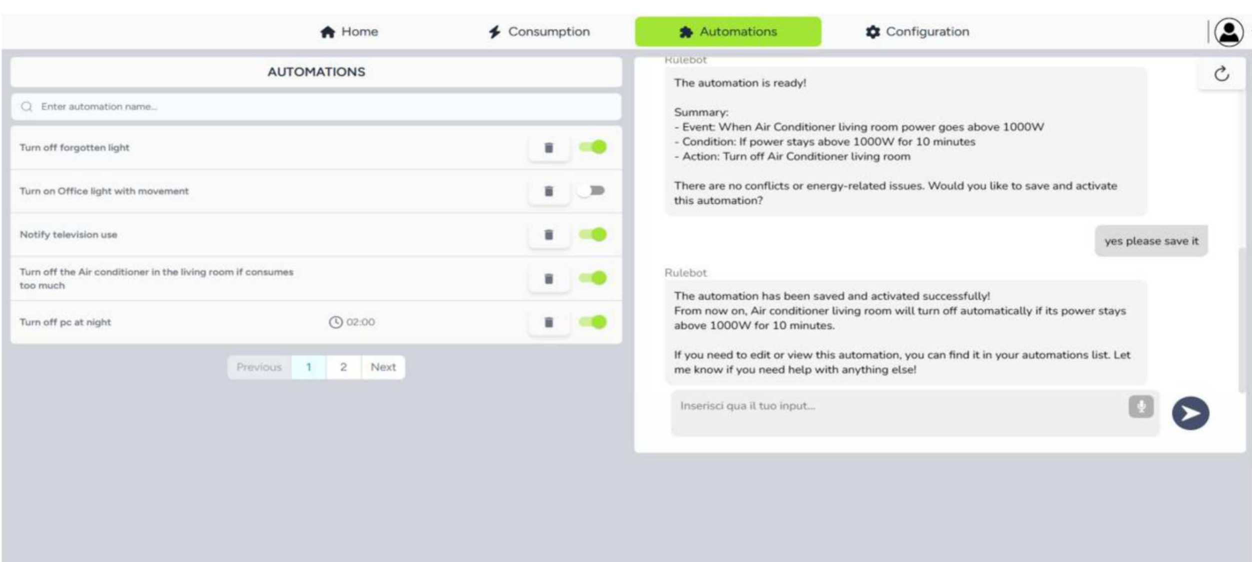

Figure 6. Automation section: on the left, the list of automations; on the right, the chat with the automation just created, and the results of the Rule Checker and Energy Checker.

left-hand list. Additionally, information on the monthly cost, power usage, and energy consumption associated with the automation is shown. Lastly, the user is provided with a set of ‘green suggestions’, offering alternative scheduling options that can further optimise energy efficiency and reduce automation costs.

More specifically, Figure 7 shows the interaction flow for creating a new automation. It begins with the user describing the desired automation in natural language through the chat interface on the right. There is no particular constraint on what the user can say. The chatbot interprets the request and asks for additional details if necessary. This is possible because the system prompt informed it about the target format of the automation to create: event, condition(s), action(s) with an example. For instance, if the user requests an automation that turns on the heating ‘when it is cold’, the chatbot will ask for the temperature below which the automation activates, e.g. 14 °C. After getting all the necessary details, the system then summarises the automation proposal and requests user confirmation.

Once an initial definition is confirmed, the chatbot generates the automation in JSON format and initiates the verification phase to ensure it does not conflict with the existing set of automations. It first queries the Rule Checker to detect potential conflicts arising from overlapping actions, interdependencies, or cascading activations. If such conflicts are identified, the Rule Feedback component proposes possible resolutions, and the user is tasked with adjusting the automation to resolve these conflicts.

The revised automation is subsequently evaluated from an energy consumption perspective. The system queries the Energy Checker, which analyses the automation’s potential impact and assesses whether the automation may result in excessive energy usage or inefficient scheduling. If issues are detected, the Energy Feedback component presents alternative scheduling options to reduce energy costs or prevent electrical overloads.

Throughout the process, the chatbot autonomously determines which verification steps to execute, ensuring

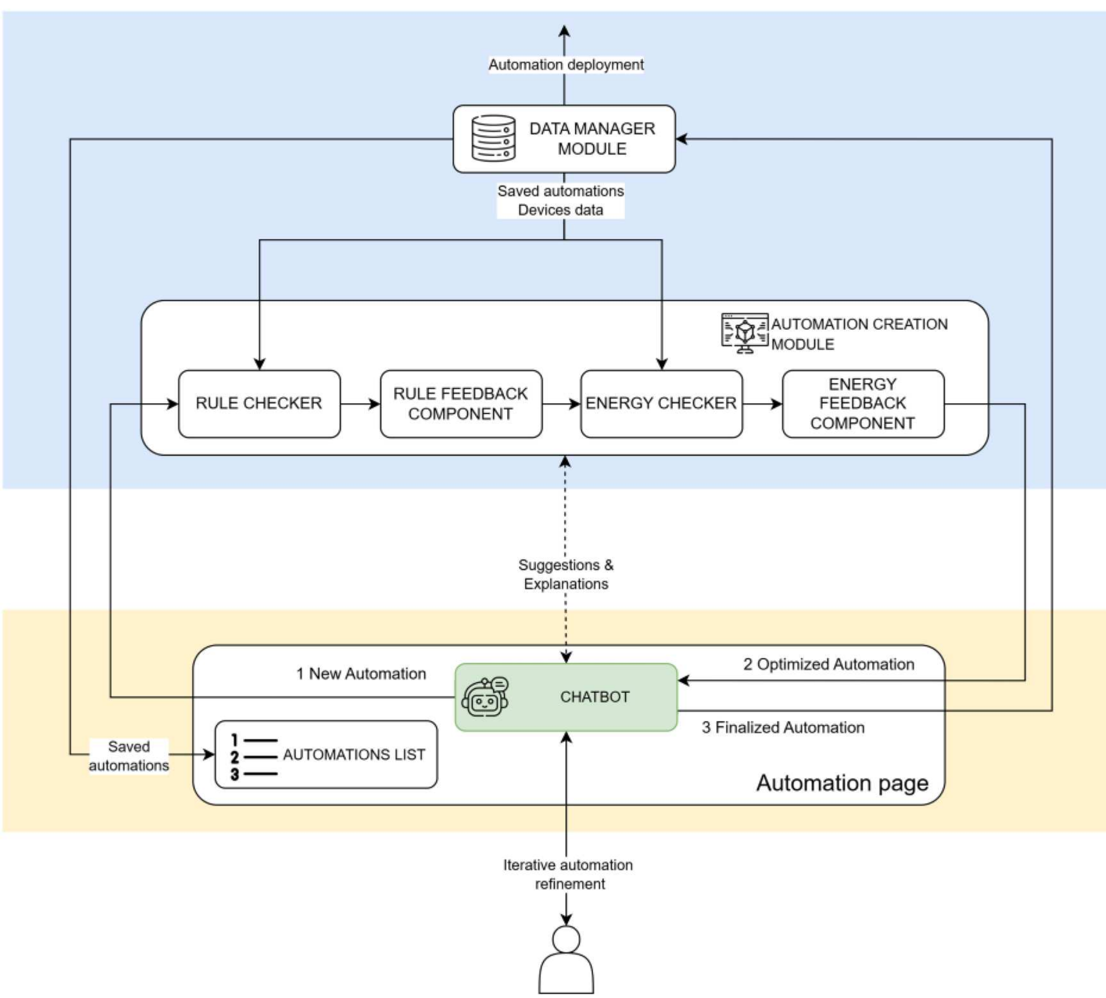

Figure 7. Interaction flow for the creation of a new automation.

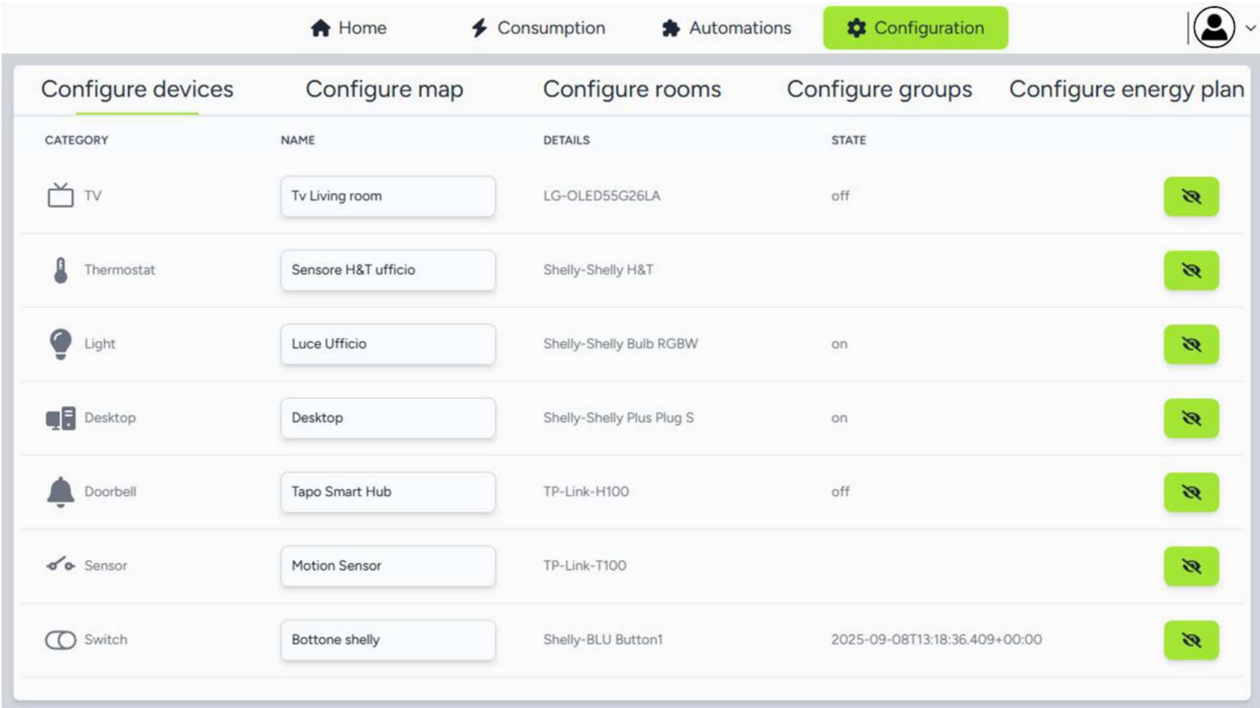

Figure 8. Configuration section: the 'Configure devices' tab is selected.

that all generated automations comply with both user preferences and system constraints. By providing suggestions rather than imposing changes, the system preserves user control over decision-making while offering structured, context-aware guidance. Once all refinements are completed, the finalised automation is stored in the system database and integrated into Home Assistant for potential execution.

4.2.4. Configuration section

The Configuration section shown in Figure 8 allows users to modify their smart home setup. This section is organised into five tabs, each dedicated to a specific aspect of system customisation. The 'Configure devices' tab enables users to select which devices to include, as well as to modify their names and icons displayed on the map. The list of the available devices, along with their details and real-time states, are provided once again by the HA API. At first, all the devices configured in HA are included in the GSH system. Then, the user can choose to ignore specific devices by interacting with the crossedeye button that appears at the end of the device row (Figure 8). The 'Configure map' tab allows users to drag and drop the configured devices to their actual physical locations within the house. The 'Configure rooms' tab provides tools for drawing rooms directly on the map and assigning each a descriptive label. The 'Configure group' tab enables the creation of device groups, allowing users to assign one or more devices to each group for easier management. Finally, the 'Configure energy plan' tab permits users to define the system's maximum admissible power and specify the energy cost schedule throughout the week.

5. The autoethnographic study

We conducted an autoethnographic study in the homes of six of the project team members (participants P1-P6). The primary objective was to evaluate the use of the GSH system in everyday real life, before conducting an in-the-wild evaluation.

The testing period started on July 24th and ended on October 20th, 2025. The participants were asked to use the GSH system in their own homes, connecting their smart devices and other project equipment that was installed for the study. To keep track of their experience, the participants kept a diary by means of a Google Form. Their participation varied in terms of time range and number of entries recorded (see Table 3). At the end of the study, 50 entries were recorded.

5.1. The setting

The connected devices were chosen from different brands and models, to reflect the heterogeneity that typically characterises real home environments. The six home configurations are detailed in Table 4.

Table 3. Time range of participation and number of diary entries for each participant in the study.

Participant	Time range	Entries
P1	12/08/2025–19/09/2025	17
P2	31/08/2025–20/10/2025	5
P3	24/07/2025–01/09/2025	6
P4	05/08/2025–01/09/2025	12
P5	25/08/2025–05/10/2025	6
P6	31/07/2025–15/09/2025	4

Table 4. The participants' home configuration.

Participant	Floors	Sensors and objects
P1	1	3 smart lights, 1 smart TV, 1 humidity and temperature sensor, 1 motion sensor, 5 smart plugs (PC, hair dryer, toothbrush, Amazon Echo Show, washing machine), 1 additional smart plug used alternately for an iron or vacuum, 3 air conditioning units, and 1 smartphone
P2	2	5 smart lights, 1 smart TV, 1 humidity and temperature sensor, 1 motion sensor, 3 smart plugs (Wi-Fi modem, Amazon Echo Dot, TV), 2 Amazon Echo Dots, 1 air purifier, 1 presence sensor, 1 door sensor, 1 bed sensor, 1 weight scale, 1 gas detector, and 1 smartphone
P3	2	1 smart button, 1 smart light, 3 smart plugs (TV, PC, washing machine), 1 smart TV, 1 Google Home Mini, 1 humidity and temperature sensor, 1 motion sensor
P4	1	1 humidity and temperature sensor, 1 motion sensor, 1 gas detector, 1 presence sensor, 1 smart TV, 1 door/window sensor, 1 air quality monitor, 1 smart cam, 1 Amazon Alexa, 1 air purifier, 2 smart plugs (washing machine, fan), 5 smart lights, 1 smartphone
P5	1	3 door/window sensors, 1 air quality monitor, 1 air purifier, 2 smart lights, 1 humidity and temperature sensor, 2 presence sensors, 2 smart plugs (dehumidifier, microwave)
P6	1	1 air purifier, 6 smart lights, 1 smart TV, 2 Amazon Alexa, 1 presence sensor, 1 door sensor, 5 motion sensors, 2 smart plugs (air conditioner, microwave), 1 temperature sensor, 1 air quality monitor sensor, 1 noise sensor, 1 smartphone

5.2. Results

5.2.1. Data collected through the diaries

The first question in each diary entry measured the perceived usefulness of the Digital Twin on a five-point scale, from 1 (lowest) to 5 (highest). The answers ranged between 2 and 5, with a mean of 3.86 (Median: 4, SD: 0.88).

The second question of the diary was 'Did you create any new automations?'. Out of the 50 entries, 34 reported the creation of an automation (68%), and when asked to assign a score to the experience, on a scale from 1 to 5, the participants provided only scores ranged between 3 and 5; specifically, 52.94% responded with '4', 35.29% with '5', and 11.76% with '3').

The participants who created new automations were required to indicate if any conflicts among the new and the existing automations were detected by the system. Here, the participant could answer 'Yes', 'No', or 'I do not know'. Half of the entries report no conflict (17 out of 34), 16 reported conflicts, and 1 reported that the participant was unsure about it. To the authors of the 16 entries that reported that conflicts arose during the creation process, it was asked what kind of conflicts were detected and how they were resolved. In some entries, more than one conflict had been described. All the described situations belong to one of these three cases: 1) the conflict was real, and the system suggested an efficient solution; 2) the conflict was real but not blocking, so the user could ignore it and proceed with saving the new automation; 3) the conflict was non-existent.

To the participants who reported creating automations (34 entries), it was asked if the system detected any issues related to energy consumption. Only one participant answered positively in one entry. Specifically, the automation was about scheduling the air conditioning and the washing machine to start at the same time; the system notified a potential conflict and suggested delaying one of the two actions by at least 30 min, a suggestion that the participant accepted, resolving the issue.

All participants were asked if there was any automation they would like to create but have not been able to. Out of the 50 entries, 14 answered positively (28%), and when asked about what kind of automations they could not create, their answers were:

- Automations on Withings sensors (2 occurrences).
- Automations on a device as the result of an event detected by another device (2 occurrences) – e.g. to have a light be turned on at sunset but only if motion is detected.
- Automation that sends a notification on a smartphone, via HA app, at a certain time (1 occurrence).
- Automation that uses the number of Watts detected via a smart plug to determine if the washing machine programme is complete (1 occurrence).
- Automations with Alexa to receive personalised messages or to switch off the Echo Show screen when nobody is at home (5 occurrences).
- Automation that turns off a light in case motion is detected and it is after 23:00 (1 occurrence).
- Automation that at the same time specifies when to turn on a device and when to turn it off (2 occurrences).

To investigate the awareness of the participants towards the system's potential, we asked if they noticed the activation of any automations. Forty-eight entries (96%) had a positive response; with the following two questions to the respondents, we found out that all of them were aware of all automations that were activated and only 5 times the participants pointed out that the effects were not those that they expected:

- Sensors that get disconnected interrupted an event detection, leading automations to fail.
- A device that was expected to be turned on and was not.
- An automation did not activate because its trigger was unable to express that something needed to happen only at night.
- An automation for a specific device can be created via the device's proprietary app, but not via GSH.
- An automation was not expected to be executed because the related device was not active.

The diary also investigated whether participants used the Digital Twin to check on the energy consumption of their home. The majority of the entries (28) reported a positive answer, and in most of them, their authors declared that they checked both the consumption of the entire home and of specific devices (13 out of 28); in the other entries, participants indicated that they checked the consumption of the entire home (8 out of 28), or just the one of specific devices (7 out of 28).

We further explored how this feature of the Digital Twin helped the participants:

- To better understand the impact of their smart home on the environment (11 occurrences).
- To better understand the reasons behind the cost of the energy bill (9 occurrences).
- To better understand the energy consumption of a specific device (7 occurrences).
- To make decisions about the status of a device – i.e. to leave it switched on or turn it off (4 occurrences).
- To check if a device was correctly turning off (2 occurrences).
- To understand the energy consumption of their home (1 occurrence).
- To predict the energy consumption (1 occurrence).

Only one entry reported that the experience did not bring any information to the participant.

5.2.2. Thematic analysis of users' comments

Each diary's entry ended with asking the participants to leave a comment on their experience, if they liked to do so. A total of 29 comments were received, on which a thematic analysis was performed. Two researchers (one not participating in the study) independently coded the comments and met to discuss and refine the codes. After a negotiation of conceptual boundaries, two codes were merged into one and consensus was reached. The code stability was reached rapidly by the researchers, and no new themes were proposed after the first review round. Codes were finally discussed with the study participants to verify the correct interpretation of their comments. The thematic analysis yielded five themes that represent users' perceptions, breakdowns, and improvement opportunities. For each theme, one or more sub-themes were identified. They are described in the following, providing one or more user quotations (between square brackets).

- **Theme 1: Conversational agent**

○ **Positive conversational interaction:** Users find interactions with the chatbot intuitive and effective. ['*The interaction was really surprising, it understood what I wanted to do without telling it everything!*'. '*I created the automation very quickly (chatbot understood instantly).*']

○ **Anthropomorphisation of AI-based assistant:** The chatbot gives the impression of interacting with a human, offering guidance, warnings, and suggestions in a human-like way. ['*It really feels like talking to a person!*']

- **Theme 2: Technical problems or limitations**

○ **Dependency on an Internet connection and Home Assistant:** A stable internet connection is essential; temporary disconnections can disrupt automations or device monitoring. ['*There can be problems if the internet connection is not stable.*']

○ **Limitations of battery-powered devices:** Devices running on batteries may enter power-saving mode, making them intermittently unavailable and complicating automation execution. ['*Creating automations that involve battery-powered devices is a bit problematic because these devices often go into energy-saving mode and are not always reachable. For example, in the case of the smart button, it happens sometimes that you must press the button several times because the first press just wakes up the device and has no effect on the automations saved on the system.*']

○ **Insecurity due to lack of (or late) feedback:** Delayed or missing feedback on automation execution creates uncertainty, sometimes causing users to repeat commands unnecessarily. ['*Especially regarding device activations, if the request takes too long, you risk sending the command again because you're afraid the first one didn't work.*']

○ **System malfunctions and technical inconsistencies:** Users encounter errors in saving automations, inaccurate device status reporting, and occasional failures in voice interactions. ['*I asked to change the time of the automation that turned on air conditioner in the living room, but even if everything went fine, it gave a* *save error and in fact the automation then didn't activate.' 'Voice interaction wasn't working.'*]

○ **Issues with conflict detection:** Chatbot's conflict detection can be overly cautious or fail to account for existing constraints, requiring user clarification. [*'I asked to modify the automation just created to turn on the bedside light on the weekend. The chatbot rightly asked me if I wanted to modify the time. I said 6:45 (before it was 6:30) but the chatbot found a conflict because it didn't consider that the automation was only for Saturday and Sunday, so I had to specify it again.'*]

○ **Issues with device integration:** Certain devices, such as Amazon smart speakers (Echo and Echo Show) with Alexa virtual assistant, or smartphones like iPhones, cannot fully participate in automations, limiting overall system flexibility. [*'The system wasn't able to modify the text of an Echo message in an automation.' 'I also tried to create automations that would send me a notification in the Home Assistant app on iPhone, but I wasn't able to save them (it gave an error when saving), so I couldn't try them.'*]

- **Theme 3: Energy consumption**

○ **Energy consumption awareness:** Users gain insights into energy use, noticing differences between devices and reflecting on consumption patterns, though questions about accuracy sometimes arise. [*'Even though the recorded costs are negligible, it's helping me understand the difference in consumption between energy-intensive and non-energy-intensive devices.'*]

- **Theme 4: Automations**

○ **Positive experience:** Automations generally function as intended and are valued by users. Moreover, the users show an appreciation for the whole creation and the user experience. [*'With the long cycle, the automation that signals the end of the washing machine's work with an alarm worked correctly and was useful (I had forgotten I left the washing machine on).'*]

○ **Technology acceptance:** Users adapt quickly to automations and express interest in expanding their setups with more sensors and connected devices. [*'After more than a month, I've gotten used to the automations that turn the lights on and off, and I find them very convenient.'. 'I want to add more sensors and connected devices so I can create more automations.'*]

○ **Limited trust:** Despite automation functionality, users sometimes hesitate to rely fully on the system, manually intervening or questioning the data provided. [*'I then noticed that it shows consumption and other information about the created automation; it's very interesting but is it accurate?'. 'The solutions the chatbot proposed to resolve conflicts are reasonable, but since I wasn't sure it could implement them, I didn't accept them.'*]

○ **Mismatch between user's mental model and systems' conceptual model:** Users identify minor technical limitations, such as misconfigured events or triggers, but can usually correct them easily. [*'I realized that the automation to turn on the air conditioning had the event and condition swapped, which is why it didn't start.'*].

- **Theme 5: New features and improvements**

○ **Editing of automations:** Users appreciate the ability to disable or delete automations and want more flexibility to modify events, conditions, and actions. [*'Another thing I'd like to mention is that it's very useful to be able to disable or delete automations through the DT, but it would be equally useful to modify events, conditions, and actions (e.g. change an activation time).'*]

○ **Visualisation:** The current display of automations lacks clarity, making details difficult to interpret despite clear naming. [*'The way automations are displayed on the DT is a bit confusing because it doesn't show the details clearly (even though the automation's name makes it clear).'*]

○ **Energy consumption and cost:** Users desire comparative views of energy usage and associated costs across devices, not just over time but also between different devices. [*'I would like to know, in addition to the energy consumed, also the cost of that consumption.'*]

○ **Access to further functionalities**: the use of the platform also stimulated new ideas for additional functionalities, such as connecting services provided by Alexa in new automations [*'I wanted to activate relaxing music on Amazon Alexa when I wake up, but it was not possible.'*]

5.3. Discussion

The results obtained from the diary analysis show that the experience with the system encouraged users to create numerous automations, actively experimenting with the features offered by the GSH system. However, this engagement has sometimes been affected by technical issues or structural limitations that have prevented the creation of additional desired automations. The most common difficulties encountered by users were related to the integration of some functionalities of connected devices, such as smartphones and smart speakers, as well as limitations on expressing more complex triggers for the automations. This suggests a strong interest among participants in greater expressiveness for automations and can be interpreted as a positive sign of technology appropriation.

The frequent detection of conflicts, both real and potential, shows that the preventive analysis mechanism is effective but sometimes perceived as overly cautious.

An automatic system tends to consider all possible cases in an exhaustive way, even those that are not particularly relevant, thus generating messages that may be perceived as excessive. At the same time, users generally adopted a collaborative attitude toward the system, often reporting that they understood and accepted the proposed solutions when relevant. Nonetheless, the abundance of alerts and confirmation prompts sometimes negatively affected the interactive experience; this might lead to a reflection on how to effectively balance safety with efficiency in this case, also considering the possible learning capabilities of the system.

The results concerning energy-related conflict detection show that during the study, the system identified only one potential overload. This scarcity is likely linked to the absence of energy-intensive appliances in the participants' homes connected to the GSH system. Overall, the system's energy-monitoring capabilities were perceived as useful and non-intrusive, achieving the objective to enhance consumption awareness and foster more sustainable behaviours.

The interaction with the chatbot has been widely positively evaluated. Users emphasised that the conversational nature of the system makes creating automations intuitive, fast, and similar to talking to a person. This anthropomorphisation element appears to be a key factor in technology acceptance, as it increases trust and reduces the perception of task complexity.

The improvements suggested in the diaries indicate that participants do not perceive the system as unreliable, but rather as promising and still in the process of evolving. The constructive nature of the comments and proposals reflects an active involvement and interest in the future use of the GSH system. Users seem willing to invest time in using the system because they see the tangible value of automations in their everyday lives.

Despite overall satisfaction with the conversational interaction, several comments suggest that having a clear visual representation of automations could be helpful. The current conversational interface is, in fact, not always easy to use for quickly understanding the structure of an automation. Users seem to prefer a hybrid approach: keeping the chatbot for automation creation and modification, and complementing it with a visual interface that clearly displays it. This suggestion opens up further system developments that the research team has already begun to consider.

The autoethnographic nature of the study, involving individuals who were also part of the GSH system's design process, introduces risks of confirmation and interpretative bias. The willingness and engagement observed over the study time cannot be taken for granted, as participants' attitudes may not reflect those of external users. The participants, in fact, had prior knowledge of the system that may have shaped their evaluation. Their involvement in the design process may also have increased their tolerance for technical issues and prompted a problem-solving interest that may not arise in non-expert users. While these limitations affect the external validity of the findings, the study was deliberately framed as a phase aimed at stress-testing the system in real houses to verify its scalability and maturity before conducting *in-the-wild* evaluations with independent participants.

6. Conclusions and future work

In this paper, we presented the design, implementation, and evaluation of a platform for supporting green smart homes. It provides a digital twin of a home equipped with several connected sensors, objects, and devices, which can deliver services to users, even those without programming experience, to manage and control the automations connecting the objects and their associated energy consumption in an integrated manner. The process of creating automations is facilitated by a conversational agent based on an LLM.

After a laboratory evaluation of the initial version, the platform has been improved and installed in six real homes for use over a longer period in various contexts. This provided us with the opportunity to reflect on the potential of the proposed approach and validate the maturity and scalability of the new version. The trials yielded positive and encouraging feedback, showing the ability of the platform to function effectively in real-world environments, and provide useful support in controlling and managing objects and devices, taking into account also their energy consumption.

Future work will be dedicated to expanding the set of services to support sustainability in homes, such as food and waste management, and the installation of the GSH platform in a broader range of locations for extended periods.

Author contributions

CRediT: **Barbara Rita Barricelli:** Conceptualization, Data curation, Formal analysis, Methodology, Writing – original draft; **Daniela Fogli:** Conceptualization, Funding acquisition, Investigation, Methodology, Project administration, Writing – original draft; **Simone Gallo:** Investigation, Methodology, Software, Writing – original draft; **Davide Guizzardi:** Investigation, Methodology, Software, Writing – original draft; **Sara Maenza:** Investigation, Methodology, Software, Writing – original draft; **Andrea Mattioli:** Methodology, Software, Writing – original draft; **Fabio Paternò:** Conceptualization, Funding acquisition, Investigation, Methodology, Project administration, Supervision, Writing – original draft.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

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

ORCID

Barbara Rita Barricelli http://orcid.org/0000-0001-9575-5542
Daniela Fogli http://orcid.org/0000-0003-1479-2240
Simone Gallo http://orcid.org/0000-0002-5162-0475
Davide Guizzardi http://orcid.org/0009-0000-7761-7103
Sara Maenza http://orcid.org/0009-0000-2205-6599
Andrea Mattioli http://orcid.org/0000-0001-6766-7916
Fabio Paternò http://orcid.org/0000-0001-8355-6909

References

Aagaard, L. K., T. H. Christensen, and K. Gram-Hanssen. 2023. “My Smart Home: An Auto-ethnography of Learning to Live with Smart Technologies.” *Personal and Ubiquitous Computing* 27:2121–2131. https://doi.org/10.1007/s00779-023-01725-0.

Alan, A. T., E. Costanza, S. D. Ramchurn, J. Fischer, T. Rodden, and N. R. Jennings. 2016. “Tariff Agent: Interacting with a Future Smart Energy System at Home.” *ACM Transactions on Computer-Human Interaction* 23: 1–28. https://doi.org/10.1145/2943770.

Andrao, M., F. Gini, F. Greco, A. Cappelletti, G. Desolda, B. Treccani, and M. Zancanaro. 2025. ““React”, “Command”, or “Instruct”? Teachers Mental Models on end-User Development.” In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems, (CHI ‘25)*, edited by Naomi Yamashita, Vanessa Evers, Koji Yatani, Xianghua (Sharon) Ding, Bongshin Lee, Marshini Chetty, and Phoebe Toups-Dugas, 1–15. New York, NY: Association for Computing Machinery. https://doi.org/10.1145/3706598.3713234.

Barricelli, B. R., E. Casiraghi, and D. Fogli. 2019. “A Survey on Digital Twin: Definitions, Characteristics, Applications, and Design Implications.” *IEEE Access* 7:167653–167671. https://doi.org/10.1109/ACCESS.2019.2953499.

Barricelli, B. R., D. Fogli, S. Gallo, D. Guizzardi, S. Maenza, A. Mattioli, and F. Paternò. 2025. “An EUD Approach to Creating Feasible and Energy-Saving Automations for Smart Homes.” In *End-User Development*, edited by C. Santoro, A. Schmidt, M. Matera, and A. Bellucci, 3–21. Cham: Springer Nature. https://doi.org/10.1007/978-3-031-95452-8_1.

Barricelli, B. R., D. Fogli, S. Gallo, D. Guizzardi, A. Mattioli, and F. Paternò. 2024. What People Think About Green Smart Homes, in: CEUR Workshop Proceedings.

Beheshtian, N., S. Moradi, A. Ahtinen, K. Väänanen, K. Kähkonen, and M. Laine. 2020. “GreenLife: a persuasive social robot to enhance the sustainable behavior in shared living spaces.” In *Proceedings of the 11th nordic conference on human-computer interaction: Shaping experiences, shaping society (NordiCHI ’20)*, edited by Ilja Šmorgun and Gerd Berget, 1–12. Tallinn: Association for Computing Machinery. https://doi.org/10.1145/3419249.3420143.

Borsci, S., E. Prati, A. Malizia, M. Schmettow, A. Chamberlain, and S. Federici. 2023. “Ciao AI: The Italian Adaptation and Validation of the Chatbot Usability Scale.” *Personal and Ubiquitous Computing* 27:2161–2170. https://doi.org/10.1007/s00779-023-01731-2.

Brich, J., M. Walch, M. Rietzler, M. Weber, and F. Schaub. 2017. “Exploring end User Programming Needs in Home Automation.” *ACM Transactions on Computer-Human Interaction* 24: 1–35. https://doi.org/10.1145/3057858.

Bu, L., W. Xiong, C. J. M. Liang, S. Han, D. Zhang, S. Lin, and X. Li. 2018. “Systematically Ensuring the Confidence of Real-Time Home Automation IoT Systems.” *ACM Transactions on Cyber-Physical Systems* 2: 1–23. https://doi.org/10.1145/3185501.

Cecchinato, M. E., A. L. Cox, and J. Bird. 2017. “Always on(Line)? User Experience of Smartwatches and Their Role within Multi-device Ecologies.” In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, Chi ‘17*, edited by Cliff Lampe, Juan Pablo Hourcade, Caroline Appert, and Daniel Wigdor, 3557–3568. Denver, CO: Association for Computing Machinery. https://doi.org/10.1145/3025453.3025538.

Chaki, D., and A. Bouguettaya. 2021. “Adaptive priority-based conflict resolution of IoT services.” 2021 IEEE international conference on web services (ICWS), Chicago, IL, USA. https://doi.org/10.1109/ICWS53863.2021.00091.

Chang, H. 2016. *Autoethnography as Method* (1st ed.). New York, NY: Routledge. https://doi.org/10.4324/9781315433370.

Chawla, D., and S. Jain. 2024. “Smart Trigger Alert and Alarm Control System for Retail Store Using IoT.” Proceedings of the International Conference on Innovative Computing & Communication (ICICC 2024). https://doi.org/10.2139/ssrn.4902505.

Chen, X., X. Zhang, M. Elliot, X. Wang, and F. Wang. 2022. “Fix the Leaking Tap: A Survey of Trigger-Action Programming (TAP) Security Issues, Detection Techniques and Solutions.” *Computers & Security* 120: 1–18. https://doi.org/10.1016/j.cose.2022.102812.

Cheng, X., and L. Zhang. 2025. “Inspiration Booster or Creative Fixation? The Dual Mechanisms of LLMs in Shaping Individual Creativity in Tasks of Different Complexity.” *Humanities and Social Sciences Communications* 12:1563. https://doi.org/10.1057/s41599-025-05867-9.

Coppers, S., D. Vanacken, and K. Luyten. 2020. “FORTNIoT: Intelligible Predictions to Improve User Understanding of Smart Home Behavior.” *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4: 1–24. https://doi.org/10.1145/3432225.

Coppers, S., D. Vanacken, and K. Luyten. 2022. "FortClash: Predicting and Mediating Unintended Behavior in Home Automation." *Proceedings of the ACM on Human-Computer Interaction* 6: 1–20. https://doi.org/10.1145/3532204.

Corno, F., L. De Russis, and A. Monge Roffarello. 2019. "Empowering end Users in Debugging Trigger-Action Rules." In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, Chi '19*, edited by Anna Cox and Vassilis Kostakos, 1–13. Glasgow: Association for Computing Machinery. https://doi.org/10.1145/3290605.3300618.

Costanza, E., S. D. Ramchurn, and N. R. Jennings. 2012. "Understanding Domestic Energy Consumption through Interactive Visualisation: A Field Study." In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing, UbiComp '12*, edited by Hao-Hua Chu and Gillian Hayes, 216–225. Pittsburgh, PA: Association for Computing Machinery. https://doi.org/10.1145/2370216.2370251.

Cotti, L., D. Guizzardi, B. R. Barricelli, and D. Fogli. 2024. "Enabling end-User Development in Smart Homes: A Machine Learning-Powered Digital Twin for Energy Efficient Management." *Future Internet* 16 (6): 1–16. https://doi.org/10.3390/fi16060208.

Coutaz, J., and J. L. Crowley. 2016. "A First-Person Experience with end-User Development for Smart Homes." *IEEE Pervasive Computing* 15:26–39. https://doi.org/10.1109/MPRV.2016.24.

Desjardins, A., O. Tomico, A. Lucero, M. E. Cecchinato, and C. Neustaedter. 2021. "Introduction to the Special Issue on First-Person Methods in HCI." *ACM Transactions on Computer-Human Interaction* 28: 1–12. https://doi.org/10.1145/3492342.

Ellis, C. 2008. "Autoethnography." In *The SAGE Encyclopedia of Qualitative Research Methods*, edited by L. M. Given, 49–51. Thousand Oaks, CA: SAGE Publications, Inc. https://doi.org/10.4135/9781412963909.

Gallo, S., F. Paternò, and A. Malizia. 2024. "A Conversational Agent for Creating Automations Exploiting Large Language Models." *Personal and Ubiquitous Computing* 28:931–946. https://doi.org/10.1007/s00779-024-01825-5.

Giudici, M., P. Crovari, and F. Garzotto. 2025. "Persuasive Conversational Agents for Environmental Sustainability: A Survey." *ACM Computing Surveys* 58:1–37. https://doi.org/10.1145/3774751.

Giudici, M., L. Padalino, G. Paolino, I. Paratici, A. I. Pascu, and F. Garzotto. 2024. "Designing Home Automation Routines Using an LLM-Based Chatbot." *Designs* 8: 1–37. https://doi.org/10.3390/designs8030043.

Grieves, M. 2015. Digital Twin: Manufacturing Excellence through Virtual Factory Replication. Technical Report, University of Michigan.

Guizzardi, D., B. R. Barricelli, and D. Fogli. 2025. "A user-in-the-loop digital twin for energy consumption prediction in smart homes." Workshops at the international conference on intelligent user interfaces (IUI) 2025, Aachen: CEUR Workshop Proceedings.

Hine, C. 2020. "Strategies for Reflexive Ethnography in the Smart Home: Autoethnography of Silence and Emotion." *Sociology* 54:22–36. https://doi.org/10.1177/0038038519855325.

Houshangi, M. 2025. "Data-driven Smart Home Automation for Energy Efficiency." *SN Computer Science* 6: 1–29. https://doi.org/10.1007/s42979-025-04183-y.

Huang, B., C. Chen, K. Y. Lam, and F. Huang. 2024. "Proactive detection of physical inter-rule vulnerabilities in IoT services using a deep learning approach." 2024 IEEE international conference on web services (ICWS), Shenzhen, China. https://doi.org/10.1109/ICWS62655.2024.00037.

Leadbeater, C., and P. Miller. 2004. *The Pro-am Revolution: How Enthusiasts Are Changing Our Society and Economy*. London: Demos.

Lee, Y. H., and F. J. Lin. 2019. "Situation awareness and conflict resolution in smart home with multiple users." 2019 IEEE 5th world forum on internet of things (WF-IoT), Limerick, Ireland. https://doi.org/10.1109/WF-IoT.2019.8767178.

Liang, G., and Q. Tong. 2025. "LLM-powered AI agent systems and their applications in industry." 2025 IEEE World AI IoT Congress (AIIoT), Seattle, WA, USA. https://doi.org/10.1109/AIIoT65859.2025.11105299.

Lieberman, H., F. Paternò, M. Klann, and V. Wulf. 2006. "End-user Development: An Emerging Paradigm." In *End User Development*, edited by H. Lieberman, F. Paternò, and V. Wulf, 1–8. Dordrecht: Springer. https://doi.org/10.1007/1-4020-5386-X_1.

Liu, W., Y. Lv, Q. Wang, B. Sun, and D. Han. 2024. "A Systematic Review of the Digital Twin Technology in Buildings, Landscape and Urban Environment from 2018 to 2024." *Buildings* 14: 1–26. https://doi.org/10.3390/buildings14113475.

Lucero, A.. 2018. "Living without a Mobile Phone: An Autoethnography." In *Proceedings of the 2018 Designing Interactive Systems Conference, Dis '18*, edited by Teresa Cerratto-Pargman, Kenny Chow, and William Odom, 765–776. Hong Kong: Association for Computing Machinery. https://doi.org/10.1145/3196709.3196731.

Malek, M. R. A., N. A. A. Aziz, S. Alelyani, M. Mohana, F. N. A. Baharudin, and Z. Ibrahim. 2022. "Comfort and Energy Consumption Optimization in Smart Homes Using bat Algorithm with Inertia Weight." *Journal of Building Engineering* 47:103848. https://doi.org/10.1016/j.jobe.2021.103848.

Manca, M., F. Paternò, and C. Santoro. 2022a. "End-user Development in Industrial Contexts: The Paper Mill Case Study." *Behaviour & Information Technology* 41:1848–1864. https://doi.org/10.1080/0144929X.2022.2089597.

Manca, M., F. Paternò, and C. Santoro. 2022b. "Remote Monitoring of end-User Created Automations in Field Trials." *Journal of Ambient Intelligence and Humanized Computing* 13:5669–5697. https://doi.org/10.1007/s12652-021-03239-0.

Manca, M., F. Paternò, C. Santoro, and L. Corcella. 2019. "Supporting end-User Debugging of Trigger-Action Rules for IoT Applications." *International Journal of Human-Computer Studies* 123:56–69. https://doi.org/10.1016/j.ijhcs.2018.11.005.

McDonald, N., S. Schoenebeck, and A. Forte. 2019. "Reliability and Inter-rater Reliability in Qualitative Research: Norms and Guidelines for CSCW and HCI Practice." *Proceedings of the ACM on Human-Computer Interaction* 3: 1–23. https://doi.org/10.1145/3359174.

Nadeem, M., S. Kostic, M. Dornhöfer, C. Weber, and M. Fathi. 2025. “A Comprehensive Review of Digital Twin in Healthcare in the Scope of Simulative Health-Monitoring.” *Digital Health* 11:20552076241304078. https://doi.org/10.1177/20552076241304078.

Nurgaliyev, K., D. D. Mauro, N. Khan, and J. C. Augusto. 2017. “Improved multi-user interaction in a smart environment through a preference-based conflict resolution virtual assistant.” 2017 international conference on intelligent environments (IE), Seoul, Korea (South). https://doi.org/10.1109/IE.2017.21.

Ospan, B., N. Khan, J. Augusto, M. Quinde, and K. Nurgaliyev. 2018. “Context aware virtual assistant with case-based conflict resolution in multi-user smart home environment.” 2018 international conference on computing and network communications (CoCoNet), Astana, Kazakhstan. https://doi.org/10.1109/CoCoNet.2018.8476898.

Stogia, M., V. Naserentin, A. Dimara, O. Eleftheriou, I. Tzitzios, C. Papaioannou, and M. Pantusheva. 2024. “A Scalable and User-Friendly Framework Integrating IoT and Digital Twins for Home Energy Management Systems.” *Applied Sciences* 14: 1–28. https://doi.org/10.3390/app142411834.

Turner, S., J. R. C. Nurse, and S. Li. 2022. ““It Was Hard to Find the Words”: Using an Autoethnographic Diary Study to Understand the Difficulties of Smart Home Cyber Security Practices.” In *Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems, Chi Ea ‘22*, edited by Simone Barbosa, Cliff Lampe, Caroline Appert, and David A. Shamma, 1–8. New Orleans, LA: Association for Computing Machinery. https://doi.org/10.1145/3491101.3503577.

Ur, B., E. McManus, M. Pak Yong Ho, and M. L. Littman. 2014. “Practical Trigger-Action Programming in the Smart Home.” In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Chi ‘14*, edited by Albrecht Schmidt and Tovi Grossman, 803–812. Toronto: Association for Computing Machinery. https://doi.org/10.1145/2556288.2557420.

VanDerHorn, E., and S. Mahadevan. 2021. “Digital Twin: Generalization, Characterization and Implementation.” *Decision Support Systems* 145:113524. https://doi.org/10.1016/j.dss.2021.113524.

Xu, Y., W. Gao, Y. Wang, X. Shan, and Y. S. Lin. 2024. “Enhancing User Experience and Trust in Advanced LLM-Based Conversational Agents.” *Computing and Artificial Intelligence* 2:1467. https://doi.org/10.59400/cai.v2i2.1467.

Yao, J. F., Y. Yang, X. C. Wang, and X. P. Zhang. 2023. “Systematic Review of Digital Twin Technology and Applications.” *Visual Computing for Industry, Biomedicine, and Art* 6:10. https://doi.org/10.1186/s42492-023-00137-4.

Zhang, L., W. He, J. Martinez, N. Brackenbury, S. Lu, and B. Ur. 2019. “AutoTap: Synthesizing and repairing trigger-action programs using LTL properties.” 2019 IEEE/ACM 41st international conference on software engineering (ICSE), Montreal, QC, Canada. https://doi.org/10.1109/ICSE.2019.00043.

Zhang, X., and R. Wakkary. 2014. “Understanding the Role of Designers’ Personal Experiences in Interaction Design Practice.” In *Proceedings of the 2014 Conference on Designing Interactive Systems, Dis ‘14*, edited by Carman Neustaedter, Shaowen Bardzell, and Eric Paulos, 895–904. Vancouver, BC: Association for Computing Machinery. https://doi.org/10.1145/2598510.2598556.