

Article

Off-the-Shelf AAL—A Practical Approach to Face the Population Shift

Gerhard Leitner 

Department for Informatics Systems, University of Klagenfurt, Universitaetsstrasse 65-67,
9020 Klagenfurt, Austria; gerhard.leitner@aau.at

Abstract

Although the concept of Active and Assisted Living (AAL) has been a prominent topic in academia and in industry for decades, the widespread adoption of related technologies remains well below expectations. The underlying causes are multifaceted. The installation and retrofitting of such systems typically require substantial financial investments, significant manual effort, and specialized expertise for setup and maintenance. Existing solutions lack flexibility and are difficult to tailor to the individual living situations and diverse needs of the primary target group, older adults. While state-of-the-art smart home platforms would, in principle, be capable of supporting a broad range of AAL functionalities and could be adapted to different usage contexts, much of the research in this domain has been conducted in artificial settings, such as laboratory environments or model houses, conditions that fail to fully capture the complexity and variability of real-world living environments of the elderly population. In this paper, we explore the potential, opportunities, and limitations of integrating low-cost hardware with open-source software components in residential environments representative of older adults' everyday lives. Our work is based on a longitudinal case study conducted over several years in an actual household, focusing on delivering fundamental AAL functionality. By documenting the iterative development and real-world deployment of the system, this study offers practical insights into the feasibility and challenges of implementing on-site AAL support under realistic conditions.

Keywords: Active and Assisted Living (AAL); smart home; relational databases; SQL; time series analysis; NoSQL

1. Introduction

For many years, and even decades, both scientific literature and popular media have addressed the challenges posed by the forthcoming population shift. A central question arising from this development is how modern societies can ensure adequate well-being and support for an aging population.

The population shift constitutes a multidimensional challenge. One important dimension is the continuously increasing shortage of qualified personnel, driven by the changing ratio of the working population to retirees. Another relevant aspect is that, due to factors such as urbanization, younger adults often live at a considerable distance from their elderly parents. Combined with increasing life expectancy, these developments lead to a growing demand for care services [1], probably drastically [2], for example, due to age-related increased risks of diseases [3], physical (falls [4]) or psychological (dementia, mild cognitive impairments) problems [5–7]. According to [8], the WHO estimates the current prevalence



Academic Editors: Luigi Bibbò and J. Artur Serrano

Received: 20 December 2025

Revised: 20 February 2026

Accepted: 22 February 2026

Published: 26 February 2026

Copyright: © 2026 by the author.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

of various types of disabilities at 15%, with a continuous increase in morbidities and functional limitations in the next years. Due to the increasing shortage of qualified support personnel and the distance to potential informal caregivers, the respective challenges are rapidly growing. As a consequence, healthcare systems face rising pressure because they are insufficiently prepared to cope with increasing demand and diversification [2].

The prevailing strategy of relocating elderly people to nursing homes and other care facilities is the least preferred option for the majority of the target group. The related literature indicates that older adults strongly prefer to remain in their familiar living environment rather than in a nursing facility [1,2,4,6,7,9]. Avoiding institutional care for as long as possible is therefore of core interest to the elderly because *institutional sanitation* [9] can be a traumatic experience, with several negative aspects, for example, social disconnection [6].

Against this background, alternative ways of ensuring support have to be found, combined with new strategies and policies [10]. In addition to psychological benefits, supporting older adults in remaining in their homes also offers economic advantages [1], as, e.g., ref. [11] reports that the costs of home-based care are about 40% lower than those of institutional support.

The concept of Active and Assisted Living (AAL), originally referred to as Ambient Assisted Living, was introduced in the early 2000s [12] and continued in the AAL Initiative of the European Commission [13]—(Ref. [9] refers to similar activities even earlier, around 1993). The basic motivation for AAL was to consider technology a promising means of addressing demographic change, at least in part. In the definition of [3], AAL is understood as “*state-of-the-art ICT-based solutions that build on the principles of ambient intelligence to create intelligent environments that provide all-encompassing, non-invasive, and proactive support to older adults and have the ultimate goal to maintain their independence, enhance their overall quality of life, and support their caregivers*”. The general assumption in the related literature is that appropriately equipped living environments can support aging in place, reduce the need for human support, and, to some extent, delay institutional care. Nevertheless, despite more than two decades of research and development, the basic technology has not yet become part of typical living environments. Existing solutions often fail to meet the practical needs of the target group [7] at the physical, social, and psychological levels, thereby failing to promote, for example, independence [10] and dignity.

One reason for this shortcoming is that standardized or one-size-fits-all smart solutions, which have proven success in industrial contexts (and are also applicable in governmental/-municipal housing and care facilities), are not suitable for private living conditions, because, as [13] describes it, the understanding of living is “*an individual's perceptions of a position in life in the context of the culture and value system in which they live and in relation to their goals, expectations, (individual) standards, and concern*”. Consequently, systems addressing private households must be adaptable, flexible, and scalable.

Although the technological foundations for such systems are meanwhile widely available in the form of smart home technology and the Internet of Things (IoT), a considerable gap remains between technical availability and actual adoption among private homeowners in general and the elderly in particular. One of the reasons is that the technologies are difficult to integrate into the diversity of built environments (e.g., family houses or flats in cities, suburbs, and rural areas) and adapt to the different lifestyle aspects pointed out by [13], e.g., differences in income, level of education, and other factors. Although the related literature highlights the elderly's wish to remain in their familiar living environments for as long as possible, these environments are typically not prepared to accommodate the ICT that would support this wish.

This paper presents a longitudinal case study constituting the most recent step in a series of AAL-related projects addressing the described problem area and its dimensions. The study is based on accompanying one household for several years, with the aim of initially implementing, improving, extending, and evaluating exemplary AAL functionality tailored to the household's characteristics and inhabitants. The work focuses on a group that, to the best of our knowledge, is not appropriately addressed in the related research but could specifically benefit from the capabilities of state-of-the-art technology.

This group, as [13] describes it, is “*adults with no critical pathological condition*”. More precisely, our approach focuses on elderly persons who are no longer fully independent, are typically infrequently supported by informal caregivers (family and friends), but do not need professional or close-knit support. This share of the population, however, does not represent a homogeneous group, neither in terms of age nor in terms of health status and support needs. The group represents different stages of aging [10], where many individuals are still healthy and independent, while others become (highly) dependent on care [2]. Potential support for this target group would need to be as flexible and adaptive as possible, as the following considerations emphasize: With age, the likelihood of health-related problems increases. When problems first arise, the environment is likely to lack the necessary preparation. One example is the (first) occurrence of falls, which typically occur unexpectedly. The probability of such events does not, in general, justify contracting professional support organizations. However, given the state-of-the-art functional capabilities of smart home systems, which will likely become standard equipment in built environments in the future, combined with stronger involvement and empowerment of the elderly themselves and the informal caregivers, solutions are likely in sight.

The main contributions of this work are threefold. First, we demonstrate both the potential and the practical challenges of a long-term (multi-year) deployment of Active and Assisted Living (AAL) technology in a real private home, thereby moving beyond short-term pilot studies. In comparison to field studies with shorter durations (e.g., ref. [14], 6–9 months), the presented study spans more than 10 years and thus accounts for technical changes and advancements over time, which constitute an additional real-world aspect of long-term field deployments. Another difference to other field-based approaches (e.g., refs. [14,15]), which is likewise influenced by technical progress (e.g., the spread and increasing usability of tablets or smartphones), is that the deployed functionality does not provide customized explicit and active user interaction. Instead, it focuses on unobtrusive, background-only sensing, enabling the observation of behavioral routines and long-term drifts, in combination with the observation and analysis of technological/infrastructural aspects that are typically not thematized in shorter-term evaluations.

Second, the proposed solution is based on a fully on-site, local architecture that operates without cloud services or other external computation resources, thereby providing a high level of security and privacy—a key aspect for sensitive application domains such as AAL. This architectural choice supports a high degree of external validity regarding the potential and barriers of AAL solutions, including availability, stability and robustness, maintainability, security, and privacy.

Third, the paper documents a deployment characterized by minimal invasiveness, the use of low-cost, off-the-shelf hardware, and open-source software, thereby providing a feasible and affordable basis for larger-scale AAL deployments.

The remainder of this paper is structured as follows: After providing an overview of related work, we describe the recent steps of our aging-in-place approach, the developed variant of basic AAL functionality, and the achieved results. The paper concludes with a reflection on the pros and cons of the followed approach, its respective possibilities and

limitations, and an outlook on future potentials (up to state-of-the-art machine learning methods, such as LSTM).

2. Related Work

Extensive research in the field of AAL has been carried out over approximately the last 20 years, ref. [10] refers to a vast body of related literature; the SCOPUS database, as of December 2025, lists around 1800 publications that include the terms *assisted living* or *AAL* in their titles and/or keywords. The number of publications addressing the topic using different terminology can therefore be expected to be significantly higher (see, for example, the overview in [10]). Despite this substantial research activity, a major gap remains in the limited proportion of on-site, real-world research activities [1]. As noted by [9], the feasibility of technical solutions is still predominantly evaluated in laboratory settings, whereas only a small number of studies focus on retrofitting the basic technology into the actual living environments of the target group (cf. e.g., ref. [14]). Such approaches do not sufficiently capture the various dimensions that influence acceptance, adoption, and long-term use of technology, nor do they adequately investigate aging in place in the literal sense. There is growing evidence (cf. e.g., ref. [1]) of a mismatch between older people's everyday lives and needs and the technologies and services currently available. This discrepancy results in what [10] describes as the “*need for a large amount of data from real environments*”. However, “*smart homes and AAL systems are complex to build and maintain*” [2], and numerous obstacles exist on different levels [13]. Another limitation is that research often emphasizes technological aspects rather than the individuals who use the systems and their concrete needs. This has led to calls for stronger involvement of the target group [14]. Field-based studies would provide considerable added value, for example, by enabling deeper insights into real living circumstances and fostering a clearer understanding of the needs of both the primary target group (the elderly) and secondary users (caregivers). Given the large number of relevant factors and their complex interdependencies, the following sections provide an attempt at categorization.

2.1. Technical Aspects

The technical infrastructure must be compatible with the built environment into which it is integrated. Fortunately, the corresponding requirements are meanwhile quite low; in most cases, state-of-the-art electrical wiring and an internet connection are sufficient. Even if the latter is not the case, ad hoc solutions can often be realized through mobile network infrastructure (see, e.g., ref. [15]). Technical solutions specifically designed for support, known as *gerontechnology* or *assistive technologies* [9], can mitigate the effect of physical distance, for example by supporting informal caregivers by expanding social connections into *virtual space*. The more recent umbrella term, IoT, similarly encompasses technologies that enable aging in place [1] and is often used synonymously with the concept of the *Smart Home*, as they share many technical fundamentals. Regardless of the terminology or conceptual framing, an essential requirement for technical solutions is unobtrusiveness and a careful assessment of its impact on everyday life. In this context, ref. [10] compare solutions based on environmental sensors with those relying on wearables. The latter may raise acceptance issues; as noted by [6], they tend to be disliked by the elderly, even though popular AAL solutions are based on them. Environmental sensing technologies are generally perceived as less intrusive, except for vision-based sensors, such as cameras, which are also considered problematic [2,6]. Vision-based or other forms of overt data collection are generally considered sensitive; consequently, solutions based on unobtrusive sensing technologies are preferable. While these approaches reduce invasiveness, they come with limitations in precision. If applied advisedly, however, the technology can

support both monitoring and assistance in everyday activities. A smart motion sensor, for example, can be used to observe activity patterns but may also provide assistive functionality, such as switching on all necessary lights when a movement across several rooms is predicted. According to [6], AAL should be conceptualized as broad as possible, combining IoT/Smart Homes and machine learning to support an independent, safe, and socially connected life for elderly people in their own homes. Nevertheless, the current dissemination of the technology shows room for improvement [3]. The technological readiness of the target group and their built environments is low, and the scientific evidence of the appropriateness of technical solutions remains weak [8]. Against this background, recent research has increasingly addressed the issue of technical applicability in the context of ongoing developments in the IoT. Small, easily integrable components representing IoT devices have emerged as suitable building blocks for flexible, adaptable systems. For example, ref. [16] describe an IoT-based framework for smart home surveillance, ref. [17] investigate the potential of IoT-driven automation, and [18] explore monitoring scenarios based on IoT, albeit with a focus on office environments. Accordingly, recent works on low-cost, small-scale system architectures—often referred to as edge-computing approaches—have proven to be particularly relevant. These systems are typically based on platforms such as the Raspberry Pi and rely on standardized communication protocols, including Zigbee or MQTT [19]. While the majority of these works address smart home applications in a general sense, the underlying concepts, insights, and reported results can likewise be transferred to the AAL context. In particular, the emphasis on local processing, modular integration, and reduced infrastructure requirements aligns well with the needs of real-world deployments.

2.2. Psychological and Sociological Aspects

Barriers to technology adoption arise not only from technical limitations but also from psychological and sociological factors. To benefit from technical capabilities, users must be open to new technology, at least accept its presence. According to [1], potential obstacles include limited motivation to learn new technologies, lack of confidence in their use, perceived inability to maintain systems, general dislike of technology, and even fear or frustration, for example, during power outages. Additional influencing factors may include limited financial resources or reluctance to invest in technological support. The concept of user acceptance could serve as a basis for in-depth analysis. For instance, ref. [3] attribute the slow dissemination of respective solutions to insufficient consideration of acceptance-related factors and their alignment to individual needs [2], as described in the various representations of the Technology Acceptance Model (TAM) (cf, e.g., ref. [20]). Beyond the primary users, additional stakeholders must be considered. Informal caregivers represent an important group of indirect or secondary users [13], and their requirements, expectations, and concerns must be addressed accordingly. They assumed a substantial share of support responsibilities [7] already in the pre-smart era, and technology-based solutions may both relieve and challenge them. The challenges arise at technological, ethical and social levels, including issues related to acceptance and social connectedness [6]. Ideally, technology should not replace human caregivers but rather relieve and disburden them [9], for example, by reducing the need for physical visits, enabling continuous monitoring, promoting healthier lifestyles, and enabling early detection of health-related problems [2]. In the best-case scenario [6], technology functions as an in-home companion [1], facilitating autonomy and self-care by reducing the need for third-party intervention, enhancing confidence in everyday activities, and providing physical and cognitive stimulation [7]. Nevertheless, as emphasized by [1], longitudinal studies are required to better understand the impact of

such systems on responsibilities, decision-making processes, and social relationships from the perspectives of the elderly, caregivers, and other stakeholders.

2.3. User Focus and Usability

Appropriate usability and user experience are essential prerequisites for the acceptance and effective use of AAL systems [7]. Achieving them requires greater research on AAL services and related concepts of utilization and interaction. Relevant principles include, for example [9], equitable, simple and intuitive use, perceptible information, tolerance for error, low physical effort, and appropriate spatial circumstances for the interaction. To ensure these requirements, the research must adopt an interdisciplinary approach [1], in a user-centred, co-creative (user-in-the-loop [13]) design process to improve integration into real living environments and address requirements that have often been neglected in earlier work. Historically, insufficient attention has been paid to users' wishes, social and emotional needs and aesthetic preferences [9], sometimes based on incomplete or stereotypical assumptions about older adults [3]. People are typically not interested in technology for its own sake, but rather in its practical applicability. Consequently, more evaluations conducted directly in the homes of older adults are needed [1] in order to understand normal behaviour, develop adaptive systems [10], identify contextual constraints, and collect and analyze representative data. The overarching goal is to reduce the discrepancy between technically "intelligent" solutions and user expectations [13], or, as formulated by [21], to strive for wise technology that surpasses smart technology, the latter being, in many cases, rather smart-ass.

2.4. Society and Policy

From a societal perspective, one of the strongest arguments for supporting aging in place [1] is the preservation of independence and autonomy for the elderly, enabling them to remain in their homes for as long as possible. An additional, equally important argument concerns economic sustainability. Compared to institutional care, home-based support models represent significantly less costly alternatives for public administrations and communities [13]. As additionally pointed out by [2], *"smart home technologies enhancing or complementing home health care can be an integral part of cost-effective healthcare systems"*.

2.5. Functional Range

As outlined in Section 1, one-size-fits-all solutions are not appropriate in the AAL context. Examples for useful functions [9] include healthcare and social care services, such as remote alarm systems based on telephone infrastructure. More advanced systems provide vital parameter monitoring (e.g., based on wearables), activity monitoring, reminder functions and a so-called "virtual neighbourhood". The range of functionality noted by [1] enumerates monitoring of health status, detecting emergencies, and notifying healthcare providers. Other advanced functionality would include automating daily tasks and maintenance, offering reminders or prompting others, supporting communication and (social) connection, and providing transportation and navigation support. Other monitoring functions, exemplified by [10], include medical and diet-related functions, sleep monitoring, and kitchen activity monitoring. Specifically for people who are not very old or do not have particular diseases, observing lifestyle activities, their respective changes, or suspicious behaviour could be useful. It is, however, important that the systems are adapted to the needs and living circumstances of the elderly, based on a limited (absolutely necessary) number of sensors and with an observable functionality, in the optimal case, based on non-intrusive technology [6].

2.6. Algorithmic Level

A central component of respective solutions is the collection and processing of data. From our perspective, this represents one of the most critical aspects of technology-based support. Ideally, because of the aforementioned potential obstacles related to usability, acceptance, and technology experience, the algorithmic processing operates automatically and independently, with as few user interventions as possible. Data generated by smart devices can provide insights into whether observed behaviour follows typical routines or deviates from established patterns. In this context, human behaviour and human activity have to be differentiated; behaviour can be understood as *activity routines* [22] and deviations in these routines may indicate problems. An overview of the field of anomaly detection is provided by [23], who analyzed several hundred publications and highlighted the diversity of approaches. For example, ref. [10] describes machine learning-based approaches, such as support vector machines, neural networks, or long short-term memory (LSTM [24]), which were successfully applied in the detection of behavioural anomalies, ref. [2] refer to specific methods to predict health-related problems. Further approaches referred to by [6] also include LSTM, statistical and Markov models for calculating probabilities, recurrent (neural network) models for analyzing sequences of sensor usage, and forecasting models based on the Holt-Winters method (cf. e.g., ref. [25]). Machine learning and statistical approaches capable of learning normal behaviour (and identifying anomalies) are referred to by [6]. Support vector machines, decision trees, k-nearest neighbour methods and random forests are examined by [4]. More recent works have further expanded the perspective toward IoT. For example, ref. [26] investigate anomaly detection in IoT environments, covering a broad range of classical machine learning approaches and comparing them with emerging approaches such as Quantum Machine Learning (QML) with regard to their applicability in IoT contexts. The work of [27] presents machine learning techniques, such as random forests and LSTMs, for monitoring elderly people. While all these approaches demonstrate considerable methodological potential, their practical applicability remains a central issue. This aspect is explicitly addressed by [6,28], who—unlike many studies (including those referred to above) based primarily on laboratory or other experimental settings—draw on observations from real-world settings.

3. Method

The main conceptual foundations derived from the related work can be summarized as follows. Older adults generally wish to remain in their familiar living environments for as long as possible. At the same time, the increasing scarcity of human support resources suggests that technology may assume selected support tasks, provided certain requirements are met. A key prerequisite is that the respective technology can be integrated into existing living environments, requires only minimal technical prerequisites, operates as unobtrusively as possible, and adapts to users' diverse needs regarding acceptance, security, usability, and overall lifestyle. In comparison to approaches described in the related literature (cf. e.g., ref. [14]), and based on experiences gained in preceding projects (see, e.g., ref. [15]), the focus of the approach presented in this paper lies on the recognition of activity and deviations from established routines. Usability aspects discussed in the literature become particularly relevant when systems provide explicit or customized interfaces to the users. In contrast to earlier projects conducted by the authors and to approaches such as those described by [14], the present study deliberately avoids requiring participants to actively interact with the system. In the preceding project, the software platform was operated on a high-performance computer. To motivate installation and provide added value, we decided on an embedded PC that met the performance requirements and offered an integrated screen and the ability to use basic internet features (e.g., access weather and news

services, write basic emails), rather than a customized interface. However, experiences from this project indicated that most participants (obviously fulfilling the characteristic of “technology openness” referred to above) already employed standard devices such as desktop computers, laptops, smartphones, or tablets, for communication and information purposes. Some participants even expressed feelings of stigmatization (“*Am I too dumb for standard devices?*”) when asked about their interest in devices designed specifically for older people. Based on these observations, the present approach deliberately refrains from providing a dedicated display or user interface. Instead, the focus is on unobtrusive, automatic AAL functionality that operates entirely in the background. As the employed platform is open and flexible, the integration of additional features or subsystems enabling explicit interaction remains possible for future extensions.

A central challenge arises from the combination of a limited penetration of smart home technology and the limited framing conditions for adopting such systems. Starting with a full-fledged integration would therefore not be feasible. Consequently, it was necessary to identify a basic set of initial functionality and carefully assess its applicability using a SWOT-oriented perspective, while simultaneously considering potential extensions that might become relevant in later stages.

When our related research activities began around 2010, the technical possibilities for approaches comparable to the one presented in this paper were limited. At that time, no readily available off-the-shelf solutions existed. Therefore, a custom platform was developed to meet the project’s hardware and software requirements. For example, ref. [29] developed an improvised version of a smart home gateway. On the software level, a middleware platform based on OSGi had to be custom-programmed (see, e.g., refs. [30,31]), enabling the integration of smart home components from different manufacturers and supporting investigations of their potentials and limitations in various application domains, including AAL [15].

In the meantime, platforms offering the features required for basic AAL functionality are now widely available. However, their installation, configuration, and maintenance still require a reasonable level of technical knowledge and expertise. The current status of the platform used in this study is described in the following section.

3.1. Installation and Evaluation Site

The case study presented in this paper is based on an on-site installation in a household inhabited by an elderly woman who had already participated in the project described in [15] and expressed the willingness to continue cooperating, facilitated by a personal relationship with the research team. Due to this personal connection, the study can be considered a form of participant observation, a method with a long tradition in psychology (cf. e.g., ref. [32]) and with demonstrated relevance also in AAL and user experience research [33].

3.2. Participant Characteristics and Ethical Aspects

The participant is an elderly woman aged 87 (reference year 2025) who lives alone in her own family home located in a rural village with approximately 100 inhabitants. The house is surrounded by a small garden. The participant generally performs her Activities of Daily Living (ADLs) independently. Her three children, aged 58 to 67 years, act as informal caregivers and live at distances of 10 to 30 kilometres. Due to the spatial distance between the elderly and the descendant generation (which is typical for the region), the potentials and limitations of a technology-based support system were of particular interest in the context of this study. The participant shows a generally positive attitude toward technology. She regularly uses a tablet computer to communicate with family members, friends, and public authorities via email, and to browse the World Wide Web in order to

stay informed about the latest news. About five years prior to the reference period, she cancelled her printed newspaper subscription due to recurring delivery problems. Since then, she primarily obtains information online with the tablet or via teletext services, which she accesses using her smart TV.

The participant had already participated in the preceding project and had provided informed consent at that time. The consent documentation included a detailed description of the project idea, the data collected and processed (in anonymized form), and the expected involvement of participants, including their willingness to participate in meetings and interviews related to the project. As informal caregivers were an essential group of stakeholders in the preceding project, they were also asked to review and co-sign the informed consent together with the main participant. All participants were provided with comprehensive, state-of-the-art opt-out options (available at any time, without justification, with no disadvantages). Some participants used these options and terminated their participation prematurely. When the original project officially ended, participants were asked to indicate their preferences with regard to two aspects: (1) whether they wished to retain the equipment installed during the project and assume ownership as well as responsibility for its operation and maintenance, and (2) whether they were willing to participate in follow-up projects. A small number of participants expressed interest in both options. Those who agreed to continue their involvement were provided with an updated informed consent form that outlined the technical changes, including modifications in hardware and a reduction in functionality, the latter limited to background activity observation. Due to the technical challenges discussed in the paper, the focus of this follow-up study was subsequently shifted toward a case-study design, with the option to increase its scale again in later stages. The original project was evaluated and approved by the ethics commission of the federal province of Carinthia. As the principal scientific approach did not change, no updated approval was requested. During the project's runtime, specifically in 2017, the University of Klagenfurt's ethics board was established. During the conceptualization of the present manuscript, the board approved the ethical validity of the approach on the basis of the original approval issued by the Carinthian ethics commission.

3.3. Hardware Basis

Continuous advances in hardware development have led to the widespread availability of miniature microcomputers capable of providing sufficient computational power for a range of smart home functionalities. Owing to strong community support and positive prior experience, the Raspberry Pi architecture was selected as the hardware basis of the presented work. Over the past years, several installations based on this platform have been successfully implemented. Initially, the Asus Tinker Board (Asustek, Taiwan) [34] was employed in various smart home projects, including the early stages of the presented case study. However, due to ongoing technological progress and increasing requirements at both the hardware and software levels, the platform underwent several successive updates.

The case study, representing a follow-up to the larger-scale field study mentioned above and detailed in [15] started in 2015, based on an updated variant of the embedded PC architecture (Asus EEE Top) from that study. Once the Raspberry Pi (Raspberry Pi Foundation, United Kingdom) platform had reached sufficient performance levels and the necessary software support became available—such as operating systems, smart home platforms, and tools for remote access—the system was migrated to the Tinker Board in 2017. Subsequently, continuously declining support from both the manufacturer and the community led to the transition to the Raspberry Pi 4 around 2020. Finally, the upgrade to the Raspberry Pi 5 followed in 2023, but it still uses microSD cards to host the operating system. It was not until 2024 that support for solid-state disks (SSDs) on the Raspberry Pi

platform reached sufficient maturity and market prices decreased to an affordable range, making a switch technologically feasible and economically reasonable.

Across the different generations, the Raspberry Pi, was configured as a server hosting the smart home middleware platform OpenHAB as the central component, which will be described in detail in the software section. Although the hardware architecture would allow for direct control of smart components—for example, via the General-Purpose Input/Output Bus (GPIO) [35]—integration with dedicated smart home subsystems proved more advantageous in terms of system stability and operational flexibility. In the concrete installation, the smart home subsystem used is Homematic [36], a platform targeting the end-consumer market and primarily addressing technology-oriented users and advanced do-it-yourself practitioners. The system focuses on the European Market, as the supplier's headquarters are in Germany. The selection of this platform was based on the research team's long-standing, positive experience with Homematic and its predecessor, FS20, particularly regarding fault tolerance, maintenance effort, and programming and configuration capabilities. Figure 1 shows the first version of the on-site installation.

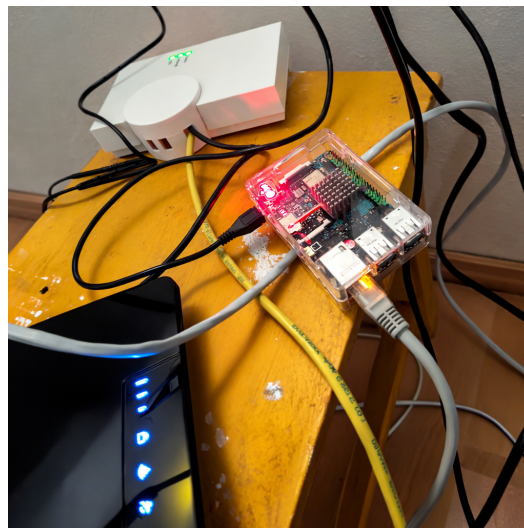


Figure 1. Picture of the initial on-site installation: On the left bottom side, the internet router is observable, top left is the gateway of the smart home sub-system, on the right, the Asus Tinker Board running OpenHAB on Ubuntu Linux.

3.4. Software Architecture

The core software component providing the required smart home functionality is the platform OpenHAB (Open Home Automation Bus, Openhab Foundation, Germany, Releases 3.x (start) to 5.x (current)) [37]. OpenHAB originated from OSGi and the Eclipse Foundation ecosystem [38], which parallels earlier developments conducted by the authors. These parallels and related experience with the underlying principles constituted one reason for selecting OpenHAB over alternatives such as X10 or Home Assistant. Another decisive factor was OpenHAB's proven suitability for thin-client hardware platforms, including Raspberry Pi, combined with a minimal set of requirements.

With respect to installation and configuration, OpenHAB provides an integrated image-based solution (OpenHaBian [39]) that enables rapid setup. However, this study adopted an alternative approach tailored to the specific project requirements. A central requirement was fully autonomous operation without on-site intervention by the supported individuals. This necessitated reliable remote maintenance capabilities. Integrated system images, such as OpenHABian, tightly couple software components, which can result in complete system failure if errors occur, thereby requiring physical on-site intervention. To mitigate this risk, the installation was based on independently operating core components.

Ubuntu Long-Term Support (LTS) was selected as the operating system for the Raspberry Pi, providing enhanced robustness, including automatic recovery following power outages. Remote maintenance and control were enabled using Teamviewer [40], which automatically starts with the operating system and allows high-probability access even under adverse conditions.

OpenHAB itself was installed as a service, started once all prerequisites—such as a Java runtime environment—were operational. A key architectural element of OpenHAB is the use of bindings, which act as software-based connectors that enable the integration of devices and services from different vendors. The variety of bindings is big, going from a few hundred hardware components (such as the aforementioned Homematic), over online services (e.g., for weather), to cloud support and software-related bindings, for example, connectors to different databases.

In the described installation, hardware bindings included Homematic components, such as sensors, lighting actuators, and motion-detection devices. Additional bindings enable the integration of a Smart TV to observe switching events. To support post-processing of trigger events (e.g., switching of Homematic components or the TV), database bindings were employed to connect to, at first, to a MariaDB database and later to InfluxDB. Other relevant OpenHAB bindings include services enabling so-called actions, such as email and SMS messaging. Finally, an important component of OpenHAB in the context of the project is rules, which enable the use of basic programming constructs, such as conditional statements (if, then, else), loops, database calls, etc., to react to observed events.

3.5. Algorithmic Approach

3.5.1. Initial Feasibility Evaluations

To assess the approach's feasibility and basic usefulness, several preliminary evaluations were conducted using simple OpenHAB rules. An example rule triggered the sending of an email to a caregiving relative whenever the resident manually operated a smart home component (shown in Listing 1). The motivation for this functionality draws on several theoretical aspects discussed in Section 2. The most important one is unobtrusiveness, since the system operates entirely in the background.

Listing 1. A rule named “Grandma OK” which observes when a blind is operated. In this case an email is sent to the relative.

```
rule "Grandma OK"
when
    Item Blind changed
then
    val mailActions = getActions("mail", "mail:smtp:XXXXXXXX")
    mailActions.sendMail("XXX@XXXXXXXX.XXX", "Grandma - OK - Blinds",
        "Grandma has operated the blinds")
end
```

The initial trials were at least technically successful, as the triggered events resulted in a corresponding email notification. However, this basic functionality was obviously not really smart and provided only limited informational value, as it did not assess whether the observed activity was typical or atypical with respect to time of the day and component usage. Subsequent improvements, therefore, focused on enhancing the informational quality of notifications. SQL (Structured Query Language) queries were integrated in OpenHAB rules to retrieve historical data from the MariaDB database. The queries were based on the comparison between the current activity level and the historical averages for the corresponding hours of the day, as shown in Listing 2.

Listing 2. A rule named “Grandma Status” which periodically fetches data from a connected database and forwards the result to the relative per E-Mail.

```
rule "Grandma Status"
  when
    Time cron "0 0 0/1 1/1 * ? *" // every hour
  then

    var sql_query = "SELECT((SELECT COUNT(VALUE) /(SELECT datediff(
      MAX(Time),MIN(Time)) from itemXXX) from itemXXX Where HOUR(
      Time) = HOUR(CURRENT_TIME));"
    var String mysql = executeCommandLine( Duration.ofSeconds(1), "
      /usr/bin/mysql", "-h", mysql_host , "-u" , mysql_user , "-p"
      + mysql_pw, "-e", sql_query, mysql_db )

    if( mysql!="")

    {
      val mailActions = getActions("mail","mail:smtp:XXXXX")
      mailActions.sendMail("XXX@XXXXX.XXX","Grandma-Status",mysql.
        substring(mysql.indexOf("\n")))
    }
  end
```

The rule uses a cron function [41] to automatically execute an SQL query [42] on the database hourly, counting how often the component (Item XXX) has been operated. Counting triggers rather than querying component states was necessary because different smart components transmit heterogeneous data to the database. Light switches, for example, typically generate binary values such as *on* and *off*. Roller blinds or light dimmers provide discrete numeric values, for instance, ranging from 0% (open) to 100% (closed), or from 1 (minimal brightness) to 16 (maximal brightness). A television similarly reports *on* and *off* states but may additionally transmit information about the active channel or the current volume level. Other components may provide status values such as *open*, *closed*, or *active/inactive*. While these values are relevant for different functional purposes, they are of limited interest for observing activity patterns. From this perspective, the decisive factor is whether a component was manually operated and how frequently such operations occurred. This information can be reliably derived using the SQL Count function, independent of the specific values transmitted by individual devices. For readability, the examples shown in this paper represent a simplified query applied to a single component. In the actual installation, however, the query included seven components, and the resulting values were aggregated into a derived average, which was forwarded to the caregiving relative. The transmitted message contained the result of comparing the number of triggered events over the past hour with the average number of triggered events for the same hour of the day across the entire observation period represented in the database. This approach significantly improved the information quality of the generated notifications. However, it also pushed the performance and stability limits of the OpenHAB runtime environment. Consequently, it has been decided to transition to alternative, more specialized mechanisms in order to further enhance both system robustness and the quality of the extracted information.

3.5.2. MariaDB and PhpMyAdmin

The next development step involved using database-level processing in PhpMyAdmin, a database management tool that provides advanced data manipulation capabilities. This approach enabled shifting complex, resource-intensive computations away from the limitations of the OpenHAB runtime environments. Instead of relying on the OpenHAB rule engine, data processing was implemented using stored procedures that operate independently of OpenHAB. In addition to relocating basic functionality from OpenHAB to PhpMyAdmin, further considerations were made regarding algorithmic advancement. As already discussed, the data generated by the attached components comprises heterogeneous data and exhibits varying triggering frequencies. Furthermore, the temporal distribution of trigger events can be influenced by seasonality, day–night cycles, and individual activities and routines. It was therefore necessary to process the stored data and transform it into a representation capable of accommodating these variations. Given the requirements discussed in Section 2, the primary objective at this stage was to avoid reliance on external computing resources, such as cloud services. This decision was motivated by technical constraints—including connection stability and limitations in bandwidths—as well as security and privacy concerns. Consequently, all data processing was designed to be performed locally. Due to the computational limitations of the Raspberry Pi, particularly in the early versions, the approach was restricted to statistical analysis methods. Specifically, data normalization and manipulation based on the z-distribution (standard normal distribution; see Equation (1)) were employed.

$$z = \frac{x - \mu}{\sigma} \quad (1)$$

In the first step, the data in MariaDB was recalculated and normalized using the z-transformation, and the results were subsequently written to separate data tables. The resulting data is characterized by constant features (mean $\mu = 0$, standard deviation $\sigma = 1$). Again, the goal was to determine the difference between the number of manual triggers observed in the most recent hour and the typical trigger frequency in the same hour of the day in the past. The z-distribution enables the identification of deviations based on calculated z-values, which express the statistical distance from expected behaviour. Deviations are commonly interpreted in terms of standard deviations (σ). Depending on the desired sensitivity of the algorithm, absolute z-exceeding a threshold of 2 (i.e., $z > 2$ or < -2) can be regarded as statistically significant deviations. The computation of the z-values and the assessment of deviation significance were performed using the aforementioned stored procedures. These procedures can be understood as programs providing enhanced functionality directly within the database system. They can be executed locally, require no external computational resources, and run automatically at predefined intervals. Compared to the earlier cron/rule combination, this mechanism offers substantially greater stability (because it runs decoupled from the smart home runtime), flexibility and supports more complex processing logic.

The described step, illustrated in Listing 3, shows an example stored procedure that calculates a z-score and writes the result to a separate database table in order to preserve the integrity of the raw sensor data. To meet the requirement to inform the caregiving relative of the current status, the email action is used again within a rule. However, instead of executing complex SQL queries directly in the rule engine, the rule only fetches the most recent z-value, corresponding to the past hour and cumulated across all components, from the database. Compared to an SQL query directly embedded in a rule, this approach significantly reduced the computational resources required.

Listing 3. Stored procedure for calculating a z-value based on the comparison between the number of triggers in the past hour and the average trigger frequency for the corresponding hour of the day in the whole observation period.

```

INSERT INTO table Y
SELECT
0.5 *(1+SIGN(z_score)*(1-EXP(-2*z_score* z_score/PI()))), ""
,CURRENT_TIMESTAMP
FROM (
SELECT
(
(
SELECT COUNT(*)
FROM 'table X'
WHERE HOUR('time') = HOUR(CURRENT_TIMESTAMP)
AND DATE('time') = CURDATE()
)
-
(
SELECT AVG(x.count)
FROM (
SELECT COUNT(*) AS count
FROM 'table X'
WHERE HOUR('time') = HOUR(CURRENT_TIMESTAMP)
GROUP BY YEAR('time'), MONIH('time'),
DAY('time')
) AS x
)
)
/
(
SELECT STDDEV_SAMP(y.count2)
FROM (
SELECT COUNT(*) AS count2
FROM 'table X'
WHERE HOUR('time') = HOUR(CURRENT_TIMESTAMP)
GROUP BY YEAR('time'), MONIH('time'), DAY('time')
) AS y
) AS z_score
) AS subquery

```

3.6. Time Series/InfluxDB

The approach based on conventional relational databases again proved useful to some extent, improving information quality. However, over time, several problems emerged that required changes at both the hardware and software levels. Regarding the hardware, the Tinker Board's architecture and peripherals exhibited several weaknesses. One issue was the limited robustness of the 32-bit architecture, combined with the decreasing vendor and community support. On the hardware level, the Tinker Board was therefore replaced by state-of-the-art Raspberry Pis—initially generation 4 and later generation 5—the latter based on a 64-bit ARM architecture equipped with 8 GB of RAM. This transition also opened opportunities for upgrading the software infrastructure. The operating system was upgraded to a 64-bit version of Ubuntu, and more recent versions of OpenHAB, the

associated runtime environments could be deployed (e.g., Java 21 instead of Java 11). At first, the new platform worked better than the previous one. However, a critical bottleneck remained: the SD Card. For short-term or simple prototyping projects, the Raspberry Pi platform provides convenient, SD card-based runtime images. These storage media are multipurpose and have proved useful in various application domains. However, they are not optimized for scenarios with high-frequency or continuous read and write operations, which is typically the case when operating systems handle multiple runtime services in parallel. The high demands placed on the operating system and concurrently running services led to instability, disconnections, and, in some cases, damage to the boot image stored on the SD Cards. These issues ultimately led to the decision to switch to an SSD (solid-state disk)-based solution. SSDs are significantly better suited for operating system execution and high-frequency read and write operations. In addition, support for SSDs on the Raspberry Pi has reached a high level of quality, and their prices are comparable to those of SD Cards.

The basic installation on the new platform—consisting of the latest-generation Raspberry Pi with SSD storage, Ubuntu (Canonical Group Ltd., United Kingdom), TeamViewer (Teamviewer GmbH, Germany), and OpenHAB—was largely comparable to the initial platform based on Tinker Board described above. The essential difference was that the new architecture enabled a transition to a more modern approach to data processing and manipulation, while still fulfilling the core requirements of unobtrusive operation, fully local execution of the required functionality and remote control. The central component of the revised software collection is the time-series database InfluxDB (version 2), which represents a NoSQL database management environment. InfluxDB uses proprietary concepts, such as organizations and buckets, and a specific database query language, Flux, to manage and manipulate data. Compared to the MariaDB/PhpMyAdmin-based approach, Influx offers a broader range of integrated data manipulation capabilities specifically designed for time-related data. This is particularly relevant in the present context, where trigger events occur at specific points in time and at frequencies influenced by temporal factors such as the hour of the day, day–night cycles, and seasonal variations. InfluxDB is directly supported by OpenHAB via a dedicated binding that treats the database as a persistence layer, enabling the storage and manipulation of data generated by smart home or IoT components.

The starting point when introducing InfluxDB was not as low-level as in the previous development stages. Since the z-distribution-based approach has already proven useful, the primary goal was to port the existing functionality to the new platform and adapt it to its specific characteristics. The connection between OpenHAB and InfluxDB is conceptually similar to that of MariaDB; the major differences lie in how data is organized and manipulated. Influx is structured around the concepts of organizations and buckets, which, in a simplified sense, correspond to databases and tables in relational systems. Unlike MariaDB, InfluxDB provides an integrated management environment and therefore does not require external add-ons such as PhpMyAdmin. The platform is based on a built-in graphical web interface that supports a variety of data visualization and manipulation. For more sophisticated programming tasks Influx provides an integrated Query Builder based on a code view. Both the graphical interface and the code-based processing support various connections to external systems. However, the version used in this work (InfluxDB 2), which is currently the only version integratively supported by OpenHAB, relies on the proprietary data manipulation language Flux and other proprietary concepts.

Due to the different paradigms on which InfluxDB is based—some of which deviate considerably from conventional database concepts and SQL—the transfer of the existing functionality required substantial adaptation and proved to be somewhat cumbersome.

The current status of the implementation is illustrated in Listing 4, which builds on the same concepts as the previous approaches (z-value-based deviation recognition, and hours as units of observation).

Listing 4. Influxdb query language (flux) based query for calculating a z-value based on the comparison between the trigger count of the past hour and the average trigger frequency for the corresponding hour over the observation period.

```
import "math"
import "date"

nowTime = now()
startOfCurrentHour = date.truncate(t: nowTime, unit: 1h)
startOfPreviousHour = date.add(to: startOfCurrentHour, d: -1h)

base = from(bucket: "openhab")
  |> range(start: -300d, stop: startOfPreviousHour)
  |> filter(fn: (r) => r["_measurement"] == "Motion3")
  |> aggregateWindow(every: 1h, fn: count, createEmpty: false)
  |> map(fn: (r) => ({ r with joinKey: "all" }))

stats = base
  |> keep(columns: ["_value", "joinKey"])
  |> reduce(
    identity: {count: 0.0, sum: 0.0, sumsq: 0.0, joinKey: "all"},
    fn: (r, accumulator) => ({
      count: accumulator.count + 1.0,
      sum: accumulator.sum + float(v: r._value),
      sumsq: accumulator.sumsq + float(v: r._value) * float(v: r._value),
      joinKey: "all"
    })
  )
  |> map(fn: (r) => ({
    joinKey: "all",
    mean: r.sum / r.count,
    stddev: math.sqrt(x: (r.sumsq / r.count) - (r.sum / r.count) * (r.sum /
      r.count))
  }))

// Current hour
currentHour = from(bucket: "openhab")
  |> range(start: startOfPreviousHour, stop: startOfCurrentHour)
  |> filter(fn: (r) => r["_measurement"] == "Motion3")
  |> aggregateWindow(every: 1h, fn: count, createEmpty: false)
  |> map(fn: (r) => ({ r with joinKey: "all" }))

join(
  tables: {base: currentHour, stats: stats},
  on: ["joinKey"]
)
  |> map(fn: (r) => ({
    _time: r._time,
    _measurement: r._measurement,
    _value: r._value,
    z_score: (float(v: r._value) - r.mean) / r.stddev
  }))
  |> filter(fn: (r) => math.abs(x: r.z_score) > 2.0)
```

Accordingly, the query also needed to be executed following an automated pattern, which required finding an equivalent mechanism to stored procedures. InfluxDB provides so-called *tasks* for this purpose. These tasks offer a broad range of functionality and

can, to a certain extent, be regarded as counterparts to stored procedures. As in the previous approach, data is retrieved from the data bucket, transformed into z-values, and written to a separate bucket in order to preserve the original raw data. The OpenHAB rule engine subsequently searches for entries corresponding to the past hour and sends a notification to the relative's email address if the observed data deviates significantly from the expected value.

4. Results

The results obtained from the approaches implemented so far can be summarized as follows. Over the full observation period, about 300,000 trigger events were recorded across the different database instances and subsequently analyzed. Due to problems of varying severity at both the software and hardware levels—such as power outages, SD card failures, and platform runtime issues—and the resulting system changes, the collected data is characterized by incompleteness, inconsistencies, and weaknesses throughout the overall period of observation. To improve interpretability, the following results represent only illustrative excerpts from selected system phases (Tinkerboard/Mariadb and Raspberry Pi/InfluxDB). The presented periods were characterized by comparatively high system stability and data consistency, which were prerequisites for formulating reasonable hypotheses and deriving meaningful conclusions.

4.1. Results of Rule-Based Notifications

The first “Hello World” function, which simply sent email notifications to a predefined email address (e.g., a relative's), operated as expected. However, during certain periods, it resulted in a very large number of messages, in some cases exceeding one hundred emails within a short time span (see Figure 2). One of the fundamental goals—to execute the required functionality locally on a slim client installation—could be achieved. The emails were only a few kilobytes in size, and their transmission frequency did not overload the infrastructure, neither on the Raspberry Pi (Tinker Board), nor on the mail server, nor on the internet connection. Nevertheless, although this approach allowed a concerned relative to remain informed that some activity was taking place, the functionality was evidently not very smart. The relative still has to manually interpret whether the observed activity was typical or potentially problematic.

omi-service@d... Omi- OK - FS (O)Mama hat den Fernseher betätigt <Ende>	Fr 19.12.2025 11:02 2 KB
omi-service@d... Omi- OK - FS (O)Mama hat den Fernseher betätigt <Ende>	Fr 19.12.2025 11:02 2 KB
omi-service@d... Omi- OK - FS (O)Mama hat den Fernseher betätigt <Ende>	Fr 19.12.2025 10:53 2 KB
omi-service@d... Omi- OK - FS (O)Mama hat den Fernseher betätigt <Ende>	Fr 19.12.2025 10:52 2 KB
omi-service@d... Omi- OK - FS (O)Mama hat den Fernseher betätigt <Ende>	Fr 19.12.2025 10:48 2 KB
omi-service@d... Omi- OK - FS (O)Mama hat den Fernseher betätigt <Ende>	Fr 19.12.2025 10:47 2 KB
omi-service@d... Omi- OK - FS (O)Mama hat den Fernseher betätigt <Ende>	Fr 19.12.2025 10:43 2 KB
omi-service@d... Omi- OK - FS (O)Mama hat den Fernseher betätigt <Ende>	Fr 19.12.2025 10:27 2 KB
omi-service@d... Omi- OK - Motion (O)Mama war in der Werkstatt <Ende>	Fr 19.12.2025 09:50 2 KB
omi-service@d... Omi- OK - Motion (O)Mama war in der Werkstatt <Ende>	Fr 19.12.2025 09:46 2 KB

Figure 2. A snippet of sent emails, representing around one hour in the morning.

4.2. Results of the Tinker Board/MariaDB/PHPMyAdmin Period

As preparation for the next development step, the recorded data was analyzed at different levels. The initial hardware and software platform did not support more complex

computations, and the attempts to run them resulted in system instability and breakdowns. One approach was to export data to external tools, in this case via .csv files to Microsoft Excel, in order to perform more detailed analysis and visualization. Figure 3 shows a snippet of a spreadsheet in which the data was further processed and examined. For improved readability, the figure represents only a three-week period but includes all relevant characteristics of the underlying dataset. White cells indicate insignificant deviations, red cells indicate significant deviations, and yellow cells denote data associated with uncertainty, for example, missing values (e.g., related to a component not sending trigger data during the observed period). Empty areas show periods during which no data were recorded, for instance as a result of power outages or problems with runtime components such as OpenHAB, the database, or the smart home sub-subsystem. The overall observation period represented by the dataset spans approximately one year (6 October 2023 to 31 October 2024). The total number of computed entries is 9493, representing hourly computed values (z-values, averages). The average number of manual triggers is—without considering daily, monthly or seasonal variations, as shown in Figure 3—approximately 3.5. This corresponds to roughly 30,000 trigger events per year and approximately 330,000 events over the entire observation period from 2015 to 2025. Within the illustrated example period 203 computed values are missing, corresponding to about 2.2% of the dataset. A detailed investigation of the missing data showed that over 70% of the data loss occurred during cold periods (November, December, January, February), which is not surprising because the region in which the observed household is located is characterized by overhead power lines that are periodically damaged by storms and snow breakage in trees. Around 10% of data losses occurred in April as well as September, which cannot be plausibly explained, especially when compared with the other months (March–2%, May–3%, June–2%, July–1%, August–0%, October–2%). The latter percentages might be an effect of the concrete observation period.

13.11.2023	14.11.2023	15.11.2023	16.11.2023	17.11.2023	18.11.2023	19.11.2023	20.11.2023	21.11.2023	22.11.2023	23.11.2023	24.11.2023	25.11.2023	26.11.2023	27.11.2023	28.11.2023	29.11.2023
0,03757505	0,03757505	0,03757505	0,03757505	0,03757505	0,03757505	0,03757505	0,03757505	0,11923988	0,07876933	0,036452	0,036452			0,036452	0,036452	0,036452
0,01503495	0,01503495	0,01503495	0,01503495	0,01503495	0,01503495	0,01503495	0,01503495	0,06746874	0,01506135	0,01506135	0,01506135			0,01506135	0,01506135	0,01506135
0,02464431	0,02464431	0,02464431	0,02464431	0,02464431	0,02464431	0,02464431	0,02464431	0,02464431	0,02464431	0,02464431	0,02464431			0,02464431	0,02464431	0,02464431
0	0	0	0	0	0	0	0	0	0	0	0			0	0	0
0,0047496	0,0047496	0,0047496	0,0047496	0,0047496	0,0047496	0,0047496	0,0047496	0,0047496	0,0047496	0,0047496	0,0047496			0,0047496	0,0047496	0,0047496
0,03494709	0,03494709	0,03494709	0,03494709	0,03494709	0,03494709	0,03494709	0,03494709	0,03494709	0,03494709	0,03494709	0,03494709			0,03494709	0,03494709	0,03494709
0,1165488	0,16875053	0,14268179	0,16894627	0,09512354	0,16904715	0,16909718	0,14261898	0,14277751	0,09540915	0,14309112	0,06932176			0,16824278	0,19450446	0,1693554
0,20165773	0,25387435	0,12138123	0,23033326	0,3103681	0,27032549	0,30971324	0,12125368	0,14096006	0,25204658	0,37499495				0,33221257	0,17336814	0,26945489
0,02678783	0,0842787	0,09124903	0,02658983	0,02658983	0,02658983	0,04917095	0,07235986	0,15338043	0,09119017	0,17909147				0,02645459	0,02645459	0,10330593
0,09634246	0,09634246	0,09634246	0,11896243	0,09650787	0,09650787	0,11906621	0,11916929	0,0967108	0,22629689	0,2165157				0,09658712	0,22595746	0,0965753
0,07986327	0,21124566	0,08016763	0,08016763	0,12793771	0,10117219	0,08011752	0,15183307	0,08002962	0,20181285	0,10101869				0,07989245	0,12790508	0,07983202
0,06923669	0,17134398	0,17102402	0,12927661	0,18859129	0,14743163	0,20605003	0,12310916	0,18732273	0,06937288	0,12308821				0,06932911	0,19852509	0,0980006
0,0811483	0,21000099	0,1860174	0,13210032	0,18566572	0,18553285	0,13197123	0,1854678	0,14697346	0,08066296	0,18533072				0,34410578	0,14702989	0,14682245
0,07752365	0,07752365	0,07752365	0,07761895	0,07761895	0,07761895	0,07761895	0,07761895	0,07761895	0,07761895	0,20197974				0,12726966	0,07738763	0,07738762
0,0459571	0,0459571	0,09894154	0,10847252	0,07658002	0,04583582	0,04583582	0,07665767	0,07673481	0,04593671	0,04593671				0,04593671	0,07681146	0,07688761
0,11443085	0,16168247	0,23195191	0,11427484	0,16163623	0,11421661	0,13709606	0,16161908	0,24311731	0,16157561	0,23137676				0,13690755	0,13696434	0,19423908
0,13369465	0,18503459	0,07781502	0,29980146	0,24853887	0,0573347	0,24833116	0,05720538	0,3288744	0,3129591	0,18163696				0,39978744	0,25951643	0,093551
0,17353732	0,34126477	0,07792262	0,07792262	0,13740621	0,12074821	0,07783431	0,19792821	0,0776437	0,12061128	0,07756124				0,22632338	0,12051381	0,31087013
0,05060072	0,23716185	0,07700164	0,38464853	0,17977844	0,11446426	0,10981972	0,05862879	0,15457219	0,13111583	0,22255421				0,05652289	0,11428002	0,09804406
0,05059266	0,30977449	0,09374012	0,15126082	0,09359601	0,05098338	0,15261387	0,10866266	0,05072064	0,16892851	0,05066978				0,33747338	0,05123627	0,17234144
0,10649734	0,06482557	0,30140888	0,06498922	0,19517363	0,18996202	0,06439846	0,10538822	0,26524131	0,06410832	0,26903127				0,17678498	0,11898337	0,06367158
0,11414627	0,07024413	0,06918243	0,06918243	0,06918243	0,17301513	0,0690935	0,0690935	0,11301694	0,06909176	0,25479325				0,2050971	0,11221908	0,15059775
0,03861358	0,1257644	0,03848915	0,1343697	0,03866021	0,03866021	0,23579658	0,03800737	0,03800737	0,03800737	0,0763627				0,03790728	0,19663951	0,03743267
0,0390601	0,0390601	0,0390601	0,0390601	0,0390601	0,0390601	0,0390601	0,0390601	0,11746566	0,03887725	0,06140382				0,03869242	0,03869242	0,25926525
																0,03832274

Figure 3. A snippet of an Microsoft Excel Sheet showing significant (red), unclear (yellow) and not-significant data, as well as missing data (empty fields), the latter, e.g., resulting from power outages.

A visualization generated in Excel and shown in Figure 4 illustrates that the typical daily distribution of trigger events follows a quite regular pattern. Activity levels between midnight (right side of the figure) and the early morning hours (left side) are consistently low. A pronounced activity peak occurs in the morning, influenced by factors such as seasonality and health status. This is followed by a medium activity level during the late morning and a noticeable decrease around noon, when the observed person typically rests. Activity levels during the afternoon and evening hours exhibit greater variation and tend to be higher during colder seasons. While this might appear counterintuitive at first glance, it can be explained by the fact that all observed components are located indoors. As the house includes a garden, activities during warmer periods often take place outside and are therefore not reflected in the sensor data.

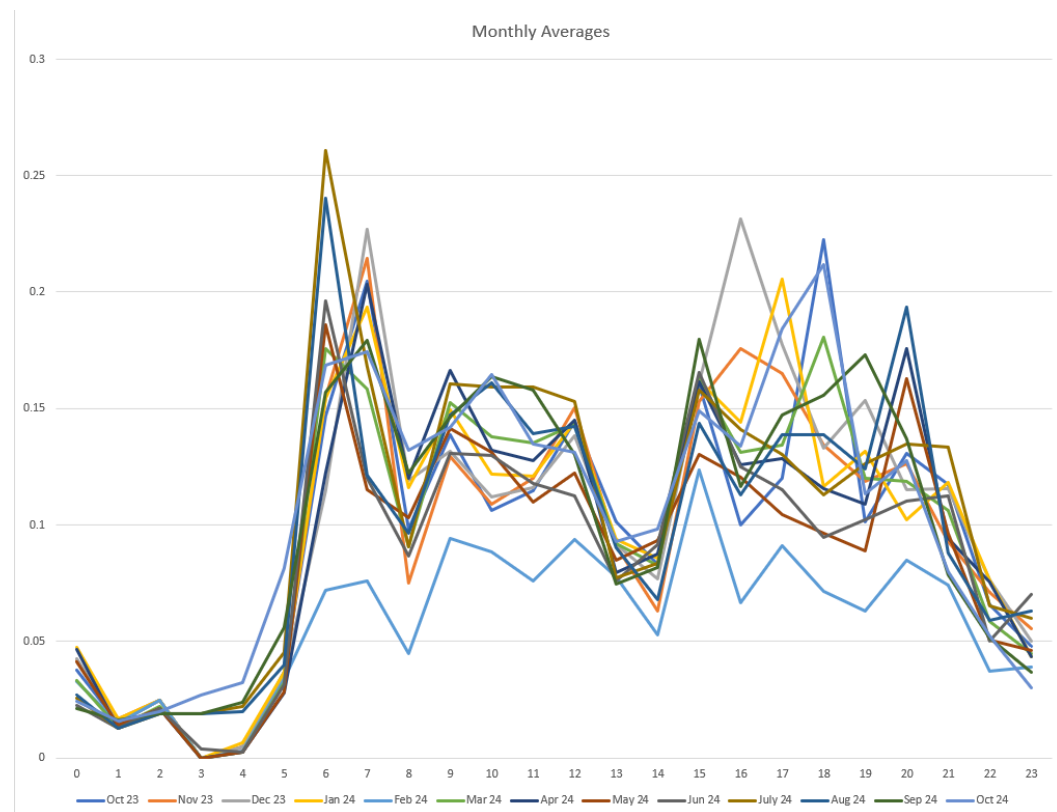


Figure 4. Overview of activities and respective deviations in a One-Year Period (Monthly averages).

The external analyses provided important insights that were incorporated into the further development of both the approach and its algorithmic basis. One key outcome was the introduction of stored procedures and the derivation of rules based on their results, as described in Section 3. This significantly increased the efficiency of the approach, on the one hand by reducing the number of transmitted emails, as only significant deviations were reported, and on the other hand by relieving the hardware through a clear decoupling of data processing and database manipulation from the email transmission process. As discussed previously (in Section 3) and as shown in the missing data in Figure 3, the platform's limited performance and robustness indicated a change.

InfluxDB

The following results represent the current status of the project, which, at the data processing level, is now based on InfluxDB. As shown in Figure 5, data handling in InfluxDB is considerably more convenient compared to the previous approach. Visualization, data manipulation, and interfaces to external sources and platforms are integral components of the web-based graphical user interface. As a result, preparatory analyses and consistency checks can be performed within a single platform, eliminating the need to export data to external tools such as Excel. At the same time, and in line with the main requirements derived from the related work, all procedures continue to operate locally and unobtrusively within the thin-client architecture.

Figure 6 illustrates different representations of the activity analysis based on the most recent system version. On the left side of the figure, a segment of the OpenHAB log protocol is shown, used to observe how values are computed and identify potential errors. On the right, the corresponding email notification is shown, including the calculated z-value and the time and date of observation. As the example reflects the platform's most recent development stage, which is still considered a testing phase, the notification threshold is set to a comparatively low absolute value of 0.5 (i.e., $z > 0.5$ or $z < -0.5$).

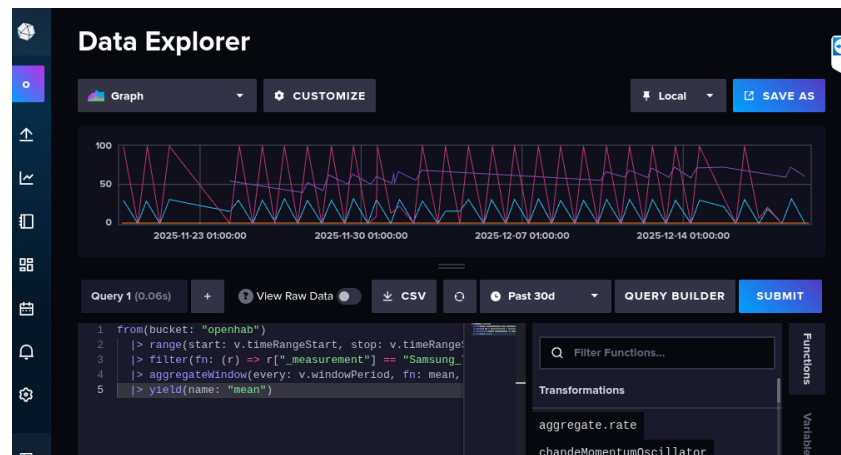


Figure 5. Exemplary view of the InfluxDB web interface.

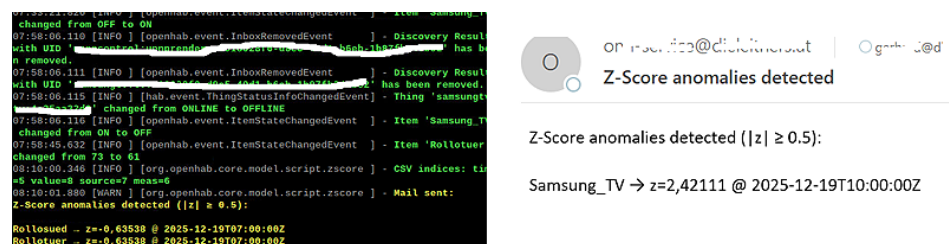


Figure 6. Computed z-values in the OpenHAB log and as email notification when a threshold is exceeded.

In summary, the current algorithms performed appropriately in the observed situations. For example, household renovation work led to increased movement in front of the motion sensor, which was correctly identified as elevated activity. Similarly, a subsequent relaxation period following the renovation led to increased television usage, which was also appropriately detected. Due to the recentness of the installation, no significant deviations were observed during the observation period that would have indicated a health-related problem. Nevertheless, based on the system behaviour observed so far, the algorithm would likely identify such deviations appropriately. Further investigations and longer observation periods based on the current platform version are required to validate this assumption.

5. Discussion

This paper represents an attempt to bring Active and Assisted Living (AAL) to a broader share of the elderly population. Although the results may appear limited and weak at first glance, they reflect real-world conditions and the practical difficulties and obstacles associated with integrating technology into environments that are typically not optimally prepared for such interventions. A cautious and considered approach was therefore necessary. In this sense, the work aimed to balance potential benefits with the drawbacks of the applied technologies. Although several requirements for appropriate AAL solutions were identified in the related work section, only a subset could be addressed in the concrete implementation examples. The reason is that the feasibility—particularly under real-life conditions, as opposed to laboratory settings or artificial living environments, which constitute the majority of related work—must be approached with sensitivity. Consequently, the focus of the illustrated AAL functionality was placed on activity observation, analysis, and reaction mechanisms that fulfill the requirements of unobtrusiveness and, in the sense of the technology acceptance model (TAM), usefulness and ease of use, as no explicit interaction by the supported person is required.

As discussed in Sections 3 and 4, several obstacles must still be overcome before the respective technologies can be integrated more broadly to support a larger share of the elderly population in their own homes. One important obstacle is the range of technical skills still required to install and maintain the necessary infrastructure and functionality. Although instructions, guidelines, and other types of documentation are available, they are typically distributed across multiple sources and require a certain level of technical literacy to interpret and apply.

From a data-processing perspective, the z-distribution-based approach proved useful, though with limitations that must be considered. The calculated z-values provide a basic indication of potential behavioural deviations. As shown in Figure 4, the participant in this case study exhibits relatively stable routine behaviour; therefore, averaging over the entire observation period provides acceptable information value. At the same time, the approach is sensitive to deviations. As discussed in Section 4, the z-values correctly indicated increased activity during renovation work. However, these deviations were also influenced by the presence of additional persons, as the participant could not carry out the renovation alone.

An in-depth investigation and detailed analysis of correctly and incorrectly identified deviations were conducted and are shown in Table 1. An observation period of two recent months was analyzed in detail for two reasons. First, the selected observation period is based on the most recent version of the platform, which is characterized by greater stability and more complete data. Second, due to the recency of the observation period, additional sources of information for comparison with the observation data were more easily accessible. We focused on true and false positives for activity deviations, using data from a single motion sensor located in the household's basement. The significantly deviant z-values were matched to data from other sources (e.g., calendar entries of the visiting relatives or their Google Timeline data). The results show that on weekdays, the percentage of true positives is very high (around 87%), while the percentage of false positives is approximately 14%. On weekends, however, the proportion of true and false positives changes significantly: the majority of identified deviations are false positives (about 61%), compared to roughly 39% true positives. These findings provide important insights and will—besides other aspects—guide the direction of our future work (discussed in Section 6).

Table 1. Overview of a detailed data analysis of a Motion Sensor during a two-month observation period (10 December 2025–10 February 2026, $n = 177$).

Group	Positives	Count	Share (%)
Overall	True	123	69.49
	False	50	28.25
Weekday	True	101	86.32
	False	16	13.68
Weekend	True	22	39.28
	False	34	60.71

From an interpretative perspective—supported by findings from the preceding project described in [15])—the attribution of activity to a specific individual is of secondary relevance. As long as activity is detected, it can be assumed that at least one person is present who could provide support if required. This reasoning also applies to multi-person households and eases the observation (when multiple users would not have to be differentiated). If no activity is recognized, the situation is effectively equivalent to that of a single-person household.

The new system architecture, in particular the replacement of SD Cards with SSD-based storage, significantly improved stability. Power outages, such as those visible in Figure 3) can be handled more robustly, as the system restarts automatically once power is restored. Nevertheless, the platform is still not entirely error-free. Contrary to expectations, situations still occur in which the system neither stores nor transmits data and is no longer accessible remotely. Although this occurs far less frequently than with the SD cards, it remains problematic. In such cases, a simple toggling of the corresponding circuit breaker in the household's fusebox—an action that can be performed by the participant—restores functionality. However, diagnostic information regarding the error cannot be recovered, and a certain amount of data loss must be accepted. Remote maintenance is further limited by the low bandwidth of rural internet connections, which does not affect data transmission itself but makes modification of algorithms or InfluxDB tasks cumbersome.

Another important learning concerns the influence of the observation system on participant behaviour. In the preceding project, noticeable behavioural changes were observed among both elderly participants and their relatives. Some participants explicitly reported modifying their behaviour—for example, deliberately triggering sensors (*"We always check that the connecting door is closed"*) to simulate "typical" activity. In one case, a relative expressed concern when the system reported prolonged inactivity while the participant was performing repair work in the cellar. Although such deviations had existed prior to the study, the changed quality and granularity of information significantly altered the sensitivity and interpretation of the involved persons.

A contrasting effect was observed during the early phases of the study regarding the high number of notification emails. The large volume of messages overwhelmed the recipient, increasing the likelihood that relevant information might be overlooked—an effect comparable to losing the overall perspective due to information overload, where information noise reduces the effective information content. In other words, *not seeing the wood for the trees*.

Insights from the preceding project further showed that behavioural changes tend to decline over time. In the long-term setting of the presented study, no *Hawthorne-like* effect could be observed. The participant is aware of the system operating in the background and actively uses the monitored components, but her behaviour does not appear to be influenced by the observation itself. However, because of the personal proximity to the research team, such an effect cannot be ruled out in other settings. No observable changes in behaviour can probably also be attributable to the substantially longer deployment duration compared to typical field studies reported in the literature, which often span only a few months. Nevertheless, further investigations of these aspects remain necessary. Privacy and security represent additional critical concerns in such sensitive application domains. We addressed these concerns by running the Raspberry Pi in a headless configuration, meaning no keyboard, mouse, or screen were attached. While these peripherals could theoretically be connected, the operating system is protected by standard authentication mechanisms. Remote access via TeamViewer is end-to-end encrypted and additionally secured by login credentials. The data processed locally is highly abstract and not interpretable without contextual metadata. For example, a recorded event such as *HM-LC-Sw2-FM-OEQ1663598:1, "On", 20 May 2015 06:30:13* does not convey any critical personal information. Additional semantic labelling (e.g., *couch-light*) is stored only within the system for better readability. Overall, the privacy and security risks do not exceed those that are typically associated with standard internet-connected or IoT-based systems.

6. Conclusions and Future Work

The presented work can be considered a small contribution within the broad field of AAL, highlighting the challenges and constraints encountered in real-world deployment. As emphasized throughout the paper, the underlying technology is largely available; however, its meaningful and sustainable application continues to resist straightforward implementation, particularly when evaluated from a comprehensive user experience perspective.

At the same time, noticeable progress can be observed compared to earlier field studies (specifically, ours). One important development is the availability of open and extensible smart home platforms such as OpenHAB, which served as one of the central technical foundations of this work. Long before major industrial stakeholders introduced the Matter standard [43], open-source communities had already established practical solutions for integrating heterogeneous components from different manufacturers via platforms such as Home Assistant or OpenHAB [18]. In this context, community-driven developments such as Zigbee2MQTT [44] also deserve particular attention. Zigbee is, in principle, an open standard which would allow for the direct integration of devices from different suppliers. However, as pointed out by [45], differences in device specification and communication behaviour between vendors [45] (e.g., mention Ikea or Philips Hue, but also Miele@home relies on a proprietary and closed Zigbee variant) significantly reduce practical interoperability. The goal of Zigbee2MQTT is to overcome such vendor-specific and proprietary deviations and thereby increase interoperability between devices from different manufacturers. From a functional perspective, the smart home platforms themselves are largely equivalent, offering broad protocol support and extensibility. Zigbee2MQTT provides an important complementary advantage to platforms such as OpenHAB or Home Assistant, as Zigbee devices can communicate directly and fully locally, without necessarily relying on vendor-specific gateways, which in turn often introduce additional cloud dependencies. Matter plays a specific and somewhat ambivalent role in the context of smart home platforms. Despite its significantly stronger market backing (including GAFA companies), Matter currently offers capabilities that are largely comparable to those of open platforms. At the same time, Matter has been subject of critical discussion, particularly with respect to privacy and security implications that arise from design choices, such as the exchange of credentials and data across different vendor-specific subsystems within the MATTER ecosystem [46]. With regard to usability—an aspect emphasized as crucial for the broader adoption of smart home technologies in general and AAL in particular—Home Assistant provides more accessible add-ons and plugins for non-technical users, whereas OpenHAB is less supportive, e.g., in terms of visualization tools to define rules for users not familiar with programming. One of the original concepts underlying such end-user programming approaches can be found in Google Blockly [47], which is based on the idea of composing functionality on the basis of connected building blocks representing programming concepts (such as loops, conditions, and actions) and has been—beside other fields of application—also researched in the context of IoT [48,49] or smart home programming [50]. Although these concepts would reduce the entry barrier for laypersons, they still constitute “programming” and therefore require a basic understanding of the underlying concepts and fundamentals of logic. A promising alternative is likely found in artificial intelligence, specifically in Large Language Models (LLMs), as they are based on natural language interaction, which is presumably more accessible for laypersons and may eventually support them in designing, configuring, and maintaining smart home systems [51], probably also in the context of AAL. Currently, however, LLM-based approaches still require manual transfer of prompt results into target IoT environments (e.g., OpenHAB or Home Assistant). Agentic AI paradigms could further enhance this context as agents could autonomously

perform the transfer and integration steps, thereby closing the gap between conversational interaction and operational system programming [52].

Given demographic change and shortages in professional personnel, such developments appear increasingly necessary. Already in 2006, ref. [53] argued that human–computer interaction would shift from the era of *easy to use* to an era of *easy to develop*. From today’s perspective, neither of these goals has yet been fully achieved, though notable exceptions exist. One illustrative example is video production, which—due to its technical complexity—was formerly reserved to technology enthusiasts. Today, however, almost any smartphone user can record, edit, and publish videos, including titles, background music, and scene transitions. This development demonstrates that, with sufficient industry motivation and commitment, highly complex functionality can indeed be transformed into user-friendly solutions. The aforementioned approaches based on agentic AI can be considered a means of supporting diverse user groups in managing their ICT-related problems independently, representing the technological state of the art. However, societal concerns accompanying such developments (e.g., ethics, security, privacy) should be critically examined [54].

Nevertheless, the system described in this paper cannot currently be configured or maintained by non-technical users. Substantial technical knowledge and sustained interest remain necessary. A potential service or commercialization model could therefore involve “AAL as a Service”, in which specialized providers centrally manage installation, configuration, and maintenance, while caregiving relatives assume responsibility for interpretation and response. As the primary target group consists of older adults who are not yet dependent on professional care or close-knit support structures, questions arise regarding affordability as well as legal and organizational aspects—particularly with respect to liability. Regarding scalability, the presented approach shows promising potential. Because it relies solely on time-based observation of sensor trigger frequencies rather than explicit activity recognition, it can likely be transferred to other environments unsupervised. This assumption is supported by results from previous work [15], which deployed diverse sensor combinations across approximately 20 real-world households. Despite participants often stating that they did not follow strict routines, observation periods revealed that even a small number of smart components (as in the presented study, which required manual triggering) was sufficient to identify routine behaviour and deviations. Following an initial learning phase, the system was able to derive typical activity patterns and detect anomalies. Effective AAL support does not require explicit interaction or wearable devices. Environmental infrastructure and passive observation can already increase safety compared to conventional support, such as periodic visits or phone calls. The presented concept may even operate using devices not traditionally associated with smart homes, including televisions, internet routers, hi-fi equipment, or household appliances. The presented approach can automatically and unobtrusively react to deviations and trigger appropriate actions, as exemplified in this work by email notifications.

The approach nevertheless exhibits limitations while offering multiple opportunities for further improvement, particularly at the algorithmic level. Deviation detection based on the z-distribution has proven suitable as a baseline method but has only been partially evaluated using standard key metrics such as precision and recall. The time-series data managed within InfluxDB provides a promising basis for more advanced analysis, for instance by accounting for seasonality and higher-order temporal patterns. Moreover, forecasting and modelling approaches are directly supported by InfluxDB. Initial experiments with the MAD (median absolute deviation) algorithm [55], which promises improved outlier detection in the InfluxDB ecosystem, finally became successful and enabled preliminary analyses. However, due to the comparatively small volume of available data, no

generalizable findings can yet be derived. Nevertheless, first insights were obtained from analyzing the ratio of false to true positives, as shown in Table 1. We applied both the z-score calculation and MAD on the data and observed the following initial differences: the proportion of identified significant deviations was comparatively low (in the range of 20–30%) for both methods. However, the majority of significantly deviating data identified by the z-score method represent false positives, whereas a slight majority of deviations identified by MAD represent true positives. It should be noted that the initial successful application of MAD—after several unsuccessful trials due to the specific characteristics of the data—were not optimized with respect to parameter tuning, observation window sizes or other algorithmic refinements. These enhancements, together with the integration of external sources (e.g., calendar information, Google Timeline as referred to in Section 5, or geofencing), will be addressed in future work. Despite the limitations, the initial results are promising in terms of validity. As MAD relies on median-based statistics, it is expected to be more robust to outliers and comparatively better than z-scores, which are based on means and averages and are therefore expected to be more sensitive to rare extreme events. In the context of in-home activity recognition, MAD is expected to capture deviations from habitual behaviour more reliably.

A further direction for future work involves the application of artificial intelligence-based methods such as LSTM and related machine learning approaches, which are frequently discussed in the related literature (cf. e.g., refs. [16–18]). Several challenges must be considered, including whether such methods can be executed locally without reliance on cloud-based infrastructures, given privacy, security, and connectivity constraints. In many application domains, comparable problems are addressed using (cloud-based) GPU hardware, which introduces additional complexity. Platforms such as NVIDIA Jetson [56] offer potential solutions by providing local GPU resources on edge-based architectures. Their applicability, compatibility with open-source platforms and sustainability will be explored in future work.

Funding: This research received no external funding.

Institutional Review Board Statement: The study was originally approved by the Ethics Committee of the Federal Province of Carinthia (Case Number A14/09) on 20 January 2010; the correctness of the approach and the continuing validity of the ethics approval were confirmed by the Ethics Board of the University of Klagenfurt on 22 January 2026.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study, separately for all study phases (and respective significant infrastructural or methodological changes).

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the author.

Acknowledgments: I thank the participant of the presented study for her long-lasting support, and the reviewers for their valuable comments. Open Access Funding support by the University of Klagenfurt.

Conflicts of Interest: The author declares no conflicts of interest.

References

1. Carnemolla, P. Ageing in place and the internet of things—how smart home technologies, the built environment and caregiving intersect. *Vis. Eng.* **2018**, *6*, 7. [CrossRef]
2. De Moraes, W.O.; Lundström, J.; Wickström, N. Active in-database processing to support ambient assisted living systems. *Sensors* **2014**, *14*, 14765–14785. [CrossRef]
3. Jaschinski, C.; Ben Allouch, S. Why should I use this? Identifying incentives for using AAL technologies. In *European Conference on Ambient Intelligence*; Springer International Publishing: Cham, Switzerland, 2015; pp. 155–170.

4. Qian, Z.; Lin, Y.; Jing, W.; Ma, Z.; Liu, H.; Yin, R.; Li, Z.; Bi, Z.; Zhang, W. Development of a real-time wearable fall detection system in the context of Internet of Things. *IEEE Internet Things J.* **2022**, *9*, 21999–22007. [CrossRef]
5. Almeida, A.; Mulero, R.; Rametta, P.; Urošević, V.; Andrić, M.; Patrono, L. A critical analysis of an IoT—Aware AAL system for elderly monitoring. *Future Gener. Comput. Syst.* **2019**, *97*, 598–619. [CrossRef]
6. Forkan, A.R.M.; Branch, P.; Jayaraman, P.P.; Ferretto, A. An internet-of-things solution to assist independent living and social connectedness in elderly. *ACM Trans. Soc. Comput.* **2019**, *2*, 1–24. [CrossRef]
7. Linner, T.; Schmailzl, M.; Bock, T.; Hu, R.; Güttler, J. Active Assisted Living Technology in the Context of the Built Environment. In *The Routledge Companion to Ecological Design Thinking*; Routledge: London, UK, 2022; pp. 555–564.
8. de Oliveira, G.A.A.; de Bettio, R.W.; Freire, A.P. Accessibility of the smart home for users with visual disabilities: An evaluation of open source mobile applications for home automation. In Proceedings of the 15th Brazilian Symposium on Human Factors in Computing Systems, São Paulo, Brazil, 4–7 October 2016; ACM: New York, NY, USA, 2016; pp. 1–10.
9. Kerbler, B. An innovative built environment form for dwellings for the elderly. *METU J. Fac. Archit.* **2016**, *31*, 119–137. [CrossRef]
10. Cicirelli, G.; Marani, R.; Petitti, A.; Milella, A.; D’Orazio, T. Ambient Assisted Living: A Review of Technologies, Methodologies and Future Perspectives for Healthy Aging of Population. *Sensors* **2021**, *21*, 3549. [CrossRef] [PubMed]
11. Genworth and CareScout Release Cost of Care Survey Results for 2024. 2025. Available online: https://d1io3yog0oux5.cloudfront.net/_5776620e19c07afdf63df34681e5ef23/genworth/news/2025-03-04_Genworth_and_CareScout_Release_Cost_of_Care_982.pdf (accessed on 21 February 2026).
12. Steg, H.; Strese, H.; Loroff, C.; Hull, J.; Schmidt, S. Europe Is Facing a Demographic Challenge Ambient Assisted Living Offers Solutions; IST Project Report on Ambient Assisted Living. 2006. Available online: <http://www.aal-europe.eu/Published/FinalVersion.pdf> (accessed on 21 February 2026).
13. Cesta, A.; Cortellessa, G.; Fracasso, F.; Orlandini, A.; Turno, M. User needs and preferences on AAL systems that support older adults and their carers. *J. Ambient. Intell. Smart Environ.* **2018**, *10*, 49–70. [CrossRef]
14. Dupuy, L.; Sauzéon, H. Effects of an assisted living platform amongst frail older adults and their caregivers: 6 months vs. 9-month follow-up across a pilot field study. *Gerontechnology* **2020**, *19*, 16–27. [CrossRef]
15. Leitner, G.; Felfernig, A.; Fercher, A.J.; Hitz, M. Disseminating ambient assisted living in rural areas. *Sensors* **2014**, *14*, 13496–13531. [CrossRef]
16. Thakur, P.; Goel, S.; Puthooran, E. Edge AI enabled IoT framework for secure smart home infrastructure. *Procedia Comput. Sci.* **2024**, *235*, 3369–3378. [CrossRef]
17. Yar, H.; Imran, A.S.; Khan, Z.A.; Sajjad, M.; Kastrati, Z. Towards smart home automation using IoT-enabled edge-computing paradigm. *Sensors* **2021**, *21*, 4932. [CrossRef] [PubMed]
18. Khomenko, Y.; Babichev, S. Modular IoT Architecture for Monitoring and Control of Office Environments Based on Home Assistant. *IoT* **2025**, *6*, 69. [CrossRef]
19. Debauche, O.; Mahmoudi, S.; Moussaoui, Y. Internet of things learning: A practical case for smart building automation. In Proceedings of the 2020 5th International Conference on Cloud Computing and Artificial Intelligence: Technologies and Applications (CloudTech), Marrakesh, Morocco, 24–26 November 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1–8.
20. Leitner, G.; Mitrea, O.; Fercher, A.J. Towards an acceptance model for AAL. In *International Conference on Human Factors in Computing and Informatics*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 672–679.
21. Leitner, G. *The Future Home Is Wise, Not Smart: A Human-Centric Perspective on Next Generation Domestic Technologies*; Springer International Publishing AG: Cham, Switzerland, 2015.
22. Mojarad, R.; Attal, F.; Chibani, A.; Amirat, Y. A context-aware hybrid framework for human behavior analysis. In Proceedings of the 2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI), Baltimore, MD, USA, 9–11 November 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 460–465.
23. Schmidl, S.; Wenig, P.; Papenbrock, T. Anomaly detection in time series: A comprehensive evaluation. *Proc. VLDB Endow.* **2022**, *15*, 1779–1797. [CrossRef]
24. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
25. Aboode, A. Anomaly Detection in Time Series Data Based on Holt-Winters Method. KTH, School of Electrical Engineering and Computer Science (EECS), Stockholm, Sweden. 2018. Available online: <http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-226344> (accessed on 21 February 2026).
26. Aparcana-Tasayco, A.J.; Deng, X.; Park, J.H. A systematic review of anomaly detection in IoT security: Towards quantum machine learning approach. *EPJ Quantum Technol.* **2025**, *12*, 112. [CrossRef]
27. Ali, A.; Montanaro, T.; Sergi, I.; Carrisi, S.; Galli, D.; Distant, C.; Patrono, L. An Innovative IoT and Edge Intelligence Framework for Monitoring Elderly People Using Anomaly Detection on Data from Non-Wearable Sensors. *Sensors* **2025**, *25*, 1735. [CrossRef]
28. Chen, C.; Das, B.; Cook, D.J. A data mining framework for activity recognition in smart environments. In Proceedings of the 2010 Sixth International Conference on Intelligent Environments, Kuala Lumpur, Malaysia, 19–21 July 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 80–83.

29. Felsing, D. Eine Erweiterbare Smart Home Plattform auf Basis des FS20 Systems. Diploma Thesis, Alpen-Adria Universität Klagenfurt, Klagenfurt am Wörthersee, Austria, 2009.
30. Fercher, A.J. Pervasive Visualization of Energy Consumption. Diploma Thesis, Alpen-Adria Universität Klagenfurt, Klagenfurt am Wörthersee, Austria, 2011.
31. Rabl, W. Multimodale Interaktion im Smart-Home-Bereich. Diploma Thesis, Alpen-Adria Universität Klagenfurt, Klagenfurt am Wörthersee, Austria, 2009.
32. Aagaard, J.; Matthiesen, N. Methods of materiality: Participant observation and qualitative research in psychology. *Qual. Res. Psychol.* **2016**, *13*, 33–46. [CrossRef]
33. Holtzblatt, K.; Beyer, H. *Contextual Design: Design for Life*; Morgan Kaufmann: San Francisco, CA, USA, 2016.
34. Tinker Board–ASUS. Available online: <https://tinker-board.asus.com/series/tinker-board.html> (accessed on 21 February 2026).
35. GPIO. Available online: https://en.wikipedia.org/wiki/General-purpose_input/output (accessed on 21 February 2026).
36. Homematic. Available online: <https://de.wikipedia.org/wiki/EQ-3#Homematic> (accessed on 21 February 2026).
37. OpenHAB. Available online: <https://en.wikipedia.org/wiki/OpenHAB> (accessed on 21 February 2026).
38. Eclipse IoT. Available online: <https://iot.eclipse.org/> (accessed on 21 February 2026).
39. OpenHabian. Available online: <https://www.openhab.org/docs/installation/openhabian.html> (accessed on 21 February 2026).
40. Teamviewer. Available online: <https://www.teamviewer.com/en/> (accessed on 21 February 2026).
41. CRON. Available online: <https://en.wikipedia.org/wiki/Cron> (accessed on 21 February 2026).
42. SQL–W3C School. Available online: https://www.w3schools.com/sql/sql_intro.asp (accessed on 21 February 2026).
43. The Matter Standard. Available online: [https://en.wikipedia.org/wiki/Matter_\(standard\)](https://en.wikipedia.org/wiki/Matter_(standard)) (accessed on 21 February 2026).
44. Zigbee 2 MQTT. Available online: <https://www.zigbee2mqtt.io/> (accessed on 21 February 2026).
45. Madsen, S.S.; Staugaard, B.E.; Ma, Z.; Yussof, S.; Jørgensen, B.N. A cost-effective edge computing gateway for smart buildings. In *Energy Informatics Academy Conference*; Springer Nature Switzerland: Cham, Switzerland, 2024; pp. 37–54.
46. Fang, Y.; Wang, H.; Jin, Z.; Liu, Q. Chaos of Functionalities: Understanding Security Risks in Heterogeneity of IoT Matter Controllers. In *Proceedings of the 2025 IEEE 24th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, Guiyang, China, 14–17 November 2025; IEEE: Piscataway, NJ, USA, 2025; pp. 95–102. [CrossRef]
47. Blockly–Google Developers. Available online: <https://developers.google.com/blockly?hl=en> (accessed on 21 February 2026).
48. Bak, N.; Chang, B.M.; Choi, K. Smart Block: A visual block language and its programming environment for IoT. *J. Comput. Lang.* **2020**, *60*, 100999. [CrossRef]
49. Koß, A.; Dartmann, G.; Gollmer, K.-U.; Fazlic, L.B. Simplifying Edge AI: A Block-Based Development Platform for IoT Applications. In *Proceedings of the 2025 MIPRO 48th ICT and Electronics Convention*, Opatija, Croatia, 2–6 June 2025; IEEE: Piscataway, NJ, USA, 2025; pp. 274–279. [CrossRef]
50. Leitner, G.; Fercher, A.J.; Lassen, C. End users programming smart homes—A case study on scenario programming. In *International Workshop on Human-Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 217–236.
51. Leitner, G. The Role of Generative AI in the Future of Smart Home Configuration. In *Proceedings of the 27th International Workshop on Configuration (ConfWS 2025)*, Bologna, Italy, 25–26 October 2025; CEUR-WS: Aachen, Germany, 2025.
52. Wang, H.; Gong, J.; Zhang, H.; Xu, J.; Wang, Z. Ai agentic programming: A survey of techniques, challenges, and opportunities. *arXiv* **2025**, arXiv:2508.11126. [CrossRef]
53. Lieberman, H.; Paternò, F.; Klann, M.; Wulf, V. End-user development: An emerging paradigm. In *End User Development*; Springer: Dordrecht, The Netherlands, 2006; pp. 1–8.
54. Chandra, J.; Navneet, S.K. Advancing responsible innovation in agentic AI: A study of ethical frameworks for household automation. *arXiv* **2025**, arXiv:2507.15901. [CrossRef]
55. Median Absolute Deviation. Available online: <https://www.influxdata.com/blog/anomaly-detection-with-median-absolute-deviation/> (accessed on 21 February 2026).
56. NVIDIA Nano. Available online: <https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/> (accessed on 21 February 2026).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.