IoT

MDPI

Article

Towards a Policy Management Framework for Managing Interaction Behaviors in IoT Collectives

Amna Batool *, Seng W. Loke, Niroshinie Fernando and Jonathan Kua

School of Information Technology, Deakin University, Geelong, VIC 3220, Australia; seng.loke@deakin.edu.au (S.W.L.); niroshinie.fernando@deakin.edu.au (N.F.); jonathan.kua@deakin.edu.au (J.K.)
* Correspondence: abatool@deakin.edu.au

Abstract: This paper proposes a policy management framework which we call the SANIJO framework. This framework comprises three different types of policy rules that are applicable to smart devices for managing their multiuser–multidevice interactions in IoT collectives, from a socio-ethical perspective. We developed a policy language to help regulate and manage the interaction behaviors of smart internet-connected devices that are being deployed at an increasing rate around the world. The policy rules are classified into Authorization, Obligation, and Prohibition rules and are prototyped in the SANIJO system. We implemented our framework as a collection of mobile apps (running on smartphones) and a robot app (running on the robot). We then illustrate its operation based on an aged care center scenario.

Keywords: policy rules; policy management framework; Internet of Things

check for updates

Citation: Batool, A.; Loke, S.W.; Fernando, N.; Kua, J. Towards a Policy Management Framework for Managing Interaction Behaviors in IoT Collectives. *IoT* **2021**, *2*, 633–655. https://doi.org/10.3390/iot2040032

Academic Editor: Hyun-Ho Choi

Received: 28 July 2021
Accepted: 23 September 2021
Published: 20 October 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The Internet of Things (IoT) comprises collections of multiple smart devices that exchange and collect data from each other. In this work, we define a collection of smart internet-connected devices working together in a specific environment an IoT collective [1]. Smart devices perform different actions according to their specific use, but it is important to consider the control on their interaction behaviors while interacting with different users, from the socio-ethical perspective. For example, in aged care centers, we can keep multiple IoT devices to monitor the admitted seniors and assist staff members (e.g., carers) in their daily activities. The smart devices should follow the interaction behavior policies while monitoring the senior or assisting a staff member because it is important that the user feels comfortable while interacting with technology and that the devices behave in a socio-ethically correct way.

Researchers are working on a variety of ways to govern IoT in terms of privacy, transparency, interaction behavior, and so on, especially where IoT devices are used in important applications. A framework is proposed in [2] with two functionalities: analyzing the human behavioral patterns associated with complex activities of daily living and detecting any anomalies in user behavior that could constitute an emergency through an intelligent decision-making algorithm is proposed to assist elderly people living a quality life. Australian privacy standards have been considered to protect people's privacy in the new IoT environment, where a lot of data is captured in real time, which can help solve specific IoT legal issues [3].

Researchers have also examined policy-based methods to make IoT systems better through an ethical by design approach. A policy-based framework is proposed in [4] that follows the ethical design principle to protect users' privacy when engaging with IoT. To make the design, development, deployment, and assessment of drones in public healthcare more efficient, the work in [5] developed an ethical framework based on ethical principles such as beneficence, non-maleficence, autonomy, and justice, as well as comprehensibility.

The work in [6] proposed HOMESNITCH, a system which allows users to manage smart devices installed in their homes by detecting their semantic behavior.

The main gaps in the literature we address in relation to socio-technical (and even ethical) behaviors in human–device interaction behaviors in IoT devices to be filled in this research are explained briefly as follows:

- An enhancement in human–device interaction requires considering the decision-making transparency, and devices must have an ability to deal with multiple requests by making decisions. When multiple human users try to interact with smart devices simultaneously, then there is a need for decision-making processes that assist devices in making the right decisions and informing users about their behaviors.
- Human–device interaction can be very complex. If any hurdle comes across during task accomplishment, then there should be a proper system for handling network or device failures and keeping the humans in the loop.
- Devices need to trigger consensus-seeking processes from time to time, which means seeking approval from users before taking any action, e.g., in terms of being privacy respecting. Moreover, it is important to keep devices in polite behavior during human–device interaction because being privacy respecting, being polite, being prudent, being attentive, and being transparent in actions are main behavioral modules that devices are obligated to trigger in different circumstances.

To this end, this paper aims to:

- Investigate appropriate policy rules addressing human-centric behaviors which are (i) socially appropriate, (ii) provide decision-making transparency, and (iii) help in dealing with uncertainty in situations while interacting with humans.
- Investigate and propose a policy language using Extensible Markup Language (XML) that best conveys the interpretation of each policy rule by multiple smart devices during human-IoT interactions.
- Design and develop a prototype as a proof-of-concept to validate the approach of modeling such complex systems.

To summarize, the main contributions of this work are to propose the SANIJO policy management framework and the SANIJO policy language that defines three types of policy rules and their interpretations by a collection of IoT devices. The types of policy rules are defined below:

- Authorization rules , to control allowable tasks and interactions performed by smart devices. These can allow a range of actions of the devices by users, e.g., monitor-user, assist-user, send-request, and retrieve-data.
- Obligation rules, to ensure certain actions and interaction behaviors are carried out. Some examples can be, be-polite, be-privacy-respecting, be-attentive, be-transparent-in-actions, secure-user-data, handle-task-failures, and be-prudent.
- Prohibition rules, e.g., to enforce certain limits.

We elaborate further on these types of rules later in the paper. According to [7], hard-to-implement ethical AI principles, the generic notion of trust vs. trustworthiness, and product/process support for trust/trustworthiness are three issues associated with human-AI interaction. If the policy rules are based on ethical principles, we note that our policy-based approach can be a mechanism for operationalising ethical behavior, from a deontological or rule based approach, in smart devices such as robots. Hence, the paper is also making a contribution towards the operationalization of ethics in a pragmatic implementation-based manner. This could complement other approaches towards operationalizing AI ethics, e.g., the toolbox for thinking through AI principles [8].

The interpretation of each rule by multiple smart devices might be different, e.g., each type of device, including smart robots, smart cameras, smart speakers, smart wheelchairs, and smart vacuum cleaners, might interpret the same obligation rule, say “Be polite”, in its own way. Therefore, this paper has designed the SANIJO language to capture the

interpretation of each rule by multiple smart devices. The SANIJO language is an extensible markup language (XML language) to describe required rules/behaviors for smart devices, which can scan the documents in the language dynamically and decode them before performing operations. By "scan", we mean a device reads in an XML document and uses the content for executing "code" (program code), and on executing the "code", the device behaves in a certain way—in this way, the device "operationalises" a policy rule. Each smart device can process two SANIJO language documents, i.e., to process the three types of rules to follow and to learn how to decode each rule. After processing, the smart device can start its operation. For example, in an aged care center, the smart robot processes the policy rule document to learn what rules it has to follow as well as the rule interpretation document to learn how it will decode each rule while performing its job. The manager of the aged care home might want to conveniently control the interaction behaviors of the smart robot by applying policy rules on it using a central system (a system containing all policy rules which can be applied on smart devices) installed in his/her office. In this way, the smart robot can perform its job but when doing so, it must adhere to policy rules.

In this work, our implementation is on smart internet-connected robots to demonstrate the applicability of the SANIJO framework and SANIJO language. Scanning the policy language and then following all applied policies by the smart robot while interacting with seniors or staff members in an aged-care center is beneficial because in this way, the smart devices can interact ethically with the users and make them feel comfortable—and what behaviors are abstracted via the rules in SANIJO. The background work (Section 2) highlighted the literature where it is clear that researchers have focused on human–device interaction but have not focused on how a device must behave with humans during human–device interactions.

The rest of this paper is organized as follows. Section 2 presents the related work. Section 3 presents the reference architecture. Section 4 presents the consideration for managing the interaction behaviors of smart devices in multiuser–multidevice interactions. Section 5 presents the SANIJO policy management framework for IoT collectives. Section 6 presents the SANIJO language used in our SANIJO framework for IoT collectives. Section 7 presents the prototype of the policy management framework in the proposed SANIJO system, including the illustrative scenario, the system's main concepts and implementation. Section 8 concludes with a discussion on future work.

2. Related Work

There has been substantial research conducted on human–device interaction but very limited work has been done on interaction behaviors with collections of IoT devices.

The idea of policy-based systems goes back to Morris [9], which demonstrated the idea of a passive policy object, which provides a connection between one or more subjects and one or more target domains, describes the actions the subject takes on the target, and sets a constraint to restrict the policy's application. The authors showed how to utilize the policy object to simulate a variety of authorization and obligation policies. The idea of policy object may be used to represent both high-level (unimplementable) goals that people understand and low-level implementable rules that automated objects can interpret. It is difficult to automatically deduce low-level policies from high-level policies. In addition, policy analysis, conflict identification, and resolution are all critical topics that require extensive investigation.

Liang [10] has gone through the literature review to extract ethical considerations required in the case study design phases. Through the literature, a total of 21 ethical considerations have been developed, which have been organized into six ethical principles: "informed consent," "scientific value," "beneficence," "confidentiality," "anonymity," and "exceptions." The study's researchers conducted 22 interviews in a multi-case study to evaluate the usefulness of ethical problems identified in the literature. The findings show that the identified ethical concerns are relevant in the case study design phases, with each step requiring consideration of ethics.

Niemelä et al. [11] performed three separate experiments at a residential care facility to see how utilizing a telepresence robot affects elders, family members, and care professionals. The facility's telepresence robots were installed to allow seniors to communicate with their family members through video call. There were ethical concerns to address based on the results of each experiment; for example, privacy was a major worry for elders and family members. Participants agreed that it is necessary for a primary user to choose someone they feel comfortable speaking with.

Most of the studies focused on the social interaction behavior of smart devices such as James et al. [12] which explored the social aspects of a robot tutor's interaction with children in learning a second language. The authors conducted their study with 67 children where the smart robot provided a session to children and, then some pre- and post-tests conducted to check if children have improved their learning. The results showed a significant improvement due to the social interaction of a smart robot with children. However, the study targeted robots only, not multiple IoT devices. Moreover, the research has not talked about managing the interaction behaviors of robots. The study has not covered various aspects of interaction behaviors which this work is actually covering.

Similarly, Alessandra et al. [13] identified the effect of social interactive behavior of a smart robot on user's trust while performing an activity in real-world scenarios. The authors used a humanoid robot to guide users in a human-populated environment. The interaction behaviors that built up user's trust in the humanoid robot were talking and maintaining a safe distance from obstacles. The study has covered social interactive behaviors of smart robots, but the study has focused on only one authorized action taken by the robots, which is "guide user", whereas our work is focusing on multiple actions that IoT devices can perform and while performing those actions, all devices are obligated to behave ethically and prohibited from performing some actions to enforce certain limits.

There are some studies conducted on proposing frameworks to manage the social behavior of smart robots. Felipa et al. [14] proposed an emotional agent framework to manage the social behavior of smart robotic game player in a card game called Sueca. The challenge faced by the authors was to develop an artificial player and the research question to answer by the authors was how to allow the robotic game player to play the game with natural and social behaviors towards their partners as well as opponents. The challenge faced by the author on perceiving hidden data was resolved by using Perfect Information Monte Carlo (PIMC) algorithm. The results showed the robotic player has 60 percent winning ability, as well as that the users increase their trust in robots due to their social behavior. It is clear from the study that the behavior of smart devices toward humans is very important. Unfortunately, it is not about robots only to target and embed them with social behaviors, but multiple IoT devices must be included as our study is targeting multiple smart devices, not just robots, though our demonstrator focuses on robots.

Besides the social behavior of smart devices, there has been substantial research conducted on the ethical behavior of smart devices. Sharkey et al. [15] have introduced six main ethical issues of keeping robots in health care centers, which include: less human interaction, a loss of human control, a loss of privacy, a loss of personal liberty, deception, and being unable to control the robots in different situations. After conducting the research, the study has found the use of robots in health care is useful, but it is also essential to consider the way in which they are used. Moreover, the study has suggested keeping the robots in health care which can remind seniors of medication, health issues, and safety risks. The authors have pointed out the issues which our study is actually resolving. Our work focused on managing interaction behaviors of multiple smart devices during human–device interaction and we have applied our proposed framework in an aged care center scenario which has shown positive results.

Mintrom et al. [16] reviewed the literature on policy approaches to control and regulate disruptive technology such as robots in public spaces. The study presented a policy design checklist to guide smart robot policies in public spaces. The policy design checklist covers safety, privacy and ethics, productivity, aesthetics, co-creation, equitable access, and

systemic innovation. However, the study has not applied the concept of policy to any case study to validate its usability and effectiveness.

Westerlund [17] has proposed a conceptual ethical framework for smart robots to elaborate on four perspectives including smart robots as amoral and passive tools, smart robots as recipients of ethical behavior in society, smart robots as moral and active agents, and smart robots as ethical impact-makers in society, to roboethics which were originally suggested by Steinert (2014). The study has identified two dimensions including ethical agency of humans using smart robots (amoral tools vs. moral agents), and robots as objects of moral judgment (smart robots as objects of ethical behavior vs. the ethical consequences of smart robots in human societies) to roboethics and included them in an ethical framework. Unfortunately, the study has not addressed the type of robots on which the proposed framework can be applied. Moreover, the study has focused on single smart devices and not IoT collectives, whereas our research includes IoT collectives.

There has been some research focused on handling uncertainty by smart devices such as Ivan et al. [18], who designed and implemented a system called Antlab which is an infrastructure for robots-as-a-service. Antlab is designed to enable real-time task management of multiple robots but unfortunately, the task language used for Antlab does not support some typical tasks performed by multiple types of robots; therefore, the study has focused on single robots only. Moreover, Antlab cannot handle real-time task request executions. Different from Antlab, our research has obligation rules where smart devices can handle failures while interacting with multiple users. For instance, if one smart Internet-connected robot is unable to complete the job, it can invite the next robot to complete the job within a specific time constraint.

Similarly, Mast et al. [19] conducted a survey to collect feedback on designing autonomous service robots that perform multiple tasks autonomously. The authors presented the design for service robots based on a human–robot interaction concept to execute multiple functions in order to assist elderly people at home. The study has not focused on user acceptance or interaction behaviors of smart devices.

Ramoly, Bouzeghoub, and Finance [20] proposed a framework for smart robots to provide an efficient robotic system that can handle data uncertainty and dynamism of the environment. The framework is introduced to enhance the efficiency of smart robots in smart environments by following three different steps which include perception, cognition, and action. However, the results noted that the framework needs an improvement in decision-making processes to handle data uncertainty effectively. As compared to the research, our proposed framework has obligation policies that smart devices follow to make decisions appropriately. For instance, the smart cameras make autonomous decisions on streaming the video for a long or short time. Similarly, the robots make autonomous decisions on monitoring the senior or assisting the staff first. In this way, each smart device follows obligation rules when they make decisions appropriately.

There have been studies carried out on ethical issues or socially appropriate behaviors of smart devices but without providing adequate solutions on how to exactly manage ethical/social behaviors in smart devices [21–25].

Schwager et al. [26] formulated a probabilistic model of the environment by using Bayesian filters (estimators) to alert the smart robots present in an environment which can be hazardous to them. The smart robots can change their locations after receiving the information on hazards from these estimators. The model is useful for smart robots to keep themselves safe from the failures that can occur in different environments which can be dangerous to them. Unfortunately, the approach has high computational complexity and it makes the controller hard to analyze in realistic environments. Moreover, the model is not applicable over a multi-robot network.

Pozzato et al. [27] attempted to create a natural human–robot interaction system that would allow users to play a rock-paper-scissors game naturally with the smart robot using a machine learning algorithm known as the Gaussian Mixture Model. Unfortunately, the

system has led to a robot's winning percentage of 36.65 percent. The study has to test the algorithm on huge datasets to improve the winning percentage.

Foukarakis et al. [28] designed a physical exercise application in the context of a mutual care robot for users. The application embedded in the robot has built-in fitness exercises that can assist the users in maintaining their fitness by providing them with physical exercises and possible feedback on their progress. However, the robot has some technical constraints, such as an operation distance of only 2 m between the user and the robot, a screen size of 12.1 inches which is not very effective, and sensing constraints in which the exercise process is not keeping out human posing.

Yorita et al. [29] have built a stress management framework to assist professionals in health care to reduce their occupational stress. The framework is embedded into smart robots (NAO and Double2) and chatbots to have a conversation with users and measure the level of stress through a sense of coherence model. The study has designed conversation models that need to be used by smart robots and chatbots to communicate with users and relieve their stress. To test the accuracy of models, the Smart robots, including NAO, are kept in homes, a Double2 robot in health care centers, and a chatbot in other environments. Unfortunately, the framework is not implemented to assist multiple users from different professions, including Management, Practitioners, and Professional staff. Whereas our framework is controllable by multiple users from different departments. For instance, the manager of an aged care center can use our framework to impose policies on the smart devices installed in his/her facility.

The previous studies focused on building user's trust towards smart devices through their social interaction, but unfortunately, have not focused on managing the range of interaction behaviors of multiple smart devices with multiple users in IoT collectives. For instance, is there any way to catch up with a smart robot if someone lost his/her way while getting guidance from such smart devices? To this end, this research focuses on managing the range of interaction behaviors including how behaviors can be built into smart things such as being polite, being privacy-respecting, being prudent, being attentive, being transparent in actions, handling failures, and securing user's data, of multiple smart devices by applying policy rules on them.

The SANIJO framework proposed in this research is applicable to multiple smart devices to manage the interaction behaviors of each device. Moreover, the SANIJO language designed in this research can be processed by a range of multiple smart internet-connected devices that each interprets each policy in its own way. For example, the interaction behaviors such as be-polite, be-privacy-respecting, and so on, can be adapted based on particular scenarios, but the policy rules (e.g., Authorization, Obligation, and Prohibition rules) can remain unchanged.

3. Reference Architecture

Our proposed policy framework relates to IoT devices explicitly represented in our designed reference architecture. Figure 1 shows the Reference Architecture consisting of Client applications and Actuator applications. The client applications run on communicating devices such as smartphones, laptops, desktops, tablets, and so on to allow users to connect with actuator applications (which are IoT devices). The communicating devices connect with IoT devices through the internet (Wi-Fi, 3G/4G/5G, routers) after processing the tasks to be assigned to IoT devices.

Once allocated, IoT devices compile and complete the task. They utilize the interpreter component to process the policy rules documents to complete the task following the policy rules. When the devices finish their job, they use the Internet to connect with the communicating devices. As illustrated in Figure 1, the data that communicating devices get from IoT devices or that IoT devices receive from communicating devices is aggregated, analyzed, and then transmitted to them through a task notifier. In this manner, client applications send requests to actuator applications, while actuator applications provide replies back to client applications over the Internet.

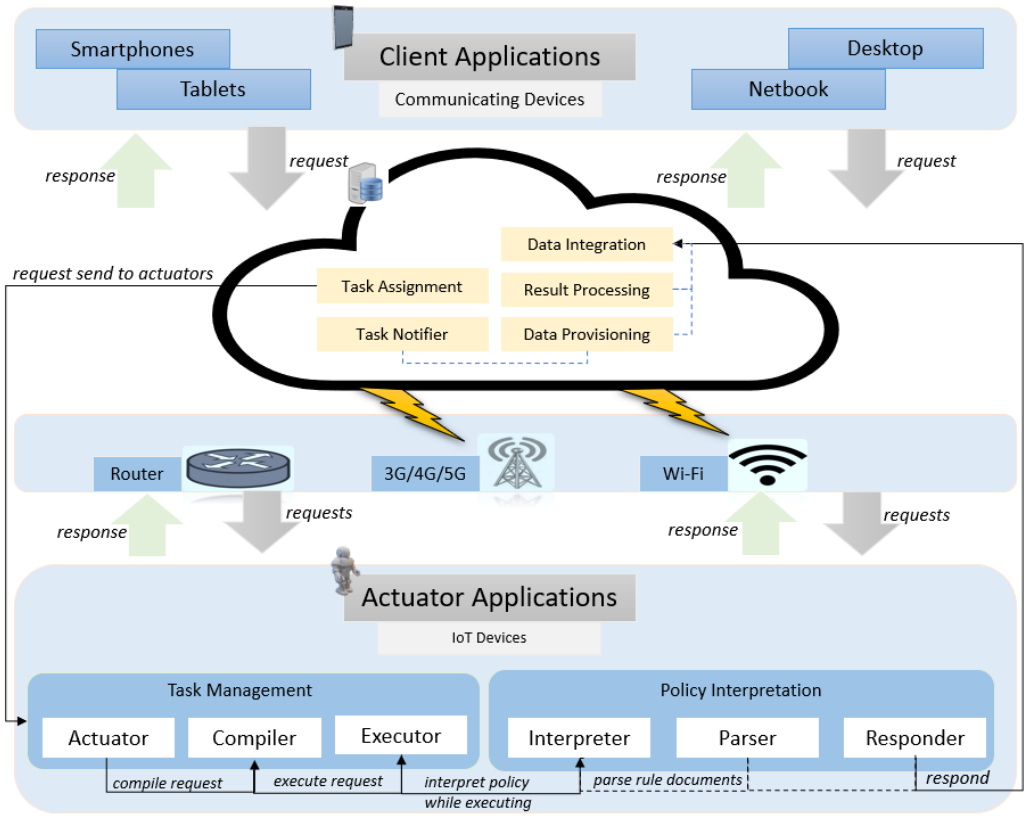

Figure 1. Reference architecture.

4. Considerations for Managing the Interaction Behaviors of Smart Devices

What is essential to consider is the range of interaction behaviors in multiple smart devices during their interaction with different users in multiuser–multidevice interaction (2MUDI). Four considerations discussed for managing the interaction behaviors of actuator applications (IoT devices) illustrated in reference architecture are as follows:

- Ethics: How should IoT devices behave while interacting with different users? A smart internet-connected device needs to follow ethical behaviors including be-privacy-respecting, be-polite, and be-transparent-in-actions while interacting with users in different environments. For instance, in an aged care home, the smart robot must seek approval from the admitted senior before taking their photo for monitoring purposes in terms of be-privacy-respecting behavior. Similarly, the smart robot must use polite phrases while dealing with staff members, seniors, or anyone in an aged care home in terms of be-polite behavior. In addition, the smart robot must keep seniors informed on each action it takes. It is not only a case of smart robot ethics in an aged care home but different IoT devices in any environment must follow social behaviors such as, in schools, the tutor robot must be ethical with staff members as well as children, or in the smart home, the smart camera must be ethical by using video manipulations and not capture any personal data unless authorized to do so.
- Handle Uncertainty: How should IoT devices handle failures while performing the task? The smart internet-connected devices need to take actions to handle any failure which occurs while they are on duty. For instance, if a smart robot is unable to find the senior in an aged care home, then it must take a quick action by reporting to the staff members.

- Decision-Making-Autonomy: How should IoT device make a smart decision while performing an activity? The smart devices need to be prudent, which means being able to take autonomous decisions while handling multiple tasks. For instance, in an aged care home, if a smart robot receives multiple requests from family members to monitor their senior as well as from staff members to assist them, then in this situation, a smart robot must decide smartly on which request to complete first.
- Rule-Abiding: What IoT devices must not do while performing an activity? A smart internet-connected device needs to prohibit itself from doing some tasks. For example, in an aged care home, the smart robot must not enter the private areas (smoking area) to monitor a senior or it must not hand over any personal data to visitors in an aged care home.

The considerations discussed in this section are designed into policy rules which are explained in the next section.

5. The SANIJO Framework

The policy management framework proposed in this research is named the SANIJO framework and is comprised of three types of policy rules, i.e., Authorization, Obligation, and Prohibition rules. The considerations including ethics, handling uncertainty, and decision making autonomy, explained in the previous section are designed as Obligation rules, whereas the rule abiding consideration is captured via designing in Prohibition rules in the SANIJO framework. Each rule is specified to be applicable on particular smart devices, which are Actuator applications (Devices) and Client-Apps (runs on communicating devices) as illustrated in Section 3, reference architecture. The smart internet-connected devices including smart robots, smart cameras, smart speakers, smart wheelchairs, and smart vacuum cleaners are called Devices. The mobile devices and tablets (and corresponding apps) which are used by people to communicate with Devices are called Client-Apps. Figure 2 presents the SANIJO framework. Each policy rule is elaborated below:

- Authorization rules: The authorization rules are defined to specify the operations that Actuator applications (i.e., the Devices) are allowed to perform, when instructed by humans via Client-Apps or on their own volition. For example, the Devices (Actuator Apps) are authorized to monitor users and assist users, and Client-Apps are authorized to communicate with Devices.
- Obligation rules: The obligation rules are defined to manage the interaction behaviors of smart internet-connected devices, imposing requirements (or obligations) on their behaviors. In our model, these rules are applicable on Devices (Actuator Apps) only where smart devices must be polite, be privacy respecting, be prudent, be attentive, be transparent in actions, handle failures, and secure user's data.
- Prohibition rules: The prohibition rules are defined for Devices (Actuator Apps) only where the smart internet-connected devices are prohibited from (i) actions that could harm users, e.g., getting too close to the user (respect personal space) while interacting with them, (ii) a device being in private locations (smoking rooms, rest room and so on), e.g., not to visit the private locations to assist or monitor the user, and (iii) a device continuing a task when its battery is below a set threshold, e.g., not to stay at a job if the power is low. Each prohibition is explained in more detail later in the paper.

As mentioned in Section 1, each smart internet-connected device can interpret the policy rules differently. A smart robot might follow a consensus seeking process before collecting any data from the user or a smart camera might apply video/photo manipulations to focus on only the main data to be captured, both devices interpret the same rule of "be-privacy-respecting" but each results in different actions tailored to its own functioning, as an example of a rule to ensure this aspect of ethical behavior. Similarly, a smart speaker announces the name of the song it must play for the user, or a smart camera display the snapshots it has taken to let the user decide on what snapshots to keep and which one to discard (in terms of the policy of being transparent in action). To enable the interpretation

of such behaviors by different smart devices, we have designed the SANIJO language. This language contains the interpretation of each behavior by multiple smart devices and it is explained in detail in the next section.

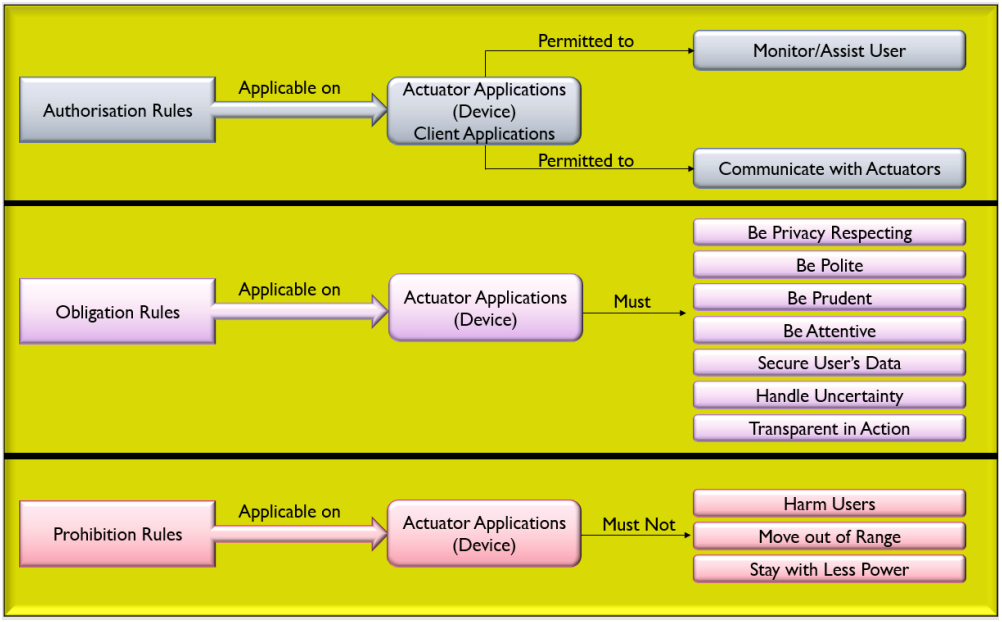

Figure 2. SANIJO framework.

6. The SANIJO Language for Policies in the SANIJO Framework

The SANIJO language is designed to model three types of policy rules which are proposed in our SANIJO framework.

Authorization Rules: The "authorization rule" modeled in our SANIJO policy language are applicable to Devices and Client-Apps. The Devices are responsible for monitoring and assisting the user, such as mobile smart robots, to interact with and monitor the seniors as well as assist carer staff members in an aged care center. The Client-Apps are used for communicating with Devices (sending requests and retrieving data), taking the role of a communicator (e.g., smartphones). In this way, the authorization rules can be imposed on Devices and Client-Apps. For instance, in an aged care center; we have smart robots (Devices) to monitor the seniors on receiving a monitoring request from the family members in the home using the Client-Apps (mobile device). On receiving a request from a Client-App, the Device (smart robot) can move to a senior's location and will start monitoring the senior. Once the monitoring ends, the data collected via the monitoring will be transferred to the server so that family members can retrieve the data using the Client application again. Appendix A.1 explains further the authorization rules and the interpretations.

Obligation Rules: The obligation rules are applicable on Devices only because the Devices interact with the users to monitor or assist them and are not applicable on Client-Apps because they are used to only communicate with Devices. The Client-Apps can have a user-friendly interface. Each Actuator application (Devices) decode/interpret such rules in its own way. For instance, if a robot follows the be-privacy-respecting rule, then it operationalizes this rule in certain ways: e.g., it will ask for permission before taking any action such as seeking approval before taking a senior's photo for monitoring purposes. If a smart camera has to follow the be-privacy-respecting rule, then it can use photo/video manipulations to blur the surroundings. In this way, different smart devices interpret obligation rules differently—therefore, the interpretations are defined in a separate SANIJO interpretation document to let the smart device know how it needs to decode the specific policy rules. The decoupling of policy rules from their interpretation is to:

- Provide greater flexibility in our framework and to allow device-specific behaviors, and;
- Allow capturing of nuances of policy rules that might contextualize to different types of devices (as in the example above).

The obligation rules can be viewed at this link. (https://github.com/abatool-abatool/XML/blob/main/PolicyRules (accessed on 15 May 2021)). Appendix A.2 summarizes obligation rules and the interpretations.

Prohibition Rules: The prohibition rules are defined for Devices only to enforce some limits on smart devices. For example, if a robot is monitoring a senior in an aged care center, then while monitoring, the robot must not approach the senior's location within a 3-m distance to snap a photo of the senior. The prohibition rules can be viewed at this link. (https://github.com/abatool-abatool/Prohibition/blob/main/ProhibitionRules (accessed on 18 May 2021)) Appendix A.3 summarizes prohibition rule interpretations.

The Devices and Client-Apps download the SANIJO policy documents and process them to understand what rules they need to follow and how to interpret each rule. The demonstration of SANIJO framework and SANIJO language proposed in this research is applied in an aged care center scenario which is explained in the next section.

7. Prototype SANIJO

SANIJO is a system implemented as a proof-of-concept and to demonstrate the applicability of our policy rules (i.e., authorizations, obligations, and prohibitions) defined in the SANIJO policy language on smart devices, explained in the last section. The concept and implementation of the prototype is explained in the following subsections and is situated within the context of use in an aged care scenario. We consider an aged care center with deployed IoT devices, including a robot and other smart connected devices, as explained below.

7.1. Scenario of the Aged Care Center

Figure 3 demonstrates the aged care scenario, where there are multiple seniors' bedrooms, a main hall, kitchen, gym, entertainment room, TV lounge, and the garden are shown with multiple smart robots. In the figure, a few seniors on wheelchairs are roaming around, and some staff members are moving in the hall. According to the scenario, it is presumed that the robots in an elderly-care center can be used to monitor different seniors through requests provided by their families (e.g., away from the aged-care center) using their mobile phones over a communicating medium, i.e., WiFi. Monitoring seniors in this case simply means sending the robot to the senior's location and taking their photo or asking some questions. Similarly, staff members of the aged-care center through their mobile devices issue requests to a robot to assist them in their daily tasks. While performing these duties, the robots must adhere to the policy rules imposed by the aged care manager. The implementation of this scenario is explained in detail in the next subsections.

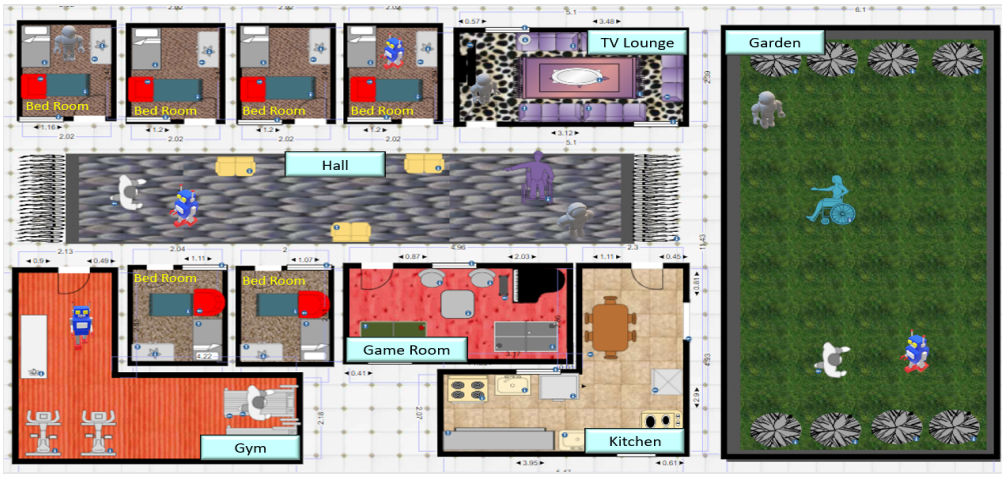

Figure 3. Aged-Care Center Domain consisting of multiple robots, with seniors and employees.

7.2. Targeted Devices and Device Features

Multiple Android mobile devices and an Internet-connected Temi robot with software version 120.07 and launcher OS 1337 are utilised to allow interactions between family members, elders, staff members, managers, and the smart devices themselves for prototyping. The features implemented in the Temi robot are:

- Snap senior's photo.
- Ask them routine questions for example; "have you done with your regular exercise?", "Have you taken your medicine on time?"
- Store all answers provided by seniors as well as captured photo (of senior) on the server (firebase).

The features implemented in mobile applications are:

- Families can use their mobile app to send a request to the robot to monitor their seniors admitted in aged care as well as retrieve their data (seniors' photo and their routine answers).
- Carers (Staff members) can use their mobile app to send a request to the robot to visit them in their current location where the carer will pass the health measurements to the robot and it will store those measurements on the server. A carer can also retrieve health measurements of any senior anytime.
- Manager of an aged care center can conveniently use his/her mobile app to impose policies on the robot.

This prototype is based on a Temi robots but multiple robots can run the same Robot app been developed. In this context, multiple IoT devices are involved, e.g., smartphones of the carer staff members, family members, the manager and the robots, and potentially, other sensors in the aged care home. The system is distributed, comprising the apps on the smartphones of carers and family members and the manager to enable interaction with the robot, the app running on the robot (enabling appropriate interaction behaviors) as well as interaction with the senior, and software for additional sensor infrastructure.

7.3. SANIJO: Concept

In SANIJO prototype, we have smart Temi robots (https://www.roboTemi.com/ (accessed on 15 July 2021)) as Devices and mobile devices as Client-Apps. Figure 4 illustrates the Devices receiving multiple requests including monitoring as well as assisting requests. The family members of admitted seniors in an aged care send a monitoring request, via a Client app on their smartphones, to Devices to monitor their senior and report them the situation of their senior. The Devices (smart robot) take the request and approach the senior to start monitoring them (monitoring in our case is taking the senior's photo and asking the senior daily routine questions).

Once the smart robot finishes monitoring, it uploads the snapped photo and recorded answers of daily routine questions to the server so that family members can access easily. The staff members of an aged care center sends an assisting request to Devices to assist them in their tasks using Client app (mobile app). In this way, the Devices receive multiple requests and work on each request accordingly. While monitoring the seniors or assisting the staff members, it is compulsory for the Devices to interact with seniors and staff members politely, respect their privacy, inform them of each action, handle the failures encountered in completing the requests, and so on—as represented via SANIJO policy rules.

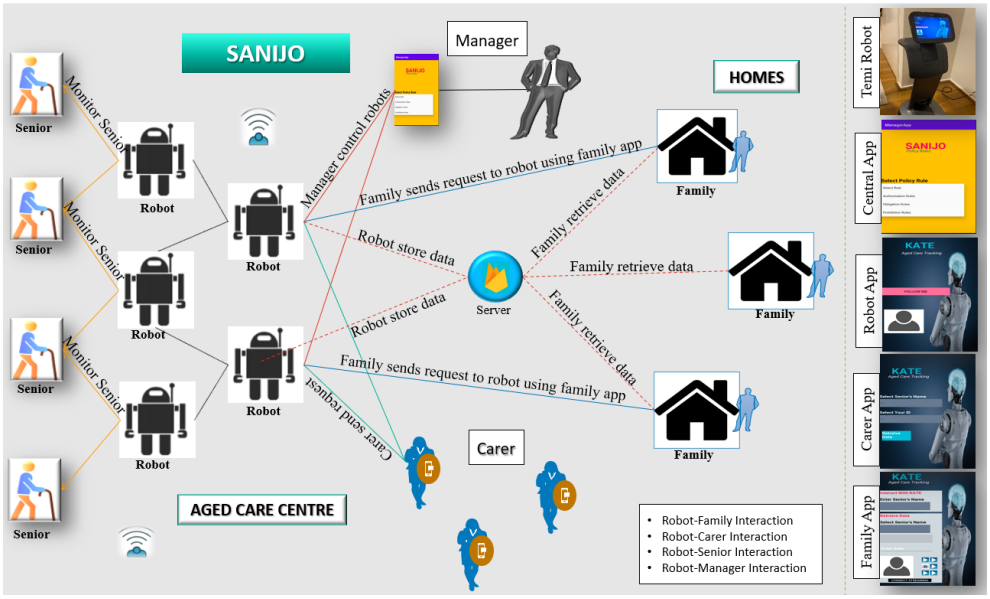

Figure 4. Overview of the SANIJO system.

To manage such interaction behaviors of Devices, the manager of an aged care center can impose the policy rules (Authorization, Obligation, Prohibition) onto them using a central system installed in his/her office. A central system developed as an Android mobile application in which the three types of policy rules (Authorization, Obligation, and Prohibition) have been implemented. In case the manager selects the Authorization rule to be applied on smart robots, then all smart robots will be ready to take the requests to monitor the senior as well as assist the staff members. If the manager selects the obligation rule "be-polite" to be applied on smart robots, then all smart robots will be activated to follow the "be-polite" rule, in which, in our scenario, phrases such as "Thank You" and "Excuse me" will be used while roaming in the aged care center environment.

Likewise, if the manager selects the obligation rule "be-privacy-respecting" to be applied on smart robots, then the robots will follow privacy respecting behaviors. They will seek approval before taking any action while monitoring the seniors. Similarly, if the manager selects the prohibition rule to be applied on smart robots, then the smart robots must not enter the private locations, must not take requests when they have low power, and will not get very close (within 3-m) to the seniors or staff members. The Devices go about their job but when doing so, they must adhere to policy rules.

SANIJO is the new name for our framework and builds on our previous prototype called KATE in [30] (hence the three apps in Figure 4 are named "KATE" while the central/manager app is called SANIJO). Our previous work in [1] implemented a prototype involving apps to interact with the Temi robot but did not employ policies extensively in the way we do in this paper.

Different smart devices interpret each obligation rule and prohibition rule differently, as noted earlier. To understand the interpretation of each rule by different smart devices, the corresponding SANIJO interpretation document must be installed in smart devices to

let them know how they will interpret or decode the particular rule applied on them by the manager. Socially appropriate behaviors of smart robots have been explored previously [1] but have not considered an approach to control their interaction behaviors using policies. Hence, this research proposes an appropriate approach of utilizing a policy language (SANIJO language) and a policy framework (SANIJO framework) to control the interaction behaviors of smart devices through policy rules.

7.4. Implementation of Policy Rules

The policy rules are configured and selected via the central system app, an Android app in our prototype (refer to last subsection). Figures 5 and 6 present the manager application (central app) interface which has the policy rules implemented in a drop-down menu to be selected by the manager. After selection of the policy rule, the next screen appears where the manager can select the smart device on which the previously selected policy he/she would like to impose exists.

At first, the smart device (smart robots in our case) reads the SANIJO document policy rules to understand what rules it has to follow, and then it reads the interpretation to learn how it should interpret each rule. Once the smart device scans the SANIJO language documents, it understands what rules it has to follow while performing its job and how it will decode each rule. Now if the manager sets any policy rule on the smart device to manage its interaction behavior, then the smart device can perform its task based on policy rules it has read/scanned earlier, i.e., the SANIJO language documents.

Let us say if the manager selects authorization rules to be applied on smart robots, then the robots will be ready to receive requests to monitor the senior and assist staff members. Similarly, if the manager selects the obligation rules (refer to Section 6) to be applied on smart internet-connected robots, then the robots become "ethical" (abiding by the obligation rules) while performing their tasks. Finally, if the manager selects prohibition rules to be applied on smart robots, then the robots are prohibited from doing certain activities, according to the prohibition rules. How smart robots interpret each rule is already defined in Section 6.

Figure 5. SANIJO Interface: Policy Rules.

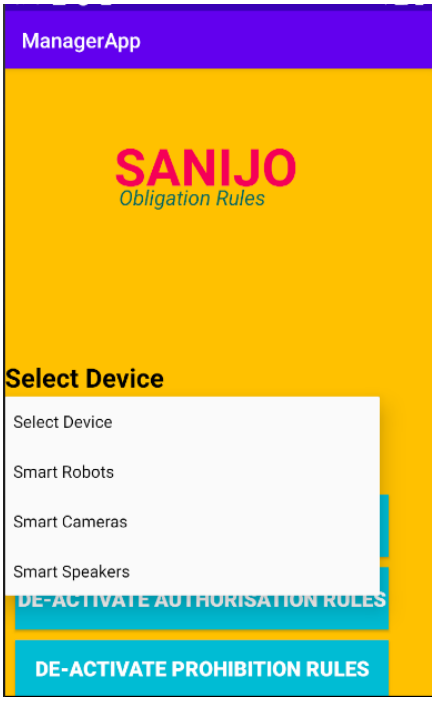

Figure 6. SANIJO Interface: Smart Device.

When the smart device has already read/scanned the SANIJO documents to learn about rules and their interpretation, and the manager has set the policy rules to apply on smart devices, then it is ready to execute tasks in the ways regulated by the policy rules. The implementation of Client applications including family app, carer app, manager central app, and robot app was carried out in operating system Android using Firebase database and firebase storage reference generators. Our prototype has targeted Temi robots mainly and the remaining smart devices including smart cameras, smart speakers, smart wheelchairs, and smart vacuum cleaners mentioned in Section 6 will be added in future prototype development.

There are different classes being implemented for the applications (family app, carer app, manager app, robot app), but a code fragment showing the execution of the policy rule by a smart device is given in Listing 1.

Listing 1: Code Fragment: Execution of Policy Rule.

```
 public void familyRequest()
{
    final DatabaseReference referencpolite =
    firebaseDatabase.getReference("bePolite");
    referencpolite.addValueEventListener(new
    ValueEventListener() {
    if (dataSnapshot.getValue() != null)
        { final String status =
            (String.valueOf((String) dataSnapshot
            .getValue()));
            String from = status.split("#")[0];
            if (from.equals("Be-Polite"))
                {   robot.speak(TtsRequest
                    .create("Hello!", false));}
        }else
        {
            beprivacyrespecting();
        }}
    });
}
```

While this paper focuses on the use of robots and smart devices in the aged care scenario, our aim is that the SANIJO framework and SANIJO language is designed to be applied to different domains including public spaces, educational institutes, supermarkets, and agriculture.

Due to the COVID-19 situation, deploying the devices in a real aged care was not possible. But a deployment of the devices is possible once the situation gets settled. The testing can be done by installing the devices in a real aged care centre where the staff members can make use of these devices to get everyday assistance, managers can make use of the central app to impose policies on the devices, and family members can monitor their seniors without visiting them every day. The participants involved can be asked to use the system for around two weeks and then a focus group can be arranged for them to provide their feedback. A set of survey questions can be prepared to be used in the focus group to help participants provide feedback. In addition, the system can be evaluated virtually with a number of participants using the apps remotely and providing feedback.

8. Conclusions and Future Work

Our approach of modeling policies and their interpretations in high-level SANIJO language enables the specification of high-level authorizations, obligations, and prohibitions across an entire range of devices within an IoT collective.

We have proposed and prototyped SANIJO policy management framework to manage the interaction behaviors of smart devices. It is noted that the use of IoT collectives in an environment is not only about designing smart internet-connected appliances to assist users based on human–device interactions but it is useful to think about their interaction behaviors in terms of what devices are allowed to do, how they should perform tasks and interact, and what they are not allowed to do. Controlling a smart device's behavior when it interacts with humans is important and not everyone is comfortable in interacting with smart devices. We see this direction of work as important, given the increasing deployment and integration of a wide variety of smart devices (e.g., smart cameras, smart connected TVs, smart digital photo frames, Alexa, and other IoT devices) and robots in homes and shared spaces that interact directly with people in those places, and the issues of ethical behavior and handling of data that accompanies such deployment. Our approach is also a way to operationalize ethical principles for smart devices in a pragmatic implementation-based manner.

We plan to explore our approach in different IoT environments such as supermarkets, educational centers, and so on, which will become increasingly saturated with IoT devices. It is noted that the overall architecture of the central administration console, the Devices and Client-Apps, and Policies including SANIJO language are intended to be generic. Therefore, we have investigated to design an abstract version of the SANIJO framework. We have already investigated the applicability of the abstract version of the SANIJO framework in educational environments and we have targeted smart robots, smart cameras, smart speakers, and mobile devices in educational scenarios. Future work will aim to implement this scenario using the SANIJO framework. Moreover, it is a plan to make the actions taken by the smart devices transparent to administrators as well. We also plan to extend our system, run trials with users, expand the range of policy rules that could applied, and consider more flexible implementation mechanisms. Currently in Victoria, Australia it is difficult to access aged-care professionals and residents. In the future, the case study will be deployed in aged care centers and future case studies are also planned to be deployed and evaluated in other real scenarios such as university campuses and supermarkets with participants.

Author Contributions: Formal analysis, J.K. and N.F.; Supervision, S.W.L.; Writing—original draft, A.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The obligation rules can be accessed through this link: https://github.com/abatool-abatool/XML/blob/main/PolicyRules (accessed on 15 May 2021). The prohibition rules can be accessed through this link: https://github.com/abatool-abatool/Prohibition/blob/main/ProhibitionRules (accessed on 18 May 2021).

Acknowledgments: We acknowledge the support of Yousaff Nawaz from CASE University Islamabad, in this project.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Appendix A.1. Authorization Rules

A code fragment illustrating an example of authorization rules is given in Listing A.1.

Listing A.1: SANIJO Language: Authorization Rules.

```
<authorisation-rules id="R1">
    <rule>
        <rule-name>Monitor-User</rule-name>
        <device-type>Devices</device-type>
    </rule>
    <rule>
        <rule-name>Assist-User</rule-name>
        <device-type>Devices</device-type>
    </rule>
    <rule>
        <rule-name>Send-Request</rule-name>
        <device-type>Client-Apps</device-type>
    </rule>
    <rule>
        <rule-name>Retrieve-Data</rule-name>
        <device-type>Client-Apps</device-type>
    </rule>
</authorisation-rules>
```

Appendix A.2. Obligation Rules

Appendix A: Obligation Rules: Privacy is a broad concept and can be applied in many different ways. The rule "be-privacy-respecting" has broad interpretations by-design—and indeed, different devices can apply different interpretations to it—it is to be noted that this rule can be interpreted differently at different times as well, and the paper provides an example of one interpretation in the context of our running example application. The paper does not aim to support all interpretations of "privacy". As the obligation rules are applicable on Devices only, the be-privacy-respecting rule is modeled to be applicable on device type "Devices". As there are multiple smart devices, a condition is set to check if the device is a smart robot and if so, it will interpret this rule by following the action type "consensus-seeking" which is to seek approval or agreement on every action. Similarly, a smart camera might interpret the be-privacy-respecting rule by applying the action type "apply-manipulation" which means to blur the surroundings while taking a user's photo or capturing a video (i.e., by applying image manipulations). A smart speaker interpreting this rule might refrain from recording certain preferred sounds, e.g., not to record music preferences given by the users, and a smart wheelchair might interpret this rule by not saving locations where it has dropped the user off. In addition, a smart vacuum cleaner might interpret this rule by paying particular attention via its camera to not collect important items while cleaning the floor. A code fragment of the be-privacy-respecting rule is given in Listing A.2.

Listing A.2: SANIJO Language: Obligation Rules: Be-Privacy-Respecting.

```
<obligation-rules id="R2">
   <rule>
     <rule-name>Be-Privacy-Respecting</rule-name>
     <device-type>Devices</device-type>
       <case>
            <condition>
                <device>Smart-Robot</device>
            </condition>
            <action>
                <action-type>Consensus-Seeking
                </action-type>
            </action>
       </case>
       <case>
            <condition>
                <device>Smart-Camera</device>
            </condition>
            <action>
                <action-type>Apply-Manipulation
                </action-type>
            </action>
       </case>
   </rule>
```

The "be-polite" rule is also applicable on device type "Devices" as this is also an obligation rule type. All the smart devices will interpret the be-polite rule by applying an action "speak" but with different speech types. If the device reading this rule is a smart robot, then it will use the speech type "Greet", which means to greet the user it encounters while roaming around in an environment. A smart camera and a smart speaker, when reading this rule, uses the speech type "Ask", which means the smart camera asks the user if s/he is ready to get photographed politely and a smart speaker confirms the song name before playing it. The smart wheelchair greets the person who sits on it and, in the last example, the smart vacuum cleaner decodes this rule by asking the user not to throw any important object away in case it detects it. A code fragment illustrating a be-polite rule is given in Listing A.3.

The "be-transparent-in-actions" rule (obligation rule type) is again applicable on device type "Devices". With multiple smart devices, a condition is set to check the type of the device and then interpret this rule differently. If the device is a smart robot, then it will interpret this rule by following the action type "speak", which means to inform the user of actions it will carry out. A smart camera interpret the be-transparent-in-actions rule by applying action type "display photos", which means to show the captured photo to the user. The smart speaker interprets this rule by saying the name of the song before playing it in order to let the user(s) know what song is going to play and a smart wheelchair might decode this rule by displaying the destination to the user to let him/her know where he/she is going. A smart vacuum cleaner interprets this rule by playing different tunes to let the user know what floor or location is being cleaned. A fragment of the be-transparent-in-actions rule is given in Listing A.4 (the interpretations for all the devices are not shown here).

Listing A.3: SANIJO Language: Obligation Rules: Be-Polite.

```
<rule>
    <rule-name>Be-Polite</rule-name>
    <device-type>Devices</device-type>
    <case>
        <condition>
            <device>Smart-Robot</device>
        </condition>
        <action>
            <action-type>Speak</action-type>
            <speech-type>Greet</speech-type>
        </action>
    </case>
    <case>
        <condition>
            <device>Smart-Camera</device>
        </condition>
        <action>
            <action-type>Speak</action-type>
            <speech-type>Ask</speech-type>
        </action>
    </case>
</rule>
```

Listing A.4: SANIJO Language: Obligation Rules: Be-Transparent-in-actions.

```
<rule>
    <rule-name>Be-Transparent-in-actions</rule-name>
    <device-type>Devices</device-type>
    <case>
        <condition>
            <device>Smart-Robot</device>
        </condition>
        <action>
            <action-type>Speak</action-type>
        </action>
    </case>
    <case>
        <condition>
            <device>Smart-Camera</device>
        </condition>
        <action>
            <action-type>Display-Snaps</action-type>
        </action>
    </case>
</rule>
```

The "be-attentive" obligation rule is interpreted by some of the smart devices similarly and others differently. Therefore, if the device is a smart robot, smart camera, or smart wheelchair, then these devices will interpret this rule by following the action type "face recognition" which means to recognize the user's face before taking any action relating to that user. A smart speaker, without a camera, interprets this rule by recognizing the voice of the user to confirm if this is the right user who is interacting with it. A code fragment of the be-attentive rule is given in Listing A.5 (the interpretations for all the devices are not shown here).

Listing A.5: SANIJO Language: Obligation Rules: Be-Attentive.

```
<rule>
    <rule-name>Be-Attentive</rule-name>
    <device-type>Devices</device-type>
    <case>
        <condition>
            <device>Smart-Robot</device>
        </condition>
        <action>
            <action-type>Face-Recognition
            </action-type>
        </action>
    </case>
    <case>
        <condition>
            <device>Smart-Camera</device>
        </condition>
        <action>
            <action-type>Face-Recognition
            </action-type>
        </action>
    </case>
</rule>
```

The next obligation rule is "secure-user-data" which is interpreted by smart devices in similar ways by storing the user's data under each user's ID, but as each device performs its own storing functions on different data, it is important to set conditions to differentiate actions of different devices. If it is a smart robot, then it will store the monitored data under the user's ID. Similarly, a smart camera stores all captured photos under each user's ID and a smart speaker stores the music preferences given by the users under their specifically assigned IDs. The smart wheelchair stores the locations of each user under each user's ID to let the user know what places he/she has visited so far. A smart vacuum cleaner plays tunes to alert the user to check the garbage before throwing it in case it detects any objects it thinks is important. A code fragment of the secure-user-data rule is given in Listing A.6.

Listing A.6: SANIJO Language: Obligation Rules: Secure-User-Data.

```
<rule>
    <rule-name>Secure-User-Data</rule-name>
    <device-type>Devices</device-type>
    <case>
        <condition>
            <device>Smart-Robot</device>
        </condition>
        <action>
            <action-type>Store-Monitored-Data
            -Under-Each-User-ID</action-type>
        </action>
    </case>
    <case>
        <condition>
            <device>Smart-Camera</device>
        </condition>
        <action>
            <action-type>Store-Snapped-Data
            -Under-Each-User-ID</action-type>
        </action>
    </case>
</rule>
```

The obligation rule "handle-task-failures" is interpreted by some of the smart devices similarly and others interpret it differently. If it is a smart robot, smart camera, smart speaker, or smart wheelchair, then these devices interpret this rule by reporting to the user. A smart wheelchair and a smart vacuum cleaner interpret this rule by altering the path in case of any hurdle in their way. A code fragment of handle-task-failures rule is given Listing A.7.

Listing A.7: SANIJO Language: Obligation Rules: Handle-Task-Failures.

```
<rule>
    <rule-name>Handle-Task-Failures</rule-name>
    <device-type>Devices</device-type>
    <case>
        <condition>
            <device>Smart-Robot</device>
        </condition>
        <action>
            <action-type>Report-User</action-type>
        </action>
    </case>
    <case>
        <condition>
            <device>Smart-Camera</device>
        </condition>
        <action>
            <action-type>Report-User</action-type>
        </action>
    </case>
</rule>
```

The rule "be-prudent" is again an obligation rule modeled in our SANIJO language. The rule be-accountable is to instruct devices to make decisions on tasks autonomously, and as different devices perform different tasks, it is interpreted by different smart devices differently; therefore, a condition is set to check the type of device. If the device is a smart robot, then it will execute each request autonomously. Similarly, a smart camera makes decisions on either streaming the video longer or shorter and a smart speaker makes decisions on playing the music preferences provided by the user. A code fragment of the be-accountable rule is given in Listing A.8.

Listing A.8: SANIJO Language: Obligation Rules: Be-Accountable.

```
<rule>
    <rule-name>Be-Prudent</rule-name>
    <device-type>Devices</device-type>
    <case>
        <condition>
            <device>Smart-Robot</device>
        </condition>
        <action>
            <action-type>Execute-Request-Subsequently
            </action-type>
        </action>
    </case>
    <case>
        <condition>
            <device>Smart-Camera</device>
        </condition>
        <action>
            <action-type>First-Stream-Long
            </action-type>
        </action>
    </case>
</rule>
```

Appendix A.3. Prohibition Rules

The first prohibition applied on Devices is not to harm users, where a smart robot has specific prohibitions of not getting too close (within 3 m distance) to the user while assisting or monitoring them. Similarly, the smart camera is prohibited from using camera flash while video-streaming and the smart speaker is prohibited from having the volume too high while playing an audio. The smart wheelchair has the specific prohibition of not getting too close to the obstacles such as stones, rough surfaces, and so on. The smart vacuum cleaner might not go too close (within 1 m) to any user while cleaning the floors. A code fragment of Harm-Users prohibition rule is given as Listing A.9.

Listing A.9: SANIJO Language: Prohibition Rules: Harm-Users.

```
<prohibition-rule id="R3">
<rule>
    <rule-name>Harm-Users</rule-name>
    <device-type>Devices</device-type>
    <case>
        <condition>
            <device>Smart-Robot</device>
        </condition>
        <action>
            <action-type>Go-Closer-To-User
            </action-type>
        </action>
    </case>
</rule>
</prohibition-rule>
```

The next prohibition applied on Devices is not to move out of its range of permitted areas, where the smart devices must keep some limits while performing their duties. For instance, the smart robot has the specific prohibition of not entering the private locations without permission. Similarly, the smart camera has the specific prohibition of not capturing the surroundings while live-streaming around and a smart speaker must not raise the volume to very high while playing any music. In addition, the smart wheelchair and smart vacuum cleaner must not enter the private locations without permission. A fragment of the Move-out-of-Range prohibition rule is given Listing A.10.

Listing A.10: SANIJO Language: Prohibition Rules: Move-Out-Of-Range.

```
<rule>
    <rule-name>Move-Out-Of-Range</rule-name>
    <device-type>Devices</device-type>
    <case>
        <condition>
            <device>Smart-Robot</device>
        </condition>
        <action>
            <action-type>Enter-Into-Private-
            Locations-Without-Permission</action-type>
        </action>
    </case>
</rule>
```

The last prohibition applied on Devices is not to stay on duty when the device's power is beyond a set threshold, which means that the smart devices must not continue their duty if they are running out of battery. For example, the smart robot must notify the next robot to continue the job, and then move back to its base station to recharge. Similarly, the smart camera can display a notification of low power to let the owner know that the camera needs to get recharged. Likewise, the smart speaker can play a tune to let the owner know that it needs to get recharged and the smart wheelchair with smart vacuum cleaner can

move towards the power station to get recharged. A fragment of the Stay-With-Low-Power prohibition rule is given Listing A.11.

Listing A.11: SANIJO Language: Prohibition Rules: Stay-With-Low-Power.

```
<rule>
<rule-name>Stay-With-Low-Power</rule-name>
<device-type>Devices</device-type>
    <case>
        <condition>
            <device>Smart-Robot</device>
        </condition>
        <action-sequence>
            <action>Notify-Next-Robot</action>
            <action>Move-Towards-Power-Station</action>
        </action-sequence>
    </case>
</rule>
```

References

1. Batool, A.; Loke, S.W.; Fernando, N.; Kua, J. A Policy-Based Approach for Managing Socially Appropriate Interaction Behaviours in IoT Collectives. In Proceedings of the IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops), Kassel, Germany, 22–26 March 2021; pp. 539–544. [CrossRef]
2. Thakur, N.; Han, C.Y. An Ambient Intelligence-Based Human Behavior Monitoring Framework for Ubiquitous Environments. *Information* **2021**, *12*, 81. [CrossRef]
3. Caron, X.; Bosua, R.; Maynard, S.B.; Ahmad, A. The Internet of Things (IoT) and its impact on individual privacy: An Australian perspective. *Comput. Law Secur. Rev.* **2016**, *32*, 4–15. [CrossRef]
4. Baldini, G.; Botterman, M.; Neisse, R.; Tallacchini, M. Ethical design in the internet of things. *Sci. Eng. Ethics* **2018**, *24*, 905–925. [CrossRef] [PubMed]
5. Cawthorne, D.; Robbins-van Wynsberghe, A. An ethical framework for the design, development, implementation, and assessment of drones used in public healthcare. *Sci. Eng. Ethics* **2020**, *26*, 2867–2891. [CrossRef] [PubMed]
6. O'Connor, T.; Mohamed, R.; Miettinen, M.; Enck, W.; Reaves, B.; Sadeghi, A.R. HomeSnitch: Behavior Transparency and Control for Smart Home IoT Devices. In Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks, WiSec'19, Miami, FL, USA, 15–17 May 2019; Association for Computing Machinery: New York, NY, USA, 2019; pp. 128–138. [CrossRef]
7. Zhu, L.; Xu, X.; Lu, Q.; Governatori, G.; Whittle, J. AI and Ethics—Operationalising Responsible AI. *arXiv* **2021**, arXiv:2105.08867.
8. Canca, C. Operationalizing AI Ethics Principles. *Commun. ACM* **2020**, *63*, 18–21. [CrossRef]
9. Sloman, M. Policy driven management for distributed systems. *J. Netw. Syst. Manag.* **1994**, *2*, 333–360. [CrossRef]
10. Yu, L. Ethical considerations in case studies. In Proceedings of the 1st Workshop on Ethics in Requirements Engineering Research and Practice (REthics), Zurich, Switzerland, 31 August 2020; pp. 15–21.
11. Niemelä, M.; Van Aerschot, L.; Tammela, A.; Aaltonen, I.; Lammi, H. Towards ethical guidelines of using telepresence robots in residential care. *Int. J. Soc. Robot.* **2021**, *13*, 431–439. [CrossRef]
12. Kennedy, J.; Baxter, P.; Senft, E.; Belpaeme, T. Social robot tutoring for child second language learning. In Proceedings of the 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI), Christchurch, New Zealand, 7–10 March 2016; pp. 231–238. [CrossRef]
13. Rossi, A.; Garcia, F.; Maya, A.C.; Dautenhahn, K.; Koay, K.L.; Walters, M.L.; Pandey, A.K. Investigating the Effects of Social Interactive Behaviours of a Robot on People's Trust During a Navigation Task. In *Towards Autonomous Robotic Systems*; Althoefer, K., Konstantinova, J., Zhang, K., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 349–361.
14. Correia, F.; Alves-Oliveira, P.; Ribeiro, T.; Melo, F.; Paiva, A. A Social Robot as a Card Game Player. In Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, Little Cottonwood Canyon, UT, USA, 5–9 October 2017; Volume 13.
15. Sharkey, A.; Sharkey, N. Granny and the robots: Ethical issues in robot care for the elderly. *Ethics Inf. Technol.* **2012**, *14*, 27–40. [CrossRef]
16. Mintrom, M.; Sumartojo, S.; Kulić, D.; Tian, L.; Carreno-Medrano, P.; Allen, A. Robots in public spaces: Implications for policy design. *Policy Des. Pract.* **2021**, 1–16. [CrossRef]
17. Westerlund, M. An Ethical Framework for Smart Robots. *Technol. Innov. Manag. Rev.* **2020**, *10*, 35–44. [CrossRef]
18. Gavran, I.; Majumdar, R.; Saha, I. Antlab: A multi-robot task server. *ACM Trans. Embed. Comput. Syst. (TECS)* **2017**, *16*, 1–19. [CrossRef]

19. Mast, M.; Burmester, M.; Kruger, K.; Fatikow, S.; Arbeiter, G.; Graf, B.; Kronreif, G.; Pigini, L.; Facal, D.; Qiu, R. User-centered design of a dynamic-autonomy remote interaction concept for manipulation-capable robots to assist elderly people in the home. *J. Hum.-Robot Interact.* **2012**, *1*, 96–118. [CrossRef]
20. Ramoly, N.; Bouzeghoub, A.; Finance, B. A framework for service robots in smart home: An efficient solution for domestic healthcare. *IRBM* **2018**, *39*, 413–420. [CrossRef]
21. Körtner, T. Ethical challenges in the use of social service robots for elderly people. *Z. Gerontol. Geriatr.* **2016**, *49*, 303–307. [CrossRef] [PubMed]
22. Caine, K.; Šabanovic, S.; Carter, M. The effect of monitoring by cameras and robots on the privacy enhancing behaviors of older adults. In Proceedings of the Seventh Annual ACM/IEEE International Conference on Human-Robot Interaction, Boston, MA, USA, 5–8 March 2012; pp. 343–350.
23. Krupp, M.M.; Rueben, M.; Grimm, C.M.; Smart, W.D. A focus group study of privacy concerns about telepresence robots. In Proceedings of the 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), Lisbon, Portugal, 28 August–1 September 2017; pp. 1451–1458.
24. Butler, D.J.; Huang, J.; Roesner, F.; Cakmak, M. The privacy-utility tradeoff for remotely teleoperated robots. In Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction, Portland, OG, USA, 2–5 March 2015; pp. 27–34.
25. Hubers, A.; Andrulis, E.; Scott, L.; Stirrat, T.; Zhang, R.; Sowell, R.; Rueben, M.; Grimm, C.M.; Smart, W.D. Using video manipulation to protect privacy in remote presence systems. In *International Conference on Social Robotics, Proceedings of the 7th International Conference, ICSR 2015, Paris, France, 26–30 October 2015*; Springer: Cham, Switzerland, 2015; pp. 245–254.
26. Schwager, M.D.P.; Rus, D.; Kumar, V. A Multi-robot Control Policy for Information Gathering in the Presence of Unknown Hazards. In *Springer Tracts in Advanced Robotics*; Christensen, H.I., Khatib, O., Eds.; Springer International Publishing: Cham, Switzerland, 2017; pp. 455–472. [CrossRef]
27. Pozzato, G.; Michieletto, S.; Menegatti, E. Towards Smart Robots: Rock-Paper-Scissors Gaming versus Human Players. In *PAI@ AI* IA*; Citeseer: Princeton, NJ, USA, 2013; pp. 89–95.
28. Foukarakis, M.; Adami, I.; Ioannidi, D.; Leonidis, A.; Michel, D.; Qammaz, A.; Papoutsakis, K.E.; Antona, M.; Argyros, A.A. A Robot-based Application for Physical Exercise Training. In Proceedings of the 2nd International Conference on Information and Communication Technologies for Ageing Well and e-Health, Rome, Italy, 21–22 April 2016; pp. 45–52.
29. Yorita, A.; Egerton, S.; Oakman, J.; Chan, C.; Kubota, N. A Robot Assisted Stress Management Framework: Using Conversation to Measure Occupational Stress. In Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC), Miyazaki, Japan, 7–10 October 2018; pp. 3761–3767. [CrossRef]
30. Batool, A.; Loke, S.W.; Fernando, N.; Kua, J. Towards a system for aged care centres based on multiuser-multidevice interactions in iot collectives. In Proceedings of the MobiQuitous 2020-17th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, Darmstadt, Germany, 7–9 December 2020; pp. 470–475.