

A Cloud-based Robot System for Long-term Interaction Principles, Implementation, Lessons Learned

Kaptein, Frank; Kiefer, Bernd; Cully, Antoine; Celiktutan, Oya; Bierman, Bert; Rijgersberg-Peters, Rifca; Broekens, Joost; Van Vught, Willeke; Van Bekkum, Michael; Demiris, Yiannis

DOI

[10.1145/3481585](https://doi.org/10.1145/3481585)

Publication date

2022

Document Version

Final published version

Published in

ACM Transactions on Human-Robot Interaction

Citation (APA)

Kaptein, F., Kiefer, B., Cully, A., Celiktutan, O., Bierman, B., Rijgersberg-Peters, R., Broekens, J., Van Vught, W., Van Bekkum, M., Demiris, Y., & Neerincx, M. A. (2022). A Cloud-based Robot System for Long-term Interaction: Principles, Implementation, Lessons Learned. *ACM Transactions on Human-Robot Interaction*, 11(1), Article 8. <https://doi.org/10.1145/3481585>

Important note

To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

A Cloud-based Robot System for Long-term Interaction: Principles, Implementation, Lessons Learned

FRANK KAPTEIN, Delft University of Technology, Netherlands

BERND KIEFER, paragon semvox GmbH, Germany

ANTOINE CULLY, Imperial College London, United Kingdom

OYA CELIKTUTAN, King's College London, United Kingdom

BERT BIERMAN, Produxi, Netherlands

RIFCA RIJGERSBERG-PETERS, Delft University of Technology, Netherlands

JOOST BROEKENS, Leiden University, Netherlands

WILLEKE VAN VUGHT and MICHAEL VAN BEKKUM, TNO, Netherlands

YIANNIS DEMIRIS, Imperial College London, United Kingdom

MARK A. NEERINCX, Delft University of Technology, Netherlands

Making the transition to long-term interaction with social-robot systems has been identified as one of the main challenges in human-robot interaction. This article identifies four design principles to address this challenge and applies them in a real-world implementation: cloud-based robot control, a modular design, one common knowledge base for all applications, and hybrid artificial intelligence for decision making and reasoning. The control architecture for this robot includes a common Knowledge-base (ontologies), Data-base, "Hybrid Artificial Brain" (dialogue manager, action selection and explainable AI), Activities Centre (Timeline, Quiz, Break and Sort, Memory, Tip of the Day, . . .), Embodied Conversational Agent (ECA, i.e., robot and avatar), and Dashboards (for authoring and monitoring the interaction). Further, the ECA is integrated with an expandable set of (mobile) health applications. The resulting system is a Personal Assistant for a healthy Lifestyle (PAL), which supports diabetic children with self-management and educates them on health-related issues (48 children, aged 6–14, recruited via hospitals in the Netherlands and in Italy). It is capable of autonomous interaction "in the wild" for prolonged periods of time without the need for a "Wizard-of-Oz" (up

B. Kiefer contributed to this work while he was with DFKI.

O. Celiktutan contributed to this work when she was with Imperial College London.

M. A. Neerincx is affiliated with both TNO and Delft University of Technology.

Yiannis Demiris was supported by a Royal Academy of Engineering Chair in Emerging Technologies. PAL has been co-funded by the Horizon 2020 Framework Program of the European Union under Grant Agreement No. 643783.

Authors' addresses: F. Kaptein, R. Rijgersberg-Peters, and M. A. Neerincx, Delft University of Technology, Van Mourik Broekmanweg 6, 2628 XE, Delft, Netherlands; emails: F.C.A.Kaptein@tudelft.nl, R.M.Peters@tudelft.nl, mark.neerincx@tno.nl; B. Kiefer, paragon semvox GmbH, Konrad-Zuse-Straße 19, 66459, Limbach, Germany; email: j_a_w_bone@gmx.de; A. Cully and Y. Demiris, Imperial College London, South Kensington, SW7 2AZ, London, United Kingdom; emails: a.cully@imperial.ac.uk, y.demiris@imperial.ac.uk; O. Celiktutan, King's College London, Strand, WC2R 2LS, London, United Kingdom; email: oya.celiktutan@kcl.ac.uk; B. Bierman, Produxi, Comeniuslaan 83, 3706 XC, Zeist, Netherlands; email: bert@produxi.nl; J. Broekens, Leiden University, Niels Bohrweg 1, 2333 CA, Leiden, Netherlands; email: joost.broekens@gmail.com; W. van Vught and M. van Bekkum, TNO, Kampweg 55, 3769 DE, Soesterberg, Netherlands; email: willeke.vanvught@tno.nl, michael.vanbekkum@tno.nl.



This work is licensed under a Creative Commons Attribution International 4.0 License.

© 2021 Copyright held by the owner/author(s).

2573-9522/2021/10-ART8 \$15.00

<https://doi.org/10.1145/3481585>

until 6 months online). PAL is an exemplary system that provides personalised, stable and diverse, long-term human-robot interaction.

CCS Concepts: • **Computer systems organization** → *External interfaces for robotics*; **Robotic autonomy**; • **Computing methodologies** → *Intelligent agents*; *Cognitive robotics*;

Additional Key Words and Phrases: Cloud-based robots, long-term human-robot interaction, conversational agents, pervasive lifestyle support

ACM Reference format:

Frank Kaptein, Bernd Kiefer, Antoine Cully, Oya Celiktutan, Bert Bierman, Rifca Rijgersberg-Peters, Joost Broekens, Willeke van Vught, Michael van Bekkum, Yiannis Demiris, and Mark A. Neerincx. 2021. A Cloud-based Robot System for Long-term Interaction: Principles, Implementation, Lessons Learned. *ACM Trans. Hum. Robot Interact.* 11, 1, Article 8 (October 2021), 27 pages.
<https://doi.org/10.1145/3481585>

1 INTRODUCTION

There is an increasing interest in long-term human-robot interaction. Social robots are commonly applied to education, health-care, public spaces, work environments, and home environments [58]. These systems often need to interact with several users and user groups at the same time and require interaction over prolonged periods of time to achieve their individual goals [58].

Current social robot systems have their own specific value, but remain simple and scripted in nature and miss the required comprehensive, prolonged, and personalised support. For example, in EU project ALIZ-E (www.aliz-e.org) a social robot was developed for children to support them in the self-management of diabetes [3, 18]. However, much of the implemented functionality remained scripted and required a “Wizard of Oz.” Furthermore, the children interacted with the robot “only” in a limited number of subsequent sessions [36, 63].

To establish long-duration pervasive human-robot interaction, our approach is to develop a personalised social-robot *with* its avatar that allows the user to always and anywhere engage in a diverse set of activities over a prolonged period of time (cf. Reference [87]). We propose four principles for the implementation of such a system: It must (1) have a connection to the cloud to delegate parts of the computational problems to external computers; (2) be modular to support parallel and incremental development of functionality; (3) have a common knowledge-base and vocabulary in the different system components and for the human-agent interaction; and (4) have hybrid artificial intelligence solutions (e.g., agent-based and machine learning) that all have their own contribution to the problem. We discuss these principles separately, and we discuss how they were incorporated into the system’s development.

This article presents the **Personal Assistant for a healthy Lifestyle (PAL)** system, an exemplary system of human-robot interaction that enables long-term support for health education and care. The robot autonomously interacts with children (aged 6–14) “in the wild” over a period of several months. The PAL system is a fully integrated and autonomous system that interacts with the children, their parents, and the health-care professionals for prolonged periods of time. It is composed of a social robot, its (mobile) avatar, and an expandable set of (mobile) health applications (diabetes diary, educational quizzes, sorting games, etc.). The system allows for adaptation to the patient’s condition and activities on the fly. It ran robustly during the duration of the experiment, i.e., more than half a year (single users interacted for two and a half to three months but started at different moments).

First, we discuss the related work in human-robot long-term interaction systems and discuss the context of our work in Section 2. Then, we discuss and motivate the four design principles of our

system in Section 3. The system architecture, as well as how the principles led to certain decisions is described in Section 4. We also describe the *process* of development and testing (i.e., decisions we made to streamline development in such a large scale project with several project partners) in Section 5. We discuss possible future extensions of the framework in Section 7. Finally, we discuss and conclude upon our efforts in Sections 8 and 9.

2 RELATED WORK AND CONTEXT

This section first describes the state-of-the-art for robot systems in long-term interaction. Then, we discuss our context, the PAL project. We argue that state-of-the-art robot systems all have their own specific value, but miss the required prolonged, comprehensive, and personalised support to successfully apply a robot system in health-care and education. Finally, we present important technical requirements for such a system.

2.1 Related Work

Leite et al. [58] surveyed existing social robot systems, identifying four domains for such systems: Health Care, Education, Work Environments and Public Spaces, and Home. It is possible for a system to fit in multiple domains simultaneously, for example, a robot might have a health support function as well as a health education function. Only in more recent work, social robot systems have been investigated in long-term studies (in EU projects like PAL, ALIZ-E, LIREC, L2TOR, UPA4SAR, WYSIWYD, and PATRICIA). This is because long-term interaction requires a degree of robustness, versatility, and autonomy. Something that technology only more recently is starting to provide.

Animal-like companion robots such as Pleo [28, 49], Paro [79, 80], and the AIBO robotic dog of SONY [1] have been used for some time in health-care and show potential with respect to treatment [8] and in maintaining adherence during prolonged interaction [45]. Such robots can provide comfort to their (elderly) users [79, 80], and develop social skills of the users (autistic children, 4–12 years old) [49]. However, such systems are limited in the richness and personalisation of the interaction, because they lack dialogue capabilities and direct educational functions.

Humanoids may have a harder time in maintaining long-term interaction with users. The embodiment of a robot influences the expectations we have of the robot's capabilities [54]. For example, we might expect a humanoid robot to communicate using natural language. Managing those expectations is challenging when attempting to maintain interaction with (especially child) users [60].

Several long-term studies have taken place where a social robot attempts to educate and/or support a user's health [18, 42, 47]. Coninx et al. [18] present a system for which a significant part of the system reaction was scripted and required Wizard-of-Oz intervention; the system was not really autonomous. Other examples are provided by the studies of Kanda et al. [42] and Kidd and Breazeal [47], who tested their systems in a two weeks, respectively, eight-week period. In general, the autonomous robots in the long-term studies are rather stand-alone applications and not integrated into a comprehensive behavioural support or healthcare system, contrary to the PAL system presented here. Cloud computing technology might enable the desired development of enduring comprehensive robot intelligence. For example, Hunziker et al. [38] present *Rapyuta* as a **Platform-as-a-Service (PaaS)** to offload demanding computation from the robot to the cloud. Several Human-Robot Interaction studies (e.g., References [15, 19, 83] worked on Internet and crowd-sourcing technology that facilitates large-scale distributed data collection, enabling internet-mediated (learning of) human-robot interaction. Most of these robot technology or software frameworks focus on the development of autonomous perception and action cycles. How to

include cognitive components that are key in our application domain (e.g., user models, beliefs, intentions, emotions, explanations and behavior change) is still rather unclear.

The final goal for our system was to deploy the *autonomous* system unchanged for a significantly longer time. This poses new challenges concerning stability, keeping interest in the system by creating a versatile and flexible dialogue and interaction system, and so on. Many principles and functionalities mentioned in previously discussed related work have been integrated into the PAL system, such as interactive suggestions and feedback to user activities, social talk and empathic behaviour, and have been taken one step further using an integrated memory of interactions with the user. To maintain a prolonged interaction with a robot it becomes vital that the robot truly has added value compared to the other technology available to the users, i.e., the robot must be functionally-relevant [21], or provide unique experiences to the user [43]. This seems quite challenging when the robot is applied to health-care and/or education. Still, a well-designed robotic system can help in the execution of educational and health-related tasks [18, 59].

2.2 Context: A Personal Assistant for a Healthy Lifestyle

The context of our system is the PAL project. The PAL project helps children (48 children, aged 6–14, recruited via hospitals in the Netherlands and in Italy) to cope with **Type 1 Diabetes Mellitus (T1DM)**. In a recent meta-analysis, Mobasseri et al. [65] showed that the incidence and prevalence of type 1 diabetes are substantial (respectively, 15 per 100,000 people and 9.5%), and that they are increasing in the world. This corresponds with the BMJ finding that, worldwide, the amount of children with T1DM is increasing by 3% every year, whereas the causes of this increase are still unknown. It is remarkable that there are substantial geographic variations in the incidence rates of T1DM (e.g., a child in Finland is far more likely to develop type 1 diabetes than a child in Asia) [74]. The growing burden of chronic illnesses on health and health-care has led to health policy responses increasingly referring to self-management. Becoming self-managed requires long-term motivation for change, which is especially difficult when the patient is a child. Zenlea et al. [93] showed that US children (ages 4–10) with diabetes type I more often had borderline/abnormal scores on the emotional symptoms scale, suggesting risk for psychological disorders, such as anxiety and depression. There are several challenges. The child needs to learn to deal with medical issues, like the proper use of an insulin pump or eating regularly, but also with psychological issues like feeling different from one's classmates. The caregivers and parents cannot always be there to support the child and their relationship will always be different than that of a peer [67].

In the PAL project, we developed a system to educate the child on, and support the child with his/her diabetes whilst continuing to be a peer of the child (a *pal*). The system consists of a social robot, its (mobile) avatar, an expandable set of (mobile) health applications (diabetes diary, educational quizzes, sorting games, etc.) for interaction with the children. Additionally, there is a monitoring dashboard that allows parents to oversee the child's progress and an authoring tool that allows caregivers to oversee and adjust how the system is configured. For example, a caregiver could increase the difficulty for a certain *learning goal* when the child shows good progress, or select a new goal to work on altogether. Some examples of such learning goals are: “*I know when to ask for help,*” or “*I know that I should take extra insulin when I eat extra carbohydrates.*”

There is interaction in hospitals and camps with scientists present, and continued long-term interaction at the children's homes. In the hospitals and camps, the PAL agent is a NAO robot. In the home interactions, there is an avatar impersonating the robot on a tablet screen. During all interactions, the PAL agent makes decisions and proposes activities to the child. It makes these proposals based on the configuration and progress of the child's personal learning goals.

In the PAL project, we have both a robot and an avatar as possible embodiments of the PAL agent. Robots have been shown to have a positive impact on motivation and learning [4, 62]. For example,

the NAO robot developed by Softbank (formerly Aldebaran) has already been used successfully in ALIZ-E, where children learn and are supported by the (robot-based) health-care system [3, 18]. However, a pragmatic problem with any sufficiently advanced humanoid of good quality is that it is an expensive device. This means that it is not feasible to provide a large group of users with their own personal robot. However to make developing content for an interactive robot attractive, a large user base is necessary. Perhaps this is a problem that will become less relevant in the future if humanoids become more affordable. Still, it may be a long time before owning a robot is as common as owning a car. Meanwhile virtual avatars are needed to support the development of human-robot interaction. In our context, the children can have an avatar of the robot on the tablet where the mobile health applications are installed. In the hospitals and camps, the children can interact with the physical robot.

3 PRINCIPLES FOR A SOCIAL ROBOT SYSTEM FOR LONG-TERM INTERACTION

In this section, we provide four main principles for developing a personalised long-term social robot system. Our vision is that such a system (1) should have a cloud-based implementation to distribute heavy computations and allow real-time adaptation of the system's functionality; (2) should be developed in a modular way to facilitate parallel development; (3) must contain a common knowledge-base and terminology for the different project partners, the different parts of the system, and the human-agent interaction; and (4) have hybrid artificial intelligence solutions (e.g., agent-based and machine learning) to contribute to the different (sub-)parts of the complexity. We discuss these four principles separately.

3.1 Principle 1: Cloud-based Robots

The first principle for a social robot system in long-term interaction is that the system should be *cloud-based*.

Cloud-based computing offers several advantages over stand-alone robot systems [44]. It allows the use of (1) *external libraries* for machine learning approaches to, for example, generate sentences for dialogue. It enables using (2) *external computers* to delegate complex computational tasks, e.g., a statistical analyses of previous behaviours and their outcomes. Enables the (3) *sharing of data and outcomes* of behaviours amongst different robots. So, when one robot learns that playing the quiz is a great way of teaching children to count carbohydrates, then it can share this knowledge with the other robots. Finally, Kehoe et al. [44] also mention that cloud robots enable (4) *Human Computation*, i.e., using crowd-sourcing for analysing, e.g., images and error recovery. However, we have not investigated this in our context, since we strove for a more autonomous system.

In addition to the advantages of cloud-based robots as stated in Reference [44], we would specifically state that it facilitates (5) *Personalisation and Adaptability* of the system. The different users of the robot system can adapt parameters online and thereby steer the robot's behaviour in desired directions. In this way, the human expert (the health-care professional within the context of PAL) can personalise the robot to the specific patient. Finally, (6) *integration with internet services* has sparked interest in the development of social robot systems. For example, recently (in 2018), AIBO was relaunched with improved artificial intelligence. It uses cloud-based techniques to apply deep learning for its reasoning and to develop a unique personality, depending on the behaviour of the owner. Another example is the ALEXA chatbot, which can be seen as a object shaped robot. ALEXA's main functionality is to easily provide internet services to the users.

There are risks associated with cloud-based computing that involve data security and privacy. In health settings patient data is often deemed particularly sensitive making security a highly important issue. One needs dedicated servers for cloud-based health applications with sophisticated data security and privacy management. In the PAL project, the servers doing the computation were

managed and secured by the hospitals or university. However, it remains a fact that the connectivity inherent in cloud applications can be of concern for security and privacy [44, 77].

3.2 Principle 2: Modular System

The second principle is that the system must be *modular*. Handling complexity in software development is facilitated by developing (nearly) modular components that are responsible for providing particular aspect of such a system [40]. Different techniques are built by software developers from different organisations at different locations (countries). For example, one German project partner provided a dialogue framework and another British partner an action selection framework. Setting up the architecture in a modular way allows us to connect these frameworks and facilitates parallel development of them.

A general problem is then solved by solving a set of interrelated sub-problems in a distributed manner [27]. The different modules need well-defined interfaces that determine how they interact with the rest of the system [46]. For the PAL system such a distribution could mean that choosing how to support a child means figuring out what educational goals there are for the child, suggesting an activity to work towards that goal, and figuring out what to say to the child, and how to explain the importance of the goals. These sub-tasks all work towards a bigger overall problem and all have their own computational complexities. The concerning functions need to be addressed as building blocks for intelligence, much like a “society of mind” [64] and in line with recent virtual agent architectures [32].

3.3 Principle 3: Common Knowledge-base and Terminology

The third principle is that the system must have a common terminology and knowledge-base that: (1) provides an unambiguous vocabulary in communication between stakeholders; (2) supports system implementation of knowledge-based reasoning functionality; and (3) serves as a basis for interoperability in human-agent interaction.

A common way of defining a knowledge-base is by means of an *ontology*. An ontology clarifies the structure of knowledge [13]. It contains explicit, formal specifications of terms in the domain and of the relations among them [34]: it is used to represent real-world objects and concepts and to specify properties of and relations between those objects.

3.4 Principle 4: Hybrid Artificial Intelligence

Finally, the fourth principle is that the system must be comprised of several artificial intelligence techniques that all have their own contribution to the interaction. On the one hand, **machine learning (ML)** techniques excel at learning optimal policies when given large amounts of data. Within the context of PAL this may mean that ML can learn what activities the robot should propose to a child to teach the child something about a specific issue (like, measuring blood sugar levels). On the other hand, symbolic representations are good for implementing expert knowledge in a human-understandable manner. For example, when the definition of a good blood sugar level differs per hospital, then an agent system can easily change a single belief without having to change anything in the implementation and logic or having to re-train.

4 SYSTEM IMPLEMENTATION FOR A SOCIAL ROBOT IN HEALTH EDUCATION AND CARE

Here, we provide a technical description of the architecture of the PAL system. Figure 1 shows an overview of the architecture. This section will discuss the sub-parts of this picture individually. First, we discuss the ontology with the knowledge and content. Second, we discuss the database. Then, we discuss the user interfaces. Finally, we discuss the “brain” and its individual elements.

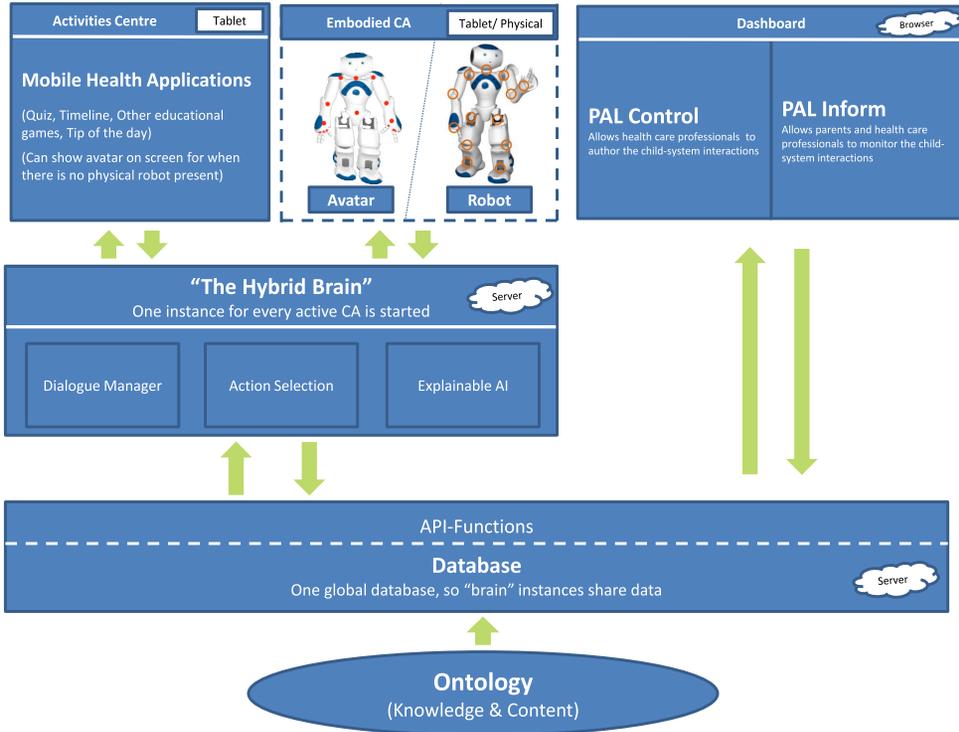


Fig. 1. The high-level architecture of the PAL system. The top row of the picture shows the user interfaces. The child interface consists of a tablet app that connects to a physical NAO robot or a NAO avatar (which is shown on the screen of the tablet application). On the app several mobile health applications are connected: educational games, a timeline where the child can keep track of his/ her blood sugar values, activities, and food regime, and so on. The health-care professional and parents can both see the child's progress on the learning objectives in the web-page interface. Only the health-care professional (in PAL control) can also adjust the robot's behaviour, i.e., steer the support to best fit the current state of the child's treatment. The "brain" is responsible for making the actor's behaviour intelligent and lifelike. For every child that logs in the system, a "brain" instance is started on the cloud. This allows the complex computations for the actor (robot or avatar) to happen on a more powerful, external computer. All elements of the system connect to the common database, and the data therein is structured along the PAL ontology, which is developed in cooperation with the health-care professionals.

4.1 The Ontology

This section specifically describes our work on developing a common knowledge-base and terminology (i.e., principle 3). The ontology represents concepts related to diabetes, actors and tasks involved in self-management, and emotions involved in human-agent interaction. It defines the definitions and relations between these concepts.

Developing such an ontology requires close collaboration with experts in the field, in this case the health-care professionals. This section gives an overview of some of the main decisions made and how they underpin the system's workings.

4.1.1 Ontology Frames. The entire ontology in the PAL project is constructed by integrating separate ontologies, linking them by means of a top-level ontology. These separate models function as high-level building blocks for smaller, more specific areas of interest (frames) [86].

- An ontology of human/machine roles and actors and locations involved in self-management.
- a generic ontology of tasks for actors (human and/or artificial), associated with goals and roles.
- An ontology of learning objectives, defining learning goals and tasks (activities) specific to the diabetes self-management domain, and child specific status of these learning objectives.
- Abstract ontologies that define notions of events and processes and various properties of time.
- A dialogue management ontology that contains dialogue acts and some semantic frames, based on the DIT++ taxonomy of dialogue acts (ISO standard 24617-2) and FrameNet, respectively.
- An episodic memory ontology as a system responsible for capturing specific events, or episodes, for the PAL system to interact with a human user in a meaningful manner over prolonged periods of time.
- An ontology for storing and reasoning over the affective process and state of a child, that allows the PAL system to estimate the emotions experienced by the child.

We have reused existing ontologies to cover the various frames wherever possible. Although the frames of interest mentioned above are typically generic in nature, pre-existing models for these frames may differ (slightly) in scope and/or intention and may thus be a partial fit to the intended scope of the frame in the context of PAL. Whereas, e.g., self-management activities of diabetes are a relevant topic, the entire professional medical diagnosis and treatment model of diabetes is out of scope. We have adapted some of the existing models by either extending them with additional concepts or by taking a profile (part) from the model whenever there are details/concepts in the model that are irrelevant to the scope of PAL. An example of reuse is displayed in the adoption of the well-known ontology for task world models [90] in the frame for tasks/goals and learning objectives. These objectives steer the behaviour of the robot and the treatment of the child.

PAL Objectives Model (POM). One important part of the ontology is the frame of **learning objectives** that covers most relevant aspects that the children (aged 6–14) with T1DM encounter in their daily lives and must learn to become self-manageable. These objectives steer the behaviour of the robot and the treatment of the child. Prior to (and during) the usage of the PAL system, the health-care professional, the parents, and the child together choose a subset of relevant learning objectives, based on the child’s individual needs, interests and knowledge. The learning objectives consist of achievements, goals and tasks. Achievements define a set of goals that are required to enable the user to carry out a real world challenge represented by the achievement. An achievement does not define new knowledge or skills but groups together goals with a similar level and related to a specific challenge to make goal setting easier. Goals, however, specify the end state of knowledge, skills or attitudes that a child should have to accomplish an achievement. Learning goals are hierarchically structured by difficulty level (i.e., novice to master) and level of complexity based on the Taxonomy of Educational Objectives of Bloom et al. [6]. An example of an achievement within our context is “*I can go to a sleepover,*” which contains goals, such as, “*I know when to ask for help,*” “*I know what to take with me to a sleepover,*” and “*I know that I should take extra insulin when I eat extra carbohydrates.*” To attain these goals, children have to do activities (and thereby complete tasks) within the myPAL application. The robot has several tasks/activities in its database that educate the child on these subjects. For example, the child might do a quiz together with the social robot, play a memory game, or the robot may simply explain something about the subject (a *tip of the day*).

Learning objectives can be attuned to a child’s developmental stage and the child’s personal and environmental context. Learning objectives are labelled with a knowledge level, difficulty level



Fig. 2. Child playing the Break and Sort game with the NAO robot using the integrated PAL system.

and prerequisite knowledge. Additionally they can be linked to a device (pen, pump, sensor) or a hospital. In this way, the ontology facilitates intelligent personalisation of interaction and learning process, which enhances motivation and learning gain [72].

4.2 The Database

The central data hub of the PAL system is based on an extended **Resource Description Framework (RDF)** hybrid storage component and reasoner (**Hans-Ulrich Krieger's forward chainer (HFC)**) [53]. Its special features allow putting the terminological knowledge (the ontology of diabetes knowledge, the definition of user and agent model, and static knowledge for language and dialogue processing) as well as the dynamic knowledge that is produced and consumed in the running instances of the myPAL app into one data repository, thereby fostering principle 3 the need for a common knowledge-base. The database plays a pivotal role for principle 1 *cloud computing*, because it helps the different computers to use the same data, have customised reasoning rules together with a streaming reasoning approach, and allows it to infer new data in real-time.

The database allows connecting different sub-ontologies (frames) by using equivalence statements. This fosters principle 2 (modularity) in the knowledge building process and facilitates re-usability of the work. Another advantage of structured databases, like RDF-based or graph databases, is the flexibility when it comes to adding or changing data structures. In general, this is much easier to achieve than for **relational databases (RDBs)**, especially because the knowledge representation, i.e., the specification of the data, and the (dynamic) data itself are in the same format. This allows quite simple checking consistency by custom reasoning rules.

The PAL system is based on a very particular implementation for RDF storage and reasoning that allows using n -tuples instead of the usual triples. This makes it possible to directly attach time and confidence information to every data chunk in a more efficient way than with currently available RDF storage solutions [50–52]. As a consequence, the database can contain a flow of events and data that is susceptible to temporal and probabilistic reasoning.

4.3 PAL Control and Inform

To be able to author the behaviour of the system, personalise the system towards the child and monitor how the child uses the system, we have developed a web-application with two dashboard modules and associated interfaces accessible depending on user role.

The monitoring dashboard, *palInform*, is the software module and associated interface that enables care takers to get an insight into how their children use the system, their progress, and what nutritional and medical values they fill in. It provides a timeline of the most important events,

based on system activity and the data the child enters, in an aggregated manner. The monitor displays glycemic, insulin, nutritional, activity and emotion-related data for a child, as entered by the child via the timeline in the myPAL app. Further, it displays goal attainment data in relation to time. The monitor is available for both **health-care professionals (HCP)** and the parent(s). To ensure data is shared with parents in a way that respects the child's privacy, the child and parent can set agreements on what information will or will not be available. Agreement options are used to balance between the conflicting values of privacy and medical safety. Requirements were developed together with the hospitals in the PAL project.

The authoring dashboard, *palControl*, is a tool to enable health-care professionals to set learning objectives (i.e., achievements, goals and tasks) for children during or in between meetings. It further enables the HCP to enter child data including personal data and preferences such as sports and hobbies and whether the child uses a pen or a pump for insulin intake. Two issues lead to its current design: (1) how to formalise the learning objectives based on the medical protocols and informal expertise of HCPs and (2) how to design the interface (and mechanisms behind it) to facilitate easy personal goal selection and progress monitoring during the intake meeting and further guidance of the education process ([72] and covered in the ontology Sections 4.1 and 4.1.1).

Both modules allow real-time changes to specified learning objectives (content and display) and personalised learning goal setting through the ontology, with the reasoning system. The modules thus most strongly link to principle 1 using cloud techniques and to principle 3 using a common knowledge-base, i.e., the modules facilitate real-time adaptation of the system's configuration and thereby its reasoning. It allows a HCP to adapt the system's functioning to the different phases of the treatment plan. The health-care professional can review the progress and tweak the system's behaviour accordingly. In line with the learning objectives discussed in Section 4.1, this can simply mean that the health-care professional updates the set of learning objectives the child and robot should currently work on.

4.4 Activity Centre

The activity centre is an application comprised of several activities that the **Conversational Agent (CA)** can perform with the user (shown on the top-left part of Figure 1). In our context this is a tablet application (myPAL) connected to a physical (NAO) robot or an avatar. myPAL contains health-related activities that support the child's treatment and that educate the child on diabetes.

MyPAL contains several games, i.e., educational quizzes, sorting games, and memory games. The activities in the application have been set up in a modular way (principle 2), allowing to add new games by implementing the required interfaces. In addition to the educational games, the PAL actor can provide a "Tip of the day," where it proactively provides the child with some information concerning diabetes. myPAL also provides a list of videos about several diabetes-related topics and a list of real-world tasks that the child must perform his-/herself. Finally, myPAL contains a timeline where the child can keep track of his/her blood sugar values, glycemic corrections, activities (sport and other), and food regime. All these activities are related to the child's current set of learning objectives. myPAL shows the child his/her progress on the learning objectives and provides "task suggestions," i.e., proposes particular educational activities to work on the learning objectives in a targeted manner. Figure 3 shows some screenshots of the different components of myPAL.

4.5 Communication Between Modules

The activity centre, conversational agent, and the separate modules in the "hybrid brain" all connect to a global communication platform, the "nexus" (see Figure 4). These modules can all subscribe to *types of messages* and send those message types. In this way, different modules can

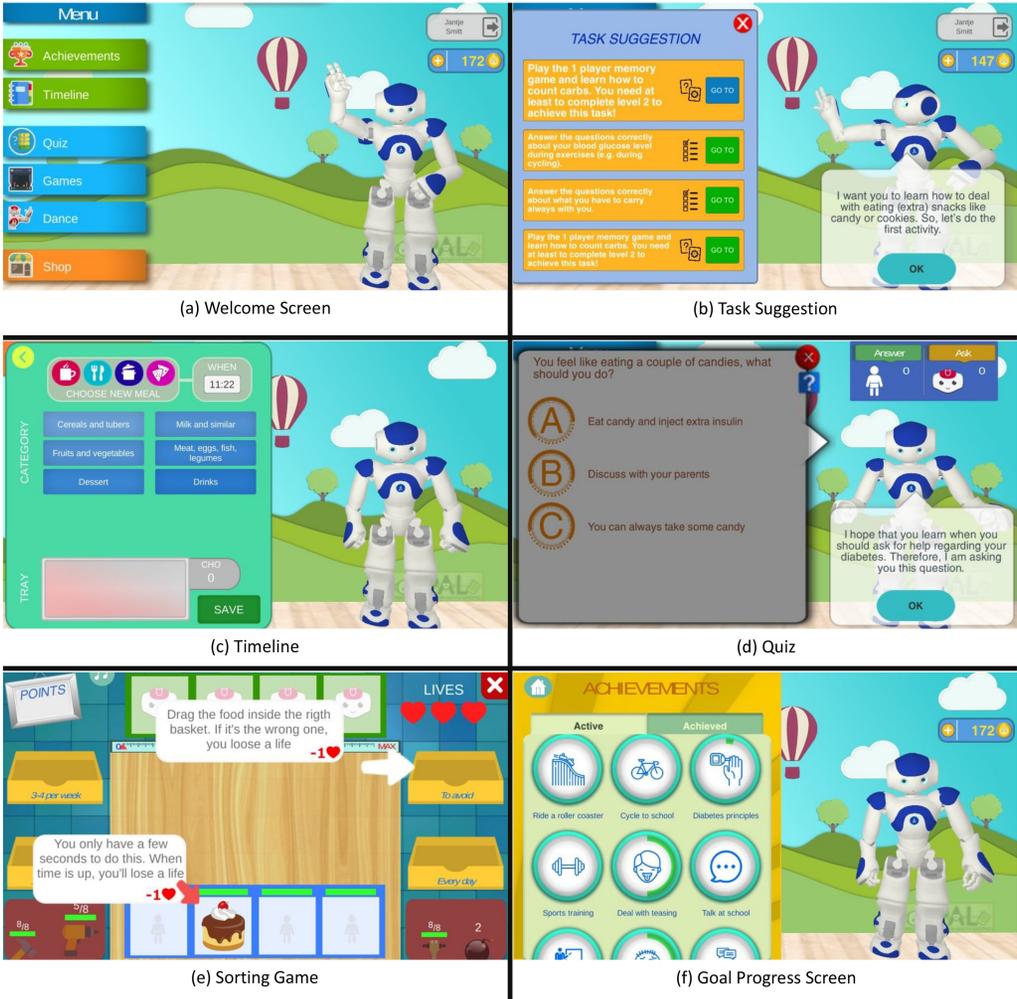


Fig. 3. Six screenshots of the myPAL application used by the children.

individually decide what information is relevant for them and how they should respond to new information.

For example, the child may click the quiz, which is then sent around on the nexus. The action selection module must respond to this message by choosing different quiz topics, the dialogue manager selects an appropriate dialogue act to initiate the game. The behaviour manager, however, might not need to know this at all. It will generate a movement only after the dialogue manager sends a high-level “speech-and-movement” message.

This greatly fosters modularity (principle 2) of the system. A new module can implement its own workings independently of the rest of the system. It can subscribe to messages that carry information relevant to the module. Of course, integration of the module still requires coordination amongst developers. The other modules should subscribe to new messages sent by the new module and should implement protocols on how to use this data for the human-agent interaction.

Modules can be independently maintained and improved as long as the interface contract is unchanged. The modularity also improves flexibility and reusability when requirements change.

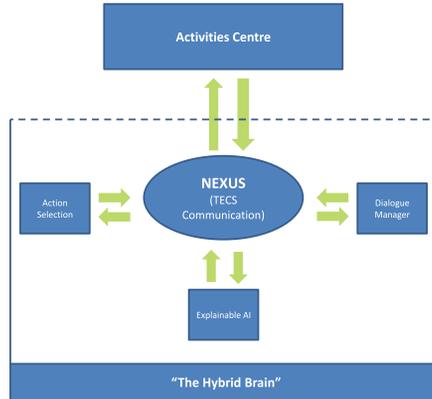


Fig. 4. Communication between the modules in the “brain” and with the child interface. All messages go through a common messaging board called the *nexus*. Modules individually decide to *subscribe* to *types of messages* and can themselves send messages of a particular type. Modules in this way individually implement what to do when a particular message is sent.

Even when the functionality of some module is extended, only modules that will profit from these enhancements have to be changed.

4.6 Multimodal Behaviour Manager

The robot-movement manager converts gesture specifications (“commands sent over the nexus”) into values about joints, angles and times, which are then sent to the robot or its avatar (i.e., it can communicate gesture and posture commands to both the robot and its avatar). The current framework meets two important challenges for cross-platform social human-agent interactions. First, the behaviours of the virtual and physical NAO have completely the same foundation and expression mechanisms, so that they can be perceived as really similar. Second, modulation of these gestures is possible to adapt affective (emotion, style) expressions conveyed in these behaviours.

The task of the robot-movement manager can be divided into two parts: (1) executing the multimodal utterances on the (virtual) NAO; and (2) making the (virtual) NAO appear lively.

Executing a multimodal utterance (1) is done when another module in the “brain” sends a specific message type over the nexus. Such a utterance consists of the name of the gestures to execute and the text to be spoken as well as an (optional) emotional (mood) modulation. From this content it constructs the necessary values to move the joints of the (virtual) NAO, from which results are then published to the messaging board. From the gesture name the robot-movement manager calculates the position of the joints of the (virtual) NAO and the time available for the joints to reach that position. When the execution of multimodal utterance is finished, the robot-movement manager sends this information the other modules.

Making the (virtual) NAO look lively (2) makes the interaction more engaging for the users. This is done by implementing continuous autonomous moves when no multimodal utterance is executed. Autonomous moves and movements resulting from a multimodal utterance need to be carefully combined to achieve smooth transitions. Furthermore, style adaptation during autonomous movement is challenging, because these motions are not pre-designed and thus require real-time modulation, which is less controllable and predictable.

The physical NAO. The implementation for the real NAO is a wrapper that converts the data received (from the robot-movement manager) into the commands that can be sent to the NAO’s

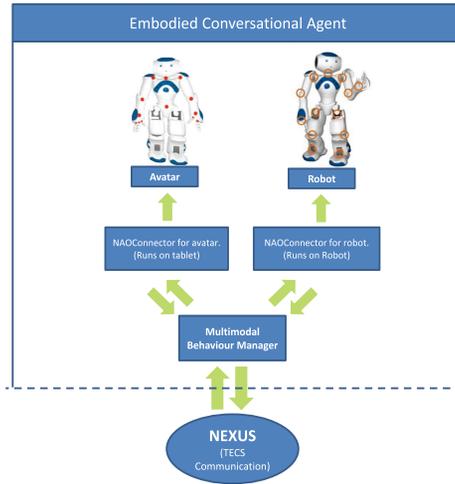


Fig. 5. Graphical representation of relations between the robot-movement manager and the embodiments.

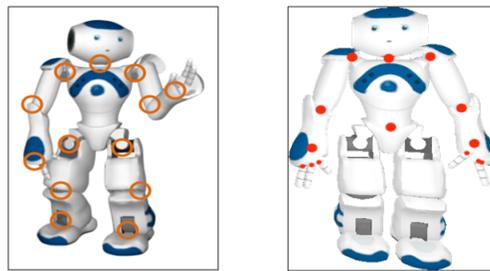


Fig. 6. Real NAO on the left with the joints indicated, avatar on the right with the junctions that were implemented indicated.

own execution system. The majority of the messages originate from the autonomous move system. The main issue to solve is handling all messages in parallel without blocking the processes.

The virtual NAO. The implementation for the virtual NAO is embedded in the Unity environment in which the tablet application (the activity centre) is developed and the task is to execute the moving of the avatar according to the values received. Since there was no implementation available it had to be developed in the context of the project.

A 3D Model was developed composed of virtual objects connected through a series of junctions, which take the place of real NAO’s gears, as displayed in Figure 6. Next, an algorithm was developed to translate the rotations of the real NAO’s gears into rotations of the NAO avatar’s junctions around the three axes (x, y, z), so that we obtain the same movements for all commands from the robot-movement manager.

Making the avatar’s leg movement realistic and correct with regard to the message sent was the most difficult part of the implementation. We developed a hybrid solution that allows calculating only a portion of the parameters required to apply a simulated gravity to some parts of the 3D model.

The implementation discussed here allows us to have all complex computations concerning what behaviours the robot *should* do on external computers (principle 1, using cloud techniques)

while the embodiment specific computations (transferring joints and times to actual movement) are done on the device itself. In the PAL project, the NAO robot was the target platform. However, the system could easily be applied to other types of robots (again linking to the principle of modularity 2). When a new robot platform would be introduced then high-level gestures like “wave arm” are still sent in the same way over the nexus. The behaviour manager and robot connectors, however, would require an extension to support the new embodiment.

Stylised Behaviours. Humans (often unconsciously) use social signals to inform others about their affective stance or attitude; based on observations, we evaluate someone as, among other things, warm or cold, competent or incompetent, friendly or hostile, and dominant or submissive (e.g., References [26, 29]). For artificial agents (both virtual and robotic), to engage in meaningful interactions with humans, the importance of social intelligence is widely acknowledged [30, 88]. On top of that, communication style is important in educational settings: teachers should use appropriate styles when interacting with students and this is not different for robots that teach [73]. Although previous work is available on the expression of affect by robots and agents [7, 11, 16, 25] and rapport building between agents and humans [33], there was no clear way of modulating style of movement for robots in an expressive and implicit way (with notable exceptions being some work on virtual agents [69, 76]). By implicit, we mean that the base behaviours and scenario of the robot are the same, but the style of the robot differs. In several works, we have shown that it is possible to manipulate style in a subtle manner. We were able to manipulate children’s perception of warmth and competence of a robot [73], the perception of warmth [75], and the perception of dominance [70]. In general, we found that different gestures as well as parameter-based modulation enabled us to express style in a recognisable manner.

Based on the theories from educational sciences, we defined three interaction styles for the PAL Actor: *friendly*, *direct*, and *neutral*. These styles are defined by the factors warmth, competence, and dominance, for which we designed and evaluated non-verbal behavioural patterns [70, 71, 73]. The factors define the styles as follows: *friendly*, high warmth, low competence, and low dominance; *direct*, low warmth, high competence, and high dominance; *neutral*, low warmth, low competence, and low dominance. Definitions of each style, and the mapping of a specific style to each activity, have been stored in the common knowledge-base.

We defined a minimal set of NAO joint adjustments to express each style. Some parameters are directly mapped to a specific joint (e.g., head tilt vertical with `HeadPitch`). Other parameters require adjustment of multiple joints (e.g., gesture openness requires adjustment of `ElbowYaw` and `ElbowRoll` relative to `ShoulderRoll`). These adjustment were applied to the original “stand” posture as designed by SoftBank (Alderbaran) to create a start and end pose for each style (see Figure 7). The relative joint adjustments were calculated for each key frame of each gesture in each style. Additionally, manual adjustments have been applied to distinctive motions for specific gestures in each style to avoid exceeding joint limits and/or creating jerky, unnatural motions.

During child-PAL Actor interactions the PAL system selects the appropriate interaction style based on the child model and the ongoing activity. Whenever the Behaviour Manager receives a nexus behaviour message (see Section 4.6) it uses a “mood” value to determine the selected style, and selects behaviours accordingly.

4.7 The “Hybrid Brain”

We call the collection of modules directly responsible for the decision making and managing the content of the behaviour of the PAL actor (i.e., the NAO robot or avatar) the “hybrid brain.” The “brain” runs on the cloud (principle 1), which allows the use of more powerful computers for computing the context dependent optimal behavioural strategies. It uses the communication

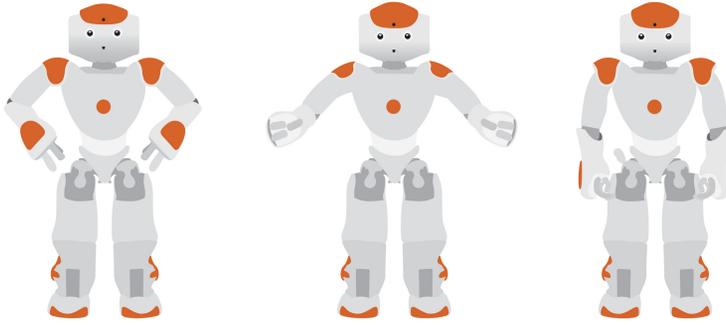


Fig. 7. Example of a base-pose for the styles implemented in PALY3, respectively, Direct, Friendly, and Neutral.

mechanism described in Section 4.5 to support easily connecting additional (sub-)modules (principle 2), and it is *hybrid*, since it consists of several artificial intelligence techniques (principle 4, hybrid). The action selection module uses mainly machine learning optimisation of protocols, while the explainable AI module uses BDI structures and ontology querying for its implementation, and the dialogue management uses a mixture of rule-based processing, BDI ideas, and statistical selection for the optimal strategy. In this section, we describe the different modules that support the human-agent interaction.

4.7.1 Action Selection and User Models. The PAL system uses an action selection module in conjunction with user models to personalise the behaviour of the system. Personalising the application to the particularities of each user is essential for two reasons: (1) it increases the engagement of the user to the PAL system, and (2) it allows the user to reach more effectively its personal goal(s) by adapting to his/her preferences.

The action selection is based on a hierarchical action selection architecture, HAMMER [23, 24], which uses multiple models to generate and evaluate multiple action possibilities. It takes place at two different levels in the PAL system. The first level is when multiple options are available during a dialogue. For instance, when the avatar can suggest to start one of the three games of the application, the action selection module will select the one that the user models predict as most beneficial for the user. An action is deemed beneficial if it increases the knowledge level of the user or if it increases its (predicted) happiness (typically, winning in a game increases happiness). The second level of decision making is within the quiz game to select the topic of the questions that are asked. Here again, a user model is used to predict the knowledge level of the users on the different topics covered by the PAL system. This information is to select of topics that are not too difficult nor too easy, but just with the right level of difficulty. Such a topic selection approach allows the user to remain in his “zone of proximal development,” which is known to provide optimal educational path [22, 89].

One of the main challenges in the action selection and user modelling is to produce accurate predictions from few data. Typically, a large amount of interactions is required to make an accurate estimation of the difficulty level of the child on one particular topic. This comes from the fact that each question returns only a limited amount of information: it only informs the system if the child managed to respond correctly to the question or not. This binary information is not enough to infer the actual knowledge level on a topic and the system needs to accumulate several dozens of responses to make an accurate assessment that is not biased by non-knowledge-related factors (e.g., random guess, mistakes caused by distraction or ambiguous formulation of questions). For instance, 10 data points only provide a rough approximation of the user level, while 100 data points

provide a more accurate estimation. This difficulty is amplified by the number of topics on which we would like an estimate of the knowledge level. For instance, if 20 questions are used to make an (potentially inaccurate) estimate on one topic, then more than 580 questions are required to form a global estimate of the user's knowledge level on each of the individual topics. The level estimations are made per topic to allow a child to excel in one topic, while progressing more slowly, or not at all, in others. This large amount of questions most of the time represents a limitation for intelligent tutoring systems, as they may be unable to provide a personalised educational path as long as the estimation of the knowledge level of the user is not completed. Designing a model that can account for sparse data and, therefore, provide accurate estimate with only a little amount of data is of crucial importance. In the PAL system this is partly overcome by the manual estimation of the child's entry level by the HCP via goal setting as described in Section 4.3.

Technical Description. In the PAL system, we have introduced a novel user model that leverages data from previous users of the application to bootstrap the predictions made by the user model. This user model is also able to track, in real-time, the evolution of the children's difficulty levels. Tracking this is important to continuously provide adequate level of difficulty for the users.

The realisation of this model is centred around three main features: (1) the model relies on Gaussian Processes to track online the evolution of the student's knowledge level over time, (2) it uses collaborative filtering to rapidly provide long-term predictions by leveraging the information from previous users, and (3) it automatically generates abstract representations of knowledge components via automatic relevance determination of covariance matrices. The model has been evaluated on three data sets, including data from real users and the results demonstrate that the model converges to accurate predictions in average 4 times faster than the compared methods. A detailed evaluation of the model's technical characteristics can be found in Reference [20].

4.7.2 Dialogue Management and Behaviour Modelling. The PAL actor needs extensive verbal and non-verbal communication skills to support the child in his/her learning process and to become a real companion. Dialogue is present in almost all activities of the app, guiding the child, or giving feedback to current and past performance in games, or to her/his treatment, in which case the data is provided by entries in the timeline.

The **dialogue manager (DM)** is responsible for multimodal generation (language and gestures) and, currently to a lesser part, multimodal interpretation. In Section 4.6, we discussed how a multimodal utterance should be processed causing the PAL actor (robot or avatar) to actually output the utterance. During idle phases, the robot-movement manager makes the actor to appear lively by constantly executing "autonomous moves." The DM, however, chooses the particular (deliberate) movements, like gesturing to something on the screen or waving to the child.

To be an interesting partner for conversation over a longer period of time, the communication strategy must be adaptive to the user. The DM takes a long-term perspective (using a user model) and takes short-term aspects into account, such as the current user mood or recent important events, as well as the parameters from the stylised behaviour model described earlier to modulate the interactions. This already points to the importance of a *world model* and a *memory* for the agent, which enables it to reason about past events and interactions, and subsequently uses its knowledge in the conversation. This, together with a high variability in dialogue strategies and a rich repertoire of verbal and non-verbal expressions to choose from, helps to make the artificial agent more appealing.

Dealing with children requires high reliability concerning the content and the way things are presented. This makes it difficult to use pure machine-learning-based methods, because the results will never be fully predictable, not to mention the problem of collecting enough data for dialogue strategies in the first place. Together with the need for very flexible dialogue that was mentioned

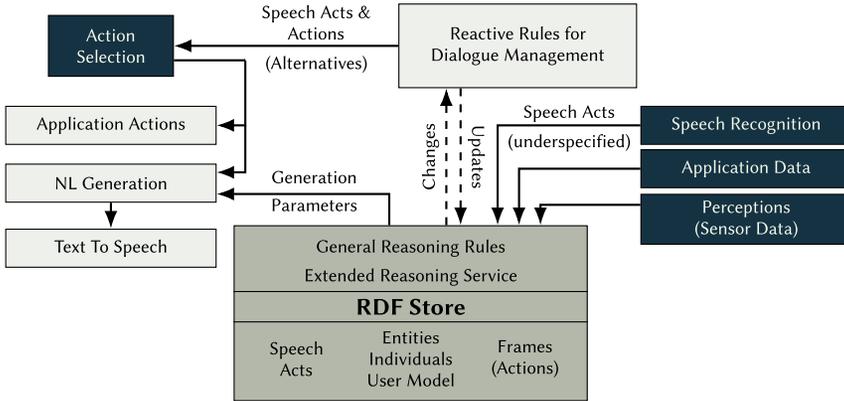


Fig. 8. A schematic VOnDA agent.

earlier, we decided to go for a hybrid rule-based approach with statistical selection, provided by the Action Selection unit (Section 4.7.1). Our special knowledge base (Section 4.2) enables us to use world knowledge, long-term memory and temporal reasoning in the dialogue management. The hybrid approach with long-term memory allows us to achieve the important goals of *reliability*, *flexibility* and *personalisation and adaptation over time* for the interaction management.

A central achievement of the PAL project is the development of the hybrid dialogue framework VOnDA [48], which facilitates creating reactive dialogue management engines following these principles. PAL’s DM follows the **Information State/Update (IS)** tradition [84] and is implemented mostly as VOnDA rules, with supporting Java code to hook it up to the “nexus,” or when specialised functionality is needed, e.g., for complex data base queries and computations.

On close inspection, the IS approach is very similar to the **Belief/Desire/Intention (BDI)** framework popular in agent modelling frameworks. Since VOnDA has all ingredients to be used for that kind of behaviour modelling, and it is anyway hardly possible to strictly separate the interactive and non-interactive behaviour modelling, we decided to make this component responsible for the agent’s actions. The reactive rule-based approach provides a lot of flexibility and generalisation at the same time, and has proven to result in a very efficient implementation that can compete in all respects with, e.g., behaviour tree implementations (for a survey, see Iovino et al. [39]). In fact, the rule system of VOnDA could easily be used to also implement behaviour trees. Treating all behaviours uniformly also allows us to solve synchronisation problems between different behaviour states more easily.

Figure 8 shows a schematic view of a VOnDA agent.

Past information states are kept inside the database, which gives dialogue strategies access to the full dialogue history. Furthermore, the RDF store also provides a flexible specification layer for domain knowledge, including knowledge about natural language concepts, such as dialogue acts and semantic frames, but also for domain-specific data structures that are used in the rule base. Since RDF/OWL is a well-established standard, there is plenty of tool support to create the ontologies that serve as basis for the dialogue management. With VOnDA, we have created a framework that tackles the following design goals:

- Flexible and uniform specification of dialogue semantics, knowledge and data structures
- Scalable, efficient, and easily accessible storage of interaction history and other data, resulting in a large information state
- Readable and compact rule specifications, facilitating access to the underlying RDF database, with the full power of a programming language

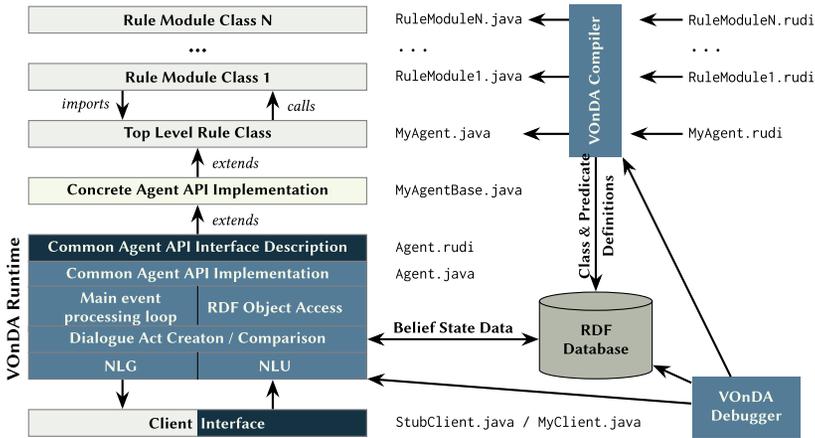


Fig. 9. A schematic VOnDA agent.



Fig. 10. VOnDA code and ontology.

- Transparent access to underlying programming code for simple integration with the host system

Technical Description. VOnDA consists of a compiler that turns rule descriptions into programming code, using data structure specifications from an ontology, and a run-time library that is used in the agent implementation and executes the rule code produced by the compiler. The compiler uses the class and property definitions in the ontology for type inference and type checking. The reactive rules take the form of if/then/else statements, and database objects and properties are mapped to OOP objects and fields. Figure 10 shows a small example with a tiny part of the ontology and some code using objects from this class hierarchy.

Apart from the main dialogues, there are interaction modules that are tightly connected to the dialogue manager and cover particular parts of the dialogue functionality, namely, *targeted feedback* to timeline entries, an *episodic memory* reasoning about past events, and a module for *off-activity talk*, which uses self-disclosure to engage the child in a conversation whose goal is not to increase his or her knowledge, but to increase the bond to the virtual agent. This again shows the modularity (principle 2) of our implementation. The modules subscribe to and send messages over the nexus. They can be activated or deactivated without impairing the rest of the system’s functionality. In a scientific project this has the additional benefit of allowing additional experiments and pilots to test the effects of specific parts of the system.

Targeted Feedback. The targeted feedback module is active during the timeline activity of the child interface. It first determines how close the child is following the routine of entering data into the timeline, and subsequently praises or encourages the child, depending on the frequency of the

entries, and the data itself. If alarmingly high or low values are reported, then the child is advised to talk to her/his parents to prevent a critical situation.

Episodic Memory. In contrast to the *targeted feedback*, the episodic memory module aggregates data from the past to detect events like the recent completion of a task or an achievement, exceptionally disciplined behaviour, and the like. These events are then subject to remarks and questions during the welcome phase directly after log-in. In this way, the agent encourages positive behaviour and shows that it takes interest in the child's daily activities.

Off-Activity Talk/Self-Disclosure. The goal of the off-activity talk (or social talk) module is to improve the virtual agent's social connection to the user, and make it more likeable. As a consequence, it should increase the user's inclination to follow the agent's advice and guidance. The current module is in large parts based on the study in Reference [9] enhanced with additional introductory dialogue moves and more elaborate prompts. Based on an estimate of "intimacy" between the user and the agent, which is based on general usage parameters, like the frequency of proper reactions to prompts by the agent, the agent reveals personal secrets of its habits or former encounters with other robots or persons.

4.7.3 Explainable Artificial Intelligence. **EXplainable Artificial Intelligence (XAI)** is the capability of a system to explain/ justify its behaviour to its users. The **General Data Protection Regulation (GDPR)** law states that users have the right to explanations [10]. It has been a significant topic during the development of the pal system.

XAI is of particular importance in applications where the system makes consequential decisions, like in health-care [35, 81], and is of particular importance in long-term interaction, since lack of trust in a behaviour change systems causes the users to not rely on the given advice [56]. Lack of trust can cause users to misuse or even abandon the system altogether [58, 66]. XAI has been shown to increase a user's trust in and understanding of the system's behaviour [17, 55, 61, 81, 92].

Within the context of the pal system, XAI is the capability of the actor to provide utterances that serve to make the system's behaviour intelligible. We implemented XAI for three different activities, quiz questions, tip of the day, and task proposals. These activities have already been discussed in the child interface Section 4.4. The XAI module monitors the system state by monitoring the messaging board. The module can provide an explanation during one of these activities by monitoring what goal the action selection is currently pursuing. It generates the explanation partly by using annotations and partly by generating the sentences. A goal has a description annotated to it. For example, "ask for help regarding your diabetes." Then there is a set of preceding sentences and following sentences that can be put in front and behind this description. The XAI module works together with the dialogue manager to produce the complete explanations and integrate them into the dialogue. For example, "I hope that you learn when you should ask for help concerning your diabetes. Therefore, I am asking you this (quiz) question." See Figure 11 for two examples of such explanations given during activities. Explanations could be given on the initiative of the pal agent or of the child's own accord. In the later case, the child could request an explanation by pressing a question mark visible on the screen (see also Figure 11(b) showing the question mark during the quiz).

5 DEVELOPMENT AND TEST PROCEDURES

Creating an integrated system requires far more solid software engineering approaches than commonly applied in scientific projects, i.e., it is required to program the modules "defensively" so that they can continue (to their best of capabilities) when other modules break. Furthermore, it requires rigorous development cycles with plenty of time to test all the new features in systematic

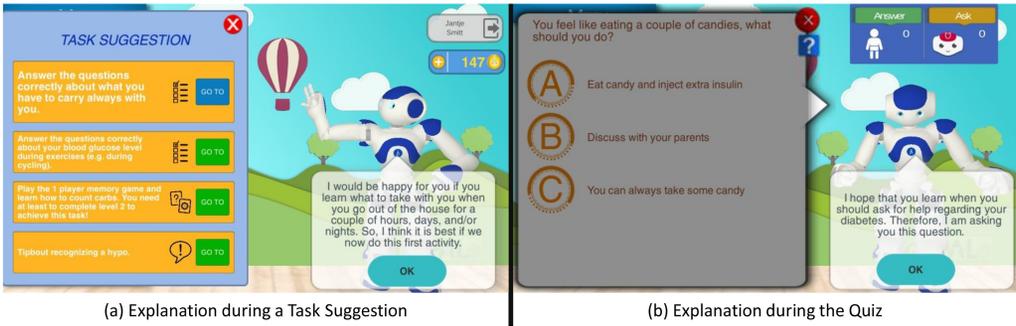


Fig. 11. Two screenshots of explanations given in the PAL system. Screenshot (a) shows an explanation for a task suggestion, screenshot (b) shows the quiz.

and thorough ways using, e.g., unit testing [14], integration testing [41], system testing [2], and performance testing [91]. Research projects need to be aware of this and have the necessary resources and knowledge available. For example, have commercial companies involved in the project.

We used **git** for code sharing and version management. Where the master branch with the stable version was only updated when features went through the proper tests. This included a stress test where 10 to >20 user accounts (controlled by the developers) logged into the system simultaneously and actively interacted with the conversational agent and did the activities. Next, non-developer researches used and tested the system. If they accepted the changes, then a new short-term experiment or pilot could be done. These served also as a final test where our target (child) users interacted with the system under supervision of researchers. This development procedure, over a course of three years, resulted in the system presented in this article and used by children for a period of two and a half to three months. The socio-cognitive engineering activities that focused on continuous stakeholder involvement, domain analyses, integration of human factors theories and claims analyses to validate the design rationale (in formative and summative evaluations) are described in a separate paper [68].

6 ANALYSES OF PERFORMANCE

In total there were 48 (25 Dutch and 23 Italian) children with **Type 1 Diabetes Mellitus (T1DM)** aged 6–14 that used the final system. The children were recruited via two hospitals in the Netherlands and one in Italy. There were no consequences to dropping out intermediately. The system was up and running more than 95% of the time. Forty-seven children had an average of 19 log-ins (STD = 12.9, minimum = 1, maximum = 55), with an average session duration of 12.8 min (STD = 16.4 min). During a session, children were involved in 14 different activities on average (STD = 24.6). One child was excluded from analyses due to a glitch in the data caused by a system error. The Randomized Controlled Test that compared the knowledge, performance and health conditions of these children with children who had “casual care” will appear in a separate paper [5]. Here, they report that children who used the system, showed a stronger increase in diabetes knowledge (i.e., competence) and experienced a higher level of autonomy in diabetes self-management [5].

7 FUTURE EXTENSIONS

Several extensions are possible in this system. It would be beyond the scope of this article to provide an exhaustive list. However, we briefly discuss a few modules that were developed during the project but did not yet reach the status of full integration in the system.

GPS Tracking. Behavioural adaption to individual differences (context awareness) is important for achieving good outcomes in health education and care through social robots. One aspect is modelling social context (e.g., home vs. hospital) through collecting and processing **global positioning system (GPS)** data. We saved pre-defined fixed locations (e.g., hospital locations) in the database and implemented a module that tracks a user's current location. This module additionally learned of and added other locations a user frequented at to the database. The pending future work is to integrate this with dialogue management and action selection so that conversations and activities are shaped to this context.

Eye-gaze Tracking. A second form of context awareness was implemented through eye-gaze tracking. We attempted to infer aspects of a user's mental state using eye gaze, which has been shown to be useful in training contexts [31]. Interaction logs and verbal protocols are generally not adequate for genuine cognitive and social profiling [82]. However, nonverbal cues such as eye gaze have been frequently studied for inferring user internal states in this context, as eye movements directly reflect what is at the centre of an individual's visual attention, and are linked to cognitive processes in the mind [85].

To address this, we implemented a state-of-the-art Convolutional Neural Network-based method for gaze tracking, which takes users' face images captured via the front facing camera as an input and predicts users' gaze fixations on the tablet screen in real-time (approximately 10 fps) [12]. We further developed a classification scheme to predict users' mental states, i.e., knowledgeability, from the estimated gaze fixations while they were playing an educational game with the robot/avatar [12]. We wanted to benefit from mental state predictions to better personalise quiz questions and make the avatar exhibit user-aware behaviours. For example, offering a hint to the user when estimating the user is having difficulties answering a quiz question. Unfortunately, eye-gaze tracking was not fully evaluated in the PAL system, because it required real-time on-device processing (to preserve the privacy of the user) and the mobile devices used did not have sufficient computational power available for this.

Automatic Speech Recognition. We considered **Automatic Speech Recognition (ASR)** to enhance interaction with our users. ASR could be a way to add more convenient conversational capabilities to the system, or add a dictation mode to the timeline, and possibly improve real-time emotion estimation for an even more empathic companion [37]. At that time, however, solutions based on openly available resources like software tools and pre-trained acoustic models turned out not to achieve the necessary recognition quality for such a project. However, cloud-based solutions posed a severe data and personality protection issue, since speech data are classified as biometric data and therefore on the highest protection level of the European Union Data Protection Guidelines. Therefore, we used ASR only for prototypical demonstration, as an outlook towards future extensions. Still, with the current advent of large coverage embedded solutions, and cloud ASR that explicitly guarantees adherence to the EU guidelines, integrating ASR would be a beneficial future extension for these types of systems.

8 LESSONS LEARNED AND DISCUSSION

This article presented a social robot system architecture that facilitates long-term human robot interaction. The four primarily important aspects of our architecture were: using cloud-based techniques (principle 1); having a modular architecture (principle 2); having a common knowledge-base to support development and support the human-agent interaction (principle 3); and using hybrid artificial intelligence (principle 4). This section discusses the global lessons learned concerning the development of such a system and concerning the four principles.

Building an integrated system in a research project poses unique challenges, which are usually avoided by implementing small, isolated prototypes that exhibit the behaviour that is under investigation. In an integrated system, the functionality of the modules must be synchronised to achieve the intended overall performance. This needs more consultation and arrangements between the implementing groups, joint definition of interfaces, down to the functional level, and thorough development cycles and testing procedures [2]. For these requirements, resources have to be allocated that are usually avoided in projects conducting basic research. As a consequence, developing and integrating work by different research groups can sometimes cause delays. Since several changes are usually developed simultaneously, a delay can impact multiple different experiments. Which means that the partners need to have a working relationship where everyone is willing to jump in and help one another when there is delay and that everyone needs to have some flexibility in their planning. Considering this, PAL is among the few research projects that created an integrated system that was used successfully by real, untrained users over a longer time and also has the potential to be commercially exploited in an extended and consolidated form.

PAL's cloud-based system architecture (principle 1) facilitates long-term interaction. First, it enables access to external libraries (Big Data, for example, used in the dialogue of this system); second, it allows for more complex computations by using external computational power (cloud-computing). In this way, the robot can provide a more stable and believable interaction with more diverse behaviours. Third, it facilitates version management, which is particularly important for research where functionality is incrementally added. Fourth, it helps the health-care professional to seamlessly personalise and adapt the system's behaviour based on the patient's individual development.

Furthermore, modularity (principle 2) reduces complexity, both for the development and application phase. However, we found that there are limits to the amount of modularity one should strive for. For example, it seems attractive to separate the agent's non-communicative behaviour (the application logic, so to say) and the (possibly multi-modal) communication layer. This, however, turns out to be challenging, because both parts heavily depend on each other to create a believable and consistent persona for the agent. For example, an ongoing dialogue can not simply be stopped at any point to perform some task, but has to be properly brought to an end for the robot not to be perceived as impolite or untrustworthy. However, the urgency of the task might require a more or less elaborate form of shutting down an ongoing conversation. We found that there is a delicate balance between modularity and coordination.

Another important aspect is that the system allows us to plug in a virtual avatar of the robot. Modularity is maintained, since other modules do not need to know whether the avatar or the physical robot is connected. This implementation could be extended by implementing controllers for other embodiments like a (physical and/or virtual) Pepper robot. Having an avatar of the robot is important, because many robots (like the NAO robot) are used for educational and/or health-care-related support, but are currently too expensive for commercial or home use. The **conversational agent (CA)** is meant to "bind" the user to the application. Having a robot present during meetings in the hospital and an avatar when using the application at home allows us to make use of the motivational gains of a CA without having to buy a robot for every child with diabetes.

For a common knowledge-base (principle 3), we chose to develop an ontology in the context domain of diabetes type 1. By working closely together with health-care professionals, we believe to have created a reusable knowledge-base that can be profitable for further research on combining health-care and artificial intelligence. For example, the formalised T1DM self-management learning goals and achievements are available as an ontology as well as in an online co-creation tool¹ to

¹<https://confluence.ewi.tudelft.nl/display/PO1/PAL+Objectives>.

allow for further specification and usage. We believe that reasoning over this expert knowledge is best done by applying agent-based or expert system implementations. Using symbolic AI, the system can propose several possible learning paths to the children to increase their self-management of their disease. Machine learning can then optimise the possible learning paths to the specific children. We show in our architecture how such hybrid techniques (principle 4) can work together successfully.

This article focused on the *technological* challenges of developing a social-robot system for long-term deployment by following four design principles, and presents how they were applied in a real-world implementation, namely, the PAL system. Complementary to this, Neerincx et al. [68] have presented the *human-centred method* (i.e., the systematic, iterative and incremental, design and test process of socio-cognitive engineering), showing how user needs and domain analyses and theories and methods on the human factors did drive PAL's support functionality and content (e.g., via different creative co-design sessions with children with T1DM and their caregivers, and the application of the *Self-determination Theory* that distinguishes three basic human needs to be addressed in behaviour change support systems (i.e., autonomy, competence, and relatedness [57, 78]). Complementary to the technological and human-centred aspects of the PAL-system development, Blanson Henkemans et al. [5] present the Randomized Controlled Trial that was conducted with the version of the PAL-system that is presented in this article, showing the effects on children their diabetes management. They report that children (age 6–14 years) with T1DM who used the PAL-system, showed a stronger increase in diabetes knowledge (i.e., competence) and experienced a higher level of autonomy in diabetes self-management [5]. This last effect was present, especially for the younger children (age up to 10 years). Overall, the children who used PAL indicated a better Quality of Life at the end of the study [5].

9 CONCLUSION

The PAL system architecture is a cloud-based, modular, long-term human-robot interaction framework that uses hybrid artificial intelligence and a common knowledge-base to shape the interaction. This system has run with over 40 users for two periods of two and a half to three months. The system remained stable and continued to show (more and more) behaviours and support in health education and care. This work can serve as a blueprint for future development of systems that need to facilitate long-term human-avatar and human-robot interaction studies.

ACKNOWLEDGMENTS

The four-year PAL project involved the research partners TNO (NL), DFKI (GER), FCSR (IT), Imperial (UK), and Delft University of Technology (NL), the hospitals Gelderse Vallei and Meander Medical Center (NL), the Diabetics Associations of Netherlands and Italy, and SME's Mixel (IT) and Produxi (NL).

REFERENCES

- [1] Marian R. Banks, Lisa M. Willoughby, and William A. Banks. 2008. Animal-assisted therapy and loneliness in nursing homes: Use of robotic versus living dogs. *J. Amer. Med. Direct. Assoc.* 9, 3 (2008), 173–177.
- [2] Boris Beizer. 1984. *Software System Testing and Quality Assurance*. Van Nostrand Reinhold Co.
- [3] Tony Belpaeme, Paul Baxter, Robin Read, Rachel Wood, Heriberto Cuayáhuil, Bernd Kiefer, Stefania Racioppa, Ivana Kruijff-Korbayová, Georgios Athanasopoulos, Valentin Enescu, et al. 2013. Multimodal child-robot interaction: Building social bonds. *J. Hum.-Robot Interact.* 1, 2 (2013), 33–53.
- [4] Tony Belpaeme, James Kennedy, Aditi Ramachandran, Brian Scassellati, and Fumihide Tanaka. 2018. Social robots for education: A review. *Sci. Robot.* 3, 21 (2018), eaat5954. <https://doi.org/10.1126/scirobotics.aat5954>
- [5] Olivier Blanson Henkemans, Elettra Oleari, Clara Pozzi, Daniele Baranzini, Sylvia Pal, Marije Bakker, Willeke van Vught, Mark A. Neerincx, Joost Broekens, Rifca Peters, Frank Kaptein, Yiannis Demiris, Bernd Kiefer, Diego

- Fumagalli, Bert Bierman, Riccardo Bonfanti, Andrea Rigamonti, Mirjam Schouten, and Gert Jan van der Burg [To appear]. Robotic playmate to develop diabetes self-management competency and performance during childhood: A randomized controlled trial.
- [6] Benjamin S. Bloom, Max D. Engelhart, Edward J. Furst, Walquer H. Hill, David R. Krathwohl, et al. 1956. *Taxonomy of Educational Objectives: the Classification of Educational Goals: Handbook I: Cognitive Domain*. Technical Report. D. McKay, New York.
 - [7] Cynthia Breazeal. 2003. Emotion and sociable humanoid robots. *Int. J. Hum.-comput. Stud.* 59, 1–2 (2003), 119–155.
 - [8] Joost Broekens, Marcel Heerink, and Henk Rosendal. 2009. Assistive social robots in elderly care: A review. *Gerontechnology* 8, 2 (2009), 94–103.
 - [9] Franziska Burger, Joost Broekens, and Mark A. Neerincx. 2017. Fostering relatedness between children and virtual agents through reciprocal self-disclosure. In *Proceedings of the 28th Benelux Conference on Artificial Intelligence (BNAIC'15)*, Tibor Bosse and Bert Bredeweg (Eds.). Springer International Publishing, Cham, 137–154.
 - [10] Peter Carey. 2018. *Data Protection: A Practical Guide to UK and EU Law*. Oxford University Press.
 - [11] Ginevra Castellano, Maurizio Mancini, Christopher Peters, and Peter W McOwan. 2012. Expressive copying behavior for social agents: A perceptual analysis. *Trans. Syst., Man, Cybernet.* 42, 3 (2012), 776–783.
 - [12] O. Celiktutan and Y. Demiris. 2018. Inferring human knowledgeability from eye gaze in mobile learning environments. In *Proceedings of the European Conference on Computer Vision Workshops (ECCVW'18)*.
 - [13] Balakrishnan Chandrasekaran, John R. Josephson, and V. Richard Benjamins. 1999. What are ontologies, and why do we need them? *IEEE Intell. Syst.* 1 (1999), 20–26.
 - [14] Yoonsik Cheon and Gary T. Leavens. 2002. A simple and practical approach to unit testing: The JML and JUnit way. In *Proceedings of the European Conference on Object-Oriented Programming*. Springer, 231–255.
 - [15] Sonia Chernova, Nick DePalma, and Cynthia Breazeal. 2011. Crowdsourcing real world human-robot dialog and teamwork through online multiplayer games. *AI Mag.* 32, 4 (2011), 100–111.
 - [16] Céline Clavel, Justine Plessier, Jean-Claude Martin, Laurent Ach, and Benoît Morel. 2009. Combining facial and postural expressions of emotions in a virtual character. In *Proceedings of the International Workshop on Intelligent Virtual Agents*. Springer, 287–300.
 - [17] Cristina Conati and Kurt VanLehn. 2001. Providing adaptive support to the understanding of instructional material. In *Proceedings of the 6th International Conference on Intelligent User Interfaces*. ACM, 41–47.
 - [18] Alexandre Coninx, Paul Baxter, Elettra Oleari, Sara Bellini, Bert Bierman, Olivier Blanson Henkemans, Lola Cañamero, Piero Cosi, Valentin Enescu, Raquel Ros Espinoza, et al. 2016. Towards long-term social child-robot interaction: Using multi-activity switching to engage young users. *J. Hum.-Robot Interact.* 5, 1 (2016), 32–67.
 - [19] Christopher Crick, Sarah Osentoski, Graylin Jay, and Odest Chadwicke Jenkins. 2011. Human and robot perception in large-scale learning from demonstration. In *Proceedings of the 6th International Conference on Human-Robot Interaction*. 339–346.
 - [20] Antoine Cully and Yiannis Demiris. 2019. Online knowledge level tracking with data-driven student models and collaborative filtering. *Trans. Knowl. Data Eng.* 32, 10 (2019), 2000–2013.
 - [21] Maartje De Graaf, Somaya Ben Allouch, and Jan Van Dijk. 2017. Why do they refuse to use my robot?: Reasons for non-use derived from a long-term home study. In *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction*. ACM, 224–233.
 - [22] Yiannis Demiris. 2009. Knowing when to assist: Developmental issues in lifelong assistive robotics. In *Proceedings of the International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC'09)*. IEEE, 3357–3360.
 - [23] Yiannis Demiris, Lisa Aziz-Zadeh, and James Bonaiuto. 2014. Information processing in the mirror neuron system in primates and machines. *Neuroinformatics* 12, 1 (2014), 63–91.
 - [24] Yiannis Demiris and Bassam Khadhour. 2006. Hierarchical attentive multiple models for execution and recognition. *Robot. Auton. Syst.* 54, 5 (2006), 361–369.
 - [25] Matthieu Destephe, Andreas Henning, Massimiliano Zecca, Kenji Hashimoto, and Atsuo Takanishi. 2013. Perception of emotion and emotional intensity in humanoid robots gait. In *Proceedings of the International Conference on Robotics and Biomimetics*. IEEE, 1276–1281.
 - [26] Steve L. Ellyson and John F. Dovidio. 1985. Power, dominance, and nonverbal behavior: Basic concepts and issues. In *Power, Dominance, and Nonverbal Behavior*. Springer, 1–27.
 - [27] Alexander G. Feoktistov, R. O. Kostromin, Ivan A. Sidorov, and Sergey A. Gorsky. 2018. Development of distributed subject-oriented applications for cloud computing through the integration of conceptual and modular programming. In *Proceedings of the 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO'18)*. IEEE, 0234–0239.
 - [28] Ylva Fernaeus, Maria Håkansson, Mattias Jacobsson, and Sara Ljungblad. 2010. How do you play with a robotic toy animal?: A long-term study of Pleo. In *Proceedings of the 9th International Conference on Interaction Design and Children*. ACM, 39–48.

- [29] Susan T. Fiske, Amy J. C. Cuddy, and Peter Glick. 2007. Universal dimensions of social cognition: Warmth and competence. *Trends Cogn. Sci.* 11, 2 (2007), 77–83.
- [30] Terrence Fong, Illah Nourbakhsh, and Kerstin Dautenhahn. 2003. A survey of socially interactive robots. *Robot. Auton. Syst.* 42, 3–4 (2003), 143–166.
- [31] Theodosia Georgiou and Yiannis Demiris. 2017. Adaptive user modelling in car racing games using behavioural and physiological data. *User Model. User-Adapt. Interact.* 27, 2 (2017), 267–311.
- [32] Jonathan Gratch, Arno Hartholt, Morteza Dehghani, and Stacy Marsella. 2013. Virtual humans: A new toolkit for cognitive science research. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, Vol. 35.
- [33] Jonathan Gratch, Ning Wang, Jillian Gerten, Edward Fast, and Robin Duffy. 2007. Creating rapport with virtual agents. In *Intelligent Virtual Agents*, Catherine Pelachaud, Jean-Claude Martin, Elisabeth André, Gérard Chollet, Kostas Karpouzis, and Danielle Pelé (Eds.). Springer, Berlin, 125–138.
- [34] Thomas Robert Gruber. 1993. A translation approach to portable ontology specifications. *Knowl. Acquis.* 5, 2 (1993), 199–220.
- [35] Steven R. Haynes, Mark A. Cohen, and Frank E. Ritter. 2009. Designs for explaining intelligent agents. *Int. J. Hum.-Comput. Studies* 67, 1 (2009), 90–110.
- [36] Olivier A. Blanson Henkemans, Bert P. B. Bierman, Joris Janssen, Rosemarijn Looije, Mark A. Neerinx, Marierose M. M. van Dooren, Jitske L. E. de Vries, Gert Jan van der Burg, and Sasja D. Huisman. 2017. Design and evaluation of a personal robot playing a self-management education game with children with diabetes type 1. *Int. J. Hum.-Comput. Studies* 106 (2017), 63–76.
- [37] Kun-Yi Huang, Chung-Hsien Wu, Qian-Bei Hong, Ming-Hsiang Su, and Yi-Hsuan Chen. 2019. Speech emotion recognition using deep neural network considering verbal and nonverbal speech sounds. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'19)*. IEEE, 5866–5870.
- [38] Dominique Hunziker, Mohanarajah Gajamohan, Markus Waibel, and Raffaello D'Andrea. 2013. Rapyuta: The roboearth cloud engine. In *Proceedings of the IEEE International Conference on Robotics and Automation*. IEEE, 438–444.
- [39] Matteo Iovino, Edvards Scukins, Jonathan Styruud, Petter Ögren, and Christian Smith. 2020. A survey of behavior trees in robotics and AI. Retrieved from <https://arXiv:2005.05842>.
- [40] Nicholas R. Jennings and Michael Wooldridge. 1998. Applications of intelligent agents. In *Agent Technology*. Springer, 3–28.
- [41] Paul C. Jorgensen and Carl Erickson. 1994. Object-oriented integration testing. *Commun. ACM* 37, 9 (1994), 30–38.
- [42] Takayuki Kanda, Takayuki Hirano, Daniel Eaton, and Hiroshi Ishiguro. 2004. Interactive robots as social partners and peer tutors for children: A field trial. *Hum.-Comput. Interact.* 19, 1–2 (2004), 61–84.
- [43] Frédéric Kaplan. 2005. Everyday robotics: Robots as everyday objects. In *Proceedings of the Joint Conference on Smart Objects and Ambient Intelligence: Innovative Context-aware Services: Usages and Technologies*. ACM, 59–64.
- [44] Ben Kehoe, Sachin Patil, Pieter Abbeel, and Ken Goldberg. 2015. A survey of research on cloud robotics and automation. *IEEE Trans. Autom. Sci. Eng.* 12, 2 (2015), 398–409.
- [45] Csaba Kertész and Markku Turunen. 2017. What can we learn from the long-term users of a social robot? In *Proceedings of the International Conference on Social Robotics*. Springer, 657–665.
- [46] Gregor Kiczales and Mira Mezini. 2005. Aspect-oriented programming and modular reasoning. In *Proceedings of the 27th International Conference on Software Engineering*. 49–58.
- [47] Cory D. Kidd and Cynthia Breazeal. 2008. Robots at home: Understanding long-term human-robot interaction. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'08)*. IEEE, 3230–3235.
- [48] Bernd Kiefer, Anna Welker, and Christophe Biwer. 2019. VONDA: A framework for ontology-based dialogue management. In *Proceedings of the International Workshop on Spoken Dialogue Systems Technology (IWSDS'19)*. Springer. Retrieved from <http://arxiv.org/abs/1910.00340>.
- [49] Elizabeth S. Kim, Lauren D. Berkovits, Emily P. Bernier, Dan Leyzberg, Frederick Shic, Rhea Paul, and Brian Scassellati. 2013. Social robots as embedded reinforcers of social behavior in children with autism. *J. Autism Dev. Disord.* 43, 5 (2013), 1038–1049.
- [50] Hans-Ulrich Krieger. 2016. Capturing graded knowledge and uncertainty in a modalized fragment of OWL. In *Proceedings of the 8th International Conference on Agents and Artificial Intelligence. International Conference on Agents and Artificial Intelligence (ICAART'16)*. INSTICC. Retrieved from https://www.dfki.de/web/forschung/publikationen/renameFileForDownload?filename=gradedmodals.pdf&file_id=uploads_2684.
- [51] Hans-Ulrich Krieger. 2016. *Integrating Graded Knowledge and Temporal Change in a Modal Fragment of OWL*. Springer-Verlag. Retrieved from http://www.dfki.de/web/forschung/iwi/publikationen/renameFileForDownload?filename=gradedmodals.pdf&file_id=uploads_2684.
- [52] Hans-Ulrich Krieger and Stefan Schulz. 2016. A modal representation of graded medical statements. In *Proceedings of the Conference on Formal Grammar*. LNCS, Vol. 9804. Springer-Verlag, Berlin, 1–17. Retrieved from https://www.dfki.de/web/forschung/publikationen/renameFileForDownload?filename=gradedmodals.pdf&file_id=uploads_2827.

- [53] Hans-Ulrich Krieger and Christian Willms. 2015. Extending OWL ontologies by cartesian types to represent n-ary relations in natural language. In *Proceedings of the Conference on Language and Ontologies*.
- [54] Minae Kwon, Malte F. Jung, and Ross A. Knepper. 2016. Human expectations of social robots. In *Proceedings of the 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI'16)*. IEEE, 463–464.
- [55] Dung N. Lam and K. Suzanne Barber. 2005. Comprehending agent software. In *Proceedings of the 4th International Joint Conference on Autonomous Agents and Multiagent Systems*. 586–593.
- [56] John D. Lee and Katrina A. See. 2004. Trust in automation: Designing for appropriate reliance. *Human Factors* 46, 1 (2004), 50–80.
- [57] Lisa Legault. 2017. Self-determination theory. *Encyclopedia of Personality and Individual Differences* (2017), 1–9.
- [58] Iolanda Leite, Carlos Martinho, and Ana Paiva. 2013. Social robots for long-term interaction: A survey. *Int. J. Soc. Robot.* 5, 2 (2013), 291–308.
- [59] Jamy Li, René Kizilcec, Jeremy Bailenson, and Wendy Ju. 2016. Social robots and virtual agents as lecturers for video instruction. *Comput. Hum. Behav.* 55 (2016), 1222–1230.
- [60] Mike Ligthart, Olivier Blanson Henkemans, Koen Hindriks, and Mark A. Neerincx. 2017. Expectation management in child-robot interaction. In *Proceedings of the 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN'17)*. IEEE, 916–921.
- [61] Brian Y. Lim, Anind K. Dey, and Daniel Avrahami. 2009. Why and why not explanations improve the intelligibility of context-aware intelligent systems. In *Proceedings of the 27th International Conference on Human Factors in Computing Systems*. 2119–2128.
- [62] Rosemarijn Looije, Mark A. Neerincx, and Vincent de Lange. 2008. Children’s responses and opinion on three bots that motivate, educate and play. Red de Agentes Físicos. https://scholar.google.com/scholar?hl=en&as_sdt=0%2C5&q=,+Children%E2%80%99s+responses+and+opinion+on+three+bots+that+motivate%2C+educate+and+play.&btnG=.
- [63] Rosemarijn Looije, Mark A. Neerincx, Johanna K. Peters, and Olivier A. Blanson Henkemans. 2016. Integrating robot support functions into varied activities at returning hospital visits. *Int. J. Soc. Robot.* 8, 4 (2016), 483–497.
- [64] Marvin Minsky. 1988. *Society of Mind*. Simon and Schuster.
- [65] Majid Mobasseri, Masoud Shirmohammadi, Tarlan Amiri, Nafiseh Vahed, Hossein Hosseini Fard, and Morteza Ghojzadeh. 2020. Prevalence and incidence of type 1 diabetes in the world: A systematic review and meta-analysis. *Health Promotion Perspect.* 10, 2 (2020), 98.
- [66] Bonnie M. Muir. 1994. Trust in automation: Part I. theoretical issues in the study of trust and human intervention in automated systems. *Ergonomics* 37, 11 (1994), 1905–1922.
- [67] Marta Murillo, Joan Bel, Jacobo Pérez, Raquel Corripio, Gemma Carreras, Xavier Herrero, Josep-Maria Mengibar, Dolors Rodríguez-Arjona, Ulrike Ravens-Sieberer, Hein Raat, et al. 2017. Health-related quality of life (HRQOL) and its associated factors in children with type 1 diabetes mellitus (T1DM). *BMC Pediatrics* 17, 1 (2017), 16.
- [68] Mark Neerincx, Willeke Vught, Olivier Blanson Henkemans, Elettra Oleari, Joost Broekens, Rifca Peters, Frank Kaptein, Yiannis Demiris, Bernd Kiefer, Diego Fumagalli, et al. 2019. Socio-cognitive engineering of a robotic partner for child’s diabetes self-management. *Front. Robot. AI* 6 (2019), 118. <https://doi.org/10.3389/frobot.2019.00118>
- [69] Truong-Huy D. Nguyen, Elin Carstensdottir, Nhi Ngo, Magy Seif El-Nasr, Matt Gray, Derek Isaacowitz, and David Desteno. 2015. Modeling warmth and competence in virtual characters. In *Proceedings of the International Conference on Intelligent Virtual Agents*. Springer, 167–180.
- [70] R. Peters, J. Broekens, K. Li, and M. Neerincx. 2019. Robot dominance expression through parameter-based behaviour modulation. In *Proceedings of the International Conference on Intelligent Virtual Agents*. ACM.
- [71] Rifca Peters, Joost Broekens, Kangqi Li, and Mark A. Neerincx. 2019. Robot dominance expression through parameter-based behaviour modulation. In *Proceedings of the 19th ACM International Conference on Intelligent Virtual Agents*. ACM, 224–226.
- [72] Rifca Peters, Joost Broekens, and Mark A. Neerincx. 2017. Guidelines for tree-based learning goal structuring. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces (IUI'17)*. ACM, 401–405. <https://doi.org/10.1145/3025171.3025188>
- [73] Rifca Peters, Joost Broekens, and Mark A. Neerincx. 2017. Robots educate in style: The effect of context and non-verbal behaviour on children’s perceptions of warmth and competence. In *Proceedings of the 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN'17)*. IEEE, 449–455.
- [74] BMJ Best Practice. 2020. *Type 1 Diabetes: Straight to the Point of Care (Last Updated: Jan 21, 2020)*. Technical Report. BMJ, UK. Retrieved from <https://bestpractice.bmj.com>.
- [75] Pooja Prajod, Mohammed Al Owayyed, Tim Rietveld, Jaap-Jan van der Steeg, and Joost Broekens. 2019. The effect of virtual agent warmth on human-agent negotiation. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS'19)*. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 71–76. Retrieved from <http://dl.acm.org/citation.cfm?id=3306127.3331676>.

- [76] Brian Ravenet, Magalie Ochs, and Catherine Pelachaud. 2013. From a user-created corpus of virtual agent's non-verbal behavior to a computational model of interpersonal attitudes. In *Proceedings of the International Conference on Intelligent Virtual Agents*, Vol. 8108 LNAI. Springer, 263–274. https://doi.org/10.1007/978-3-642-40415-3_23
- [77] Kui Ren, Cong Wang, and Qian Wang. 2012. Security challenges for the public cloud. *IEEE Internet Comput.* 16, 1 (2012), 69–73.
- [78] Richard M. Ryan and Edward L. Deci. 2017. *Self-determination Theory: Basic Psychological Needs in Motivation, Development, and Wellness*. Guilford Publications.
- [79] Selma Sabanovic, Casey C. Bennett, Wan-Ling Chang, and Lesa Huber. 2013. PARO robot affects diverse interaction modalities in group sensory therapy for older adults with dementia. In *Proceedings of the IEEE International Conference on Rehabilitation Robotics (ICORR'13)*. IEEE, 1–6.
- [80] Takanori Shibata and Kazuo Tanie. 2001. Physical and affective interaction between human and mental commit robot. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'01)*, Vol. 3. IEEE, 2572–2577.
- [81] Edward H. Shortliffe, Randall Davis, Stanton G. Axline, Bruce G. Buchanan, C. Cordell Green, and Stanley N. Cohen. 1975. Computer-based consultations in clinical therapeutics: Explanation and rule acquisition capabilities of the MYCIN system. *Comput. Biomed. Res.* 8, 4 (1975), 303–320.
- [82] Veikko Surakka, Marko Illi, and Poika Isokoski. 2003. Voluntary eye movements in human–computer interaction. In *The Mind's Eye*, J. Hyona, R. Radach, and H. Deubel (Eds.). North-Holland, Amsterdam, 473–491. <https://doi.org/10.1016/B978-044451020-4/50026-8>
- [83] Russell Toris, David Kent, and Sonia Chernova. 2014. The robot management system: A framework for conducting human-robot interaction studies through crowdsourcing. *J. Hum.-Robot Interact.* 3, 2 (2014), 25–49.
- [84] David R. Traum and Staffan Larsson. 2003. The information state approach to dialogue management. In *Current and New Directions in Discourse and Dialogue*. Springer, 325–353.
- [85] G. Underwood. 2005. *Cognitive Processes in Eye Guidance*. Oxford University Press.
- [86] Michael A. van Bekkum, Hans-Ulrich Krieger, Mark A. Neerinx, Frank Kaptein, Bernd Kiefer, Rifca Peters, and Stefania Racioppa. 2016. Ontology engineering for the design and implementation of personal pervasive lifestyle support., In SEMANTiCS (Posters, Demos, SuCESS). In *Proceedings of the CEUR Workshop*. 5–8. <https://doi.org/10.1587/elex.3.299>
- [87] J. G. van der Drift Esther, Robbert-Jan Beun, Rosemarijn Looije, Oliver A. Blanson Henkemans, and Mark A. Neerinx. 2014. A remote social robot to motivate and support diabetic children in keeping a diary. In *Proceedings of the 9th ACM/IEEE International Conference on Human-Robot Interaction (HRI'14)*. IEEE, 463–470.
- [88] A. Vinciarelli, M. Pantic, D. Heylen, C. Pelachaud, I. Poggi, F. D'Errico, and M. Schroeder. 2011. Bridging the gap between social animal and unsocial machine: A survey of social signal processing. *Affect. Comput.* 3, 1 (2011), 69–87.
- [89] Lev Semenovich Vygotsky. 1980. *Mind in Society: The Development of Higher Psychological Processes*. Harvard University Press.
- [90] M. Van Welie, G. C. Van Der Veer, and A. Elins. 1998. An ontology for task world models. In *Proceedings of the Conference on Design, Specification and Verification of Interactive Systems*. 57–70.
- [91] Elaine J. Weyuker and Filippos I. Vokolos. 2000. Experience with performance testing of software systems: Issues, an approach, and case study. *IEEE Trans. Softw. Eng.* 26, 12 (2000), 1147–1156.
- [92] L. Richard Ye and Paul E. Johnson. 1995. The impact of explanation facilities on user acceptance of expert systems advice. *Mis Quart.* (1995), 157–172.
- [93] Ian Spencer Zenlea, Lauren Mednick, Jennifer Rein, Maryanne Quinn, Joseph Wolfsdorf, and Erin T. Rhodes. 2014. Routine behavioral and mental health screening in young children with type 1 diabetes mellitus. *Pediatric Diabetes* 15, 5 (2014), 384–388.

Received October 2019; revised June 2021; accepted July 2021