Goal ontology for personalized learning and its implementation in child's health self-management support

Rijgersberg-Peters, R.; Vught, W. van; Broekens, D.J.; Neerincx, M.A.

Goal Ontology for Personalized Learning and Its Implementation in Child's Health Self-Management Support

Rifca Rijgersberg-Peters, Willeke van Vught, Joost Broekens, and Mark A. Neerincx

***Abstract*—Intelligent tutoring systems need a model of learning goals for the personalization of educational content, tailoring of the learning path, progress monitoring, and adaptive feedback. This article presents such a model and corresponding interaction designs for the coaches and learners (respectively, a monitor-and-control dashboard and mobile app with supportive communications trough a virtual agent), all deployed and tested in a system for child diabetes self-management training. We developed a domain-independent upper ontology to structure learning goals and related concepts (such as achievements and tasks) and a domain ontology that specifies the knowledge base (for, in our case, diabetes self-management training). With this approach, we relate knowledge elements (e.g., skill) to educational tasks and to learners' knowledge development (e.g., achievements). The ontology was implemented in a multimodal tutoring system consisting of mobile educative games, a health diary, an embodied conversational agent (ECA), and a web application for authoring and monitoring. We show that our model provides a coherent and concise foundation for: 1) the formalization of learning in the diabetes self-management domain, but also for other domains such as mathematics; 2) personal goal setting and thereby personalization of the educational process including ECA's guidance; and 3) creating awareness of progress on the personal educational path. We found that a motivational tutoring system requires a rich set of learning activities and accompanying materials of which a subset is offered to the learner based on personal relevance. The implemented model proved to accommodate the personal agent-guided learning paths of children with diabetes, under different treatments from hospitals in Italy and the Netherlands.**

***Index Terms*—Diabetes self-management, education, human–agent interaction, knowledge base, learning goal, objective, ontology.**

Manuscript received 5 January 2023; revised 6 July 2023; accepted 9 October 2023. Date of publication 23 October 2023; date of current version 11 January 2024. This work was supported in part by the EU Horizon 2020 PAL Project under Grant 643783. *(Corresponding author: Rifca Rijgersberg-Peters.)*

Rifca Rijgersberg-Peters is with the Department of Interactive Intelligence, Delft University of Technology, 2628 Delft, The Netherlands (e-mail: r.m.peters@tudelft.nl).

Willeke van Vught is with the TNO, Institute for Applied Research, 3769 Soesterberg, The Netherlands (e-mail: willeke.vanvught@tno.nl).

Joost Broekens is with the Institute of Advanced Computer Science, Leiden University, 2311 Leiden, The Netherlands (e-mail: joost.broekens@gmail.com).

Mark A. Neerincx is with the Department of Interactive Intelligence, Delft University of Technology, 2628 Delft, The Netherlands, and also with the TNO, Institute for Applied Research, 3769 Soesterberg, The Netherlands (e-mail: mark.neerincx@tno.nl).

This article has supplementary downloadable material available at https://doi.org/10.1109/TLT.2023.3326892, provided by the authors.

Digital Object Identifier 10.1109/TLT.2023.3326892

I. Introduction

New media technologies, such as intelligent tutoring systems (ITSs), offer opportunities to advance educational processes and guidance by direct and personalized instructions and feedback. These systems support student independence (of classroom and teacher) and improve learning gain. Specifically, extracurricular self-regulated learning may benefit from this, for example, in eHealth applications designed to support behavior change into a healthy lifestyle by increasing knowledge and control over ones well-being. Success rates depend on the user's (intrinsic) motivation for system usage and adherence to the educational program [1]. Embodied conversational agents (i.e., virtual or physical robots) may stimulate the motivation. Here, the challenge is to overcome the so-called *novelty effect* of educational robots: humans initially show high motivation with general interest in the new interactive technology, but this interest can wear off quickly after repeated interactions [2].

Motivation (for learning) was suggested to benefit from goal setting and feedback on goal attainment (e.g., [3]), as well as tailoring of the educational path to the learner's prior knowledge. It is motivating for the learners to be aware of the goals [4] and to be involved in goal setting [5]. Self-regulated learning systems can exploit this by focusing on goal-oriented learning. A personalized guidance approach in ITSs provides learners with the most appropriate learning content accounting for individual learner characteristics, such as current knowledge, interests, and motivation [6]. This supports application of scaffolding, an approach where the individual learner is constantly challenged to perform slightly above their achieved level of proficiency. This approach is based on Vygotsky [7] theory on the zone of proximal development (ZPD). The ZPD indicates what a learner can do with guidance and lies between what a learner can do without help and what he/she cannot do. Providing learners with experiences within their ZPD are expected to encourage and advance their individual learning [8], [9].

The goal of this article is to advance learner's self-education, supporting and augmenting educators' guidance by the construction of reusable learning objectives and improvement of automated selection thereof. To realize this, we needed: 1) the formalization of learning objectives and 2) the representation of this knowledge in a way software technology (agents) can reason about objectives. For the first challenge, we need to identify relevant items of knowledge and skills within a specified domain. Furthermore, we need insight into the learner's current knowledge and capabilities and a mechanism for personal goal selection and task selection. Hence, we define the following research questions.

Research question 1: How to model learning objectives.

Research question 2: How to deploy the model in a learning environment.

We use domain modeling to support personalized guidance by formalizing the domain knowledge elements (DKEs), the knowledge content, and the user's current knowledge level. Furthermore, we developed interfaces for various user groups to author and/or visualize this information, and we formalized reasoning about this information; we implemented algorithms using this information for personalized content selection and adaptive interaction. In the rest of this article, we will discuss these elements and user evaluation thereof. We conclude with recommendations and future work.

A. PAL Case Study

The PAL project[1] developed a multimodal application supporting children (7–14 years old) acquiring knowledge, skills, and attitudes required for diabetes self-management. The PAL system is composed of an embodied conversational agent (robot and its avatar), an extendable set of (mobile) health applications, and dashboards for caregivers, which all connect to a common knowledge base and reasoning mechanism.

Within the project, we constructed an integrated ontology, *PALO*, the PAL Ontology[2] serving as a common language to support flexible normative behavior, establish mutual understanding in (child–agent) interactions, and integrate and use existing knowledge from various domains. The PALO is assembled from several independently developed, but interrelated, ontologies expressed in Web Ontology Language (OWL) [10]. Neerincx et al. [11] provide an overview of the PAL development and evaluation. This article presents the *PALObjectives Model* (*POM*) ontology, a subontology of *PALO* that defines the knowledge, skills, and attitudes required for diabetes self-management as well as its implementation in the personalized learning support of the PAL system.

II. Background

We aim to construct clear consistent learning objectives that both require and facilitate shared common understanding of the meaning of these objectives. Therefore, we look into how learning goals are formulated in pedagogy and ontologies for education that formalize learning objectives. An ontology allows for making concepts explicit with specification of necessary attributes and relationships [12]. Finally, we present some necessary background on the domain of diabetes self-management learning.

[1]Personal Assistant for a healthy Lifestyle, http://www.pal4u.eu

[2]Publicly available upon request, https://confluence.ewi.tudelft.nl/display/PALsCE/PAL+ONTOLOGY

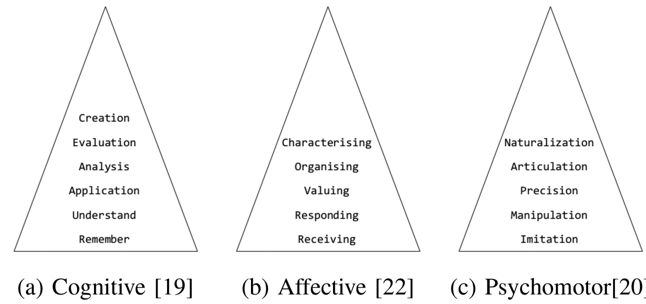


Fig. 1. Taxonomies for learning goal skill levels. (a) Cognitive [19]. (b) Affective [22]. (c) Psychomotor [20].

A. Effective Learning and Learning Goals

Clear and concise learning goals play an important role in student motivation; it directs activities and guides assessment and gives meaning. Knowing why things must be learned improves motivation and effectiveness of learning [13]. A learning goal must include a description of what the learner should be able to do, under what conditions, and how well it should be done. These are encoded in the properties description and proficiency level (i.e., novice, etc.) and skill level [14].

Current learning goals must fit with the learner's prior knowledge to facilitate effective learning. One way of structuring past knowledge is by Bloom's skill levels; see below. Furthermore, a distinction is made between declarative knowledge (facts, what a learner should know) and procedural knowledge (processes, what a learner should be able to do) [15]. The latter includes the following items: showing how it is done, doing by the learner, and internalization. Procedural knowledge is often preceded by declarative knowledge.

Bloom's taxonomy of learning goals [16] is the predominant model structuring educational learning objectives into levels of complexity and specificity. There are three submodels covering learning objectives in cognitive (knowledge), affective (emotional), and psychomotor (action) domains (see Fig. 1). Each domain is composed of categories with a sequential hierarchical link; learning of the lower levels enables the learning of skills in higher levels. This scaffolding of higher level skills from lower level skills is an application of constructivism [17]. There are different versions of the categories (see [18] for an overview). We choose to adhere for the cognitive domain to the revised taxonomy by Krathwohl [19] in favor of the original version by Bloom et al. [16]. In the psychomotor domain, we listed the categorization by Dave [20] in favor of the version by Simpson [21] because it presents more concrete actions and is thereby more applicable to differentiate levels of skill goals.

B. Ontologies

Advancements of semantic web for education allow for developments toward collaborative personalized and self-organized learning in intelligent systems. Where prior hyperlinking is limited to passive access to materials, the semantic web supports automated reasoning about it. Furthermore, an ontology facilitates shared common understanding of a domain [12].

1) Ontology Engineering: An ontology is a representation of a domain through a formal specification of concepts and their relations and limitations [23], [24]. In education, this means a network of semantically related learning objects within a specific instructional domain, for our applications, a hierarchy of concepts with a set of properties and relations defined in the OWL.

OWL is a logic-based knowledge representation that is understandable for humans and interpretable by computer programs, including:

individuals instances of classes and properties; for example, individual *Emma* is instance of class `Child`;

classes set of individuals with some common properties; for example, *Emma* and *Noah* are both children and thus belong to the class `Child`;

attributes characteristics that individuals can have; for example, *Emma* `om:hasAge` *8*, and *Emma* `hasSibling` *Noah*;

relations interrelations classes and individuals can have; for example, `Child` is-a `subClassOf Actor`. Subclasses allow for hierarchical arrangements from which the computer program can infer that *Emma* is an `Actor` (because she is a child and a child is an actor).

An ontology is defined by a set of resource description framework (RDF) triples, where each triple contains a subject, object, and predicate. The object of an property can be either an individual or a certain data type. All classes, properties, and individuals are uniquely identified with a uniform resource identifier (URI). Ontologies can be described in Protégé.[3]

There are different types of ontologies. In education, most common are the upper and domain ontologies; furthermore, ontologies exist for learner data, curriculum, and services [23]. An upper ontology presents the formal semantics, the representation of a knowledge base. A domain ontology represents the factual information, the knowledge base that represents facts about the world that a machine can reason about and use rules and other forms of logic to deduce new facts.

2) Ontologies for Education: In recent years, the popularity of ontology engineering for the development of e-learning systems increased. In this section, we discuss relevant literature on the use of ontologies for learning, focusing on ontologies that include learning goals.[4]

In education, domain-dependent and domain-independent ontologies have been developed focusing on, respectively: 1) the definition and (hierarchical) structure of knowledge concepts (e.g., for IT education and English language and mathematics [25], [26]) and 2) the learning goals.

Ontologies are proposed for various usages. For example, to explain core concepts to instructors and to generate lesson plans [25] or define essential knowledge for an online tutoring program by automated scraping of learning goals from instruction materials [26]. Furthermore, the formalization of existing learning goals may be used for course, curriculum, and syllabus modeling, as well as automated calculation of learning goal attainment [27], for course alignment of goals, tasks, and assessment [28], [29], to optimize discovery of learning materials [24], [30] or to form collaborative learning groups based on goals, triggers, materials, and scenarios [31]. We found limited usage of ontologies for personalized learning and/or goal setting. In one study, a hierarchy between learning goals was used to suggest materials that fit the learner's ability and requirements including personal learning goals [30], whereas David et al. [28] developed a "digital twin" guiding learners based on current learner profile and status. Ontologies are used in recommender systems for learning content based on user interest of needs. These ontologies typically represent knowledge about the learner and the learning content (objects) but do not specify learning objectives, thereby personalizing the content but not necessarily the learning path. For example, Assami et al. [32] facilitate online course selection, and Wongthongtham et al. [33] identify suitable online resources for concepts based on motivational interviewing (MI) profiles. In [34], assessment-based content adaptation does support personalizing of the learning path by recommending content based on performance. We expect a rise of use of ontologies for personalized learning in the future. Rashid and McGuinness [26] explicitly stated their intention to implement their ontology in a smart tutoring framework.

There appears not to be one single best practice for learning goal modeling. For example, Chung and Kim [27] define an identifier, description, cognitive level, attitude level, and skill level and form relations between activities and defined goals, whereas Ng [30] defines title, description, sequencing rules, related tasks, proficiency level, performance indicator, cognitive state, and relations with knowledge reference in another ontology. However, the common denominator seems to be a formalization of hierarchical relations between learning goals and inclusion of a human-understandable description of the goal and an indicator of the depth of knowledge. The latter is referencing often to Bloom skill levels (i.e., in [24], [25], [27], and [30], but may also embed 21st century skills (i.e., [29]), or SOLO (i.e., [28]). Furthermore, relations between goals and tasks occur (see, e.g., [27], [29], and [30]), in some cases referencing Bigg's triangle for effective learning (i.e., [28]).

C. Current Practices in Diabetes Education and PAL

Type 1 diabetes mellitus (T1DM) is a high-impact digestion disease, which requires daily self-management and is diagnosed in a growing number of children. To improve well-being and avoid complications, long-term behavior change is necessary [35]. Current training practices are informal, personalized, and aimed at optimizing child/patient autonomy and intrinsic motivation. Learning objectives are personal and change while ageing; therefore, self-management education is highly personalized and directed by challenges faced in daily life and aimed at gradual development of attitudes, knowledge, and skills needed for autonomous self-management [36]. Formalization of

[3][Online]. Available: https://protege.stanford.edu/

[4]Learning goals are often referenced as "learning objectives" and learning materials and learning activities as "learning objects." For clarity, we adhere to the terms goal, task, and material as used throughout this article. We use the term learning objective as superclass of the terms learning goal, learning task, and achievement.

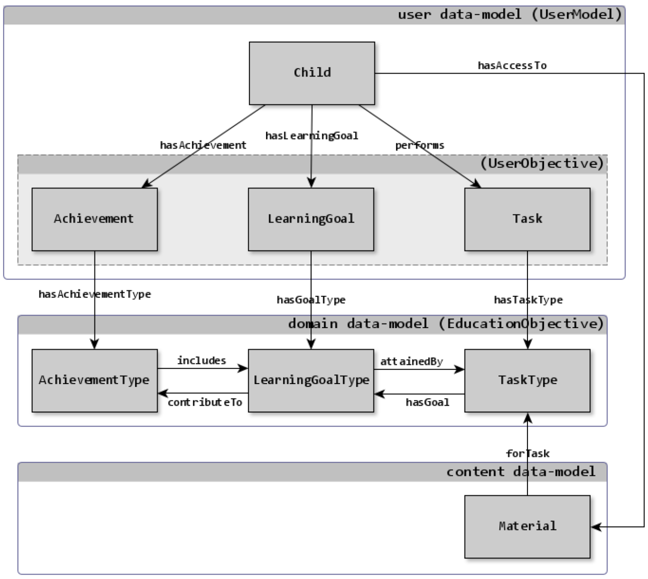


Fig. 2. Overview of the three submodels in the upper ontology, the objectives model, with their most important classes and relations.

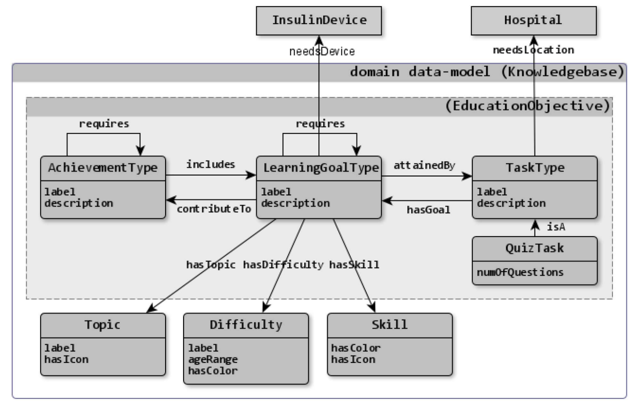


Fig. 3. Objectives-model-domain model classes, properties, and relations (extended version).

the learning process is limited to annual checklists such as the EADV[5] "weet- & doe-doelen" [knowledge- and skill goals].

There is a need for an ontology that underpins the personalized learning process of children via educative games and an embodied conversational agent. Current ontologies show the benefits of a hierarchical structure and the inclusion of core elements of Bloom's hierarchy, but do not provide a complete ontological model yet. The next section provides the development of such a model, elaborating on the research presented above.

III. PAL Objectives Model

We created an ontology formalizing learning objectives for self-regulated learning using Protégé (see Section II-B). The ontology is developed in an iterative process. The initial goal ontology was loosely based on the task ontology by Welie [37] and then further developed with experts from education, data science, and pediatric health care and taking into consideration the usability requirements from the PAL project. The results are two separate OWL files for the upper ontology and the domain ontology.[6] The upper ontology (*Objectives Model*) presents the formal semantics, a data-model representation for a knowledge base. The domain ontology (*POM*) is the knowledge base presenting factual information about the domain of diabetes self-management education.

The upper ontology, *Objectives Model* (*OM*), is domain independent and describes relevant concepts and relations between them. The *OM* contains three submodels: 1) domain model; 2) user model; and 3) content model (see Fig. 2). The three submodels store different types of data, as was also suggested for models for adaptive educational systems [6]. The domain model (this is *not* the same as the domain ontology) is a domain-independent structure of DKEs; it structures the global knowledge base consisting of three main concepts: `Achievement`, `Learning Goal`, and `Task`. These concepts all are subclasses of `Objective`. The user model is an overlay model structuring the storage of personal user data, child-specific (progress) data on these concepts, and it models the `Child` itself. The content model structures the learning `Material` related to tasks.

The domain ontology, *POM*, instantiates the upper ontology. Domain-specific instances are created accordingly to the upper ontology data model. In the rest of this section, we present the domain-independent upper ontology and illustrate this with examples from the domain ontology created for the PAL project in the domain of diabetes self-management training for children aged 7–14 and starting at diabetes onset. See Appendix A for an additional maths example.

A. Domain Model

The domain model structures the formal representation of knowledge about learning objectives in a specific domain, as formalization of a "concept map" in OWL, with a class structure, machine readable and with automated reasoning possibilities, and consists of three main concepts and three supporting classes, as presented below and in Fig. 3. Learning goal is the primary concept of the knowledge base structure. Achievements group goals in relevant challenges. Tasks prescribe activities to train new knowledge defined in goals. Supporting constructs are present that provide additional information about mainly the learning goals.

The three main classes (i.e., `LearningGoalType`, `AchievementType`, and `LearningTaskType`) each have two data properties `om:label` and `om:description`. The latter is a full sentence description of the objective. For example, "I remember how much carbs standard foods contain." The former is a short-hand description in a few words, intended for small GUI elements; for example, "Sleepover" and "Amount of CHO." See Listing 1 for further examples.

[5]The Dutch organization for diabetes nurses, http://www.eadv.nl
[6]Available upon request at the first author.

```
%Learning goal
pom:carbs2 om:label "Amount of CHO (standard)"^^xsd:
    string
pom:carbs2 om:description "I remember how much carbs
     standard foods contain."^^xsd:string
%Achievement
pom:sleepover om:label "Sleepover"^^xsd:string
pom:sleepover om:description "I can go to a
    sleepover."^^xsd:string
%Task
pom:memoryAH om:label "Memory CHO (hard)"^^xsd:
    string
pom:memoryAH om:description "Play the 1 player
    memory game and remember the amount of carbs per
    food. You need a 'hard' score to complete this
    task."^^xsd:string
```

Listing 1: Example for objective data properties `label` and `description`.

1) Learning Goal: Learning goal defines the desired end level of knowledge. Individuals from the class `LearningGoalType` form a structured set of DKEs that collectively span the total domain or a desired subpart. Each individual present precisely one knowledge element such as "Amount of CHO" from Listing 1. A goal may directly follow up on one or multiple other (easier/less advanced) goals and/or precede one or multiple advanced goals. For example, learning to count carbohydrates is preceded by knowing the amount of carbohydrates in various food items. Relations between goals are formalized in object properties `om:requires` and its inverse `om:requiredFor` as follows.

```
pom:carbs2 om:requiredFor pom:carbs3
pom:carbs3 om:requires pom:carbs2
pom:carbs3 om:requiredFor pom:countCarbs3
pom:carbs3 om:requiredFor pom:countCarbsK4
pom:countCarbs3 om:requires pom:carbs3
pom:countCarbsK4 om:requires pom:carbs3
```

Listing 2: Example for goal relational properties `requires` and `requiredFor`.

Goals also hold relationships with other objective types; a goal may be `om:attainedBy` one or more tasks and `om:contributeTo` one or more achievements. For example, "Amount of CHO" is trained playing a memory game (*AH* about the *A*mount of carbohydrates in food products on *H*ard difficulty) and two sorting games (*CEH* about the amount of *C*arbohydrates in food and *NEH* about *N*utrition in general in food products, both on *E*asy mode but requiring a *High* score) and belongs to six achievements among whom to go to a sleepover and to eat at a restaurant.

```
pom:carbs2 om:attainedBy pom:memoryAH
pom:carbs2 om:attainedBy pom:sortCEH
pom:carbs2 om:attainedBy pom:sortNEH
pom:carbs2 om:contributeTo pom:sleepover
pom:carbs2 om:contributeTo pom:restaurant
```

Listing 3: Example for goal relational properties `attainedBy` and `contributeTo`.

Finally, goals have object properties providing information about the subject and educational levels. A generic description of the subject(s) in given by `om:hasTopic`, such as "Nutrition" and "Carbs." The level is given by `om:hasDifficulty` and `om:hasSkill`, where skill indicates the desired end level of factual knowledge, motor skill, or attitude. For example, "Amount of CHO" has a desired end state of the child being able to appoint the amount of carbohydrate in everyday food products, thus remembering of factual knowledge. Whereas the goal to count carbohydrates over combined food products requires the child to apply knowledge in new situations. Difficulty indicates the proficiency level on a scale [1–6]. Remembering the amount of carbohydrates in various foods is not typically the first thing learned after diabetes onset and is not expected at early age (<8 years old); therefore, it has difficulty level 2 (advanced beginner).

```
pom:carbs2 om:hasTopic pom:carbs
pom:carbs2 om:hasTopic pom:nutrition
pom:carbs2 om:hasDifficulty om:difficulty2
pom:carbs2 om:hasSkill bloom:remember
pom:countCarbs3 om:hasSkill bloom:apply
```

Listing 4: Example for goal object properties.

In addition, a goal may hold a requirement for usage of a certain physical device. In the PAL case, an insulin device is either a pen or a pump. For example, a goal related to learning to bolus (a correction of the amount of insulin administered via an insulin pump) is only relevant for children who use a pump for their insulin intake.

```
pom:bolus1 om:needsDevice pom:pump
```

Listing 5: Example for goal object property `needsDevice`.

2) Achievement: Individuals of the class `AchievementType` form a set of domain knowledge components (KC). Each KC represents a fragment of knowledge within a specific domain (i.e., a relevant challenge). Achievements do *not* define new knowledge but form a set of related goals on a similar difficulty level. Achievements should be phrased as a main accomplishment that the child is eager to achieve to promote understanding and relevance of the achievements and optimize motivation for learning. Each achievement may, similar to goals, precede or succeed one or multiple other achievement(s). For example, the achievement "Sleepover" is preceded by an achievement related to basic diabetes knowledge indicating that attaining these goals is required before.

```
pom:sleepover om:requires pom:basicDiabetes
pom:basicdiabetes om:requiredFor pom:sleepover
```

Listing 6: Example for achievement relational properties.

Furthermore, goal relations are formalized by the property `om:includes` (this is the inverse of goal property `om:contributeTo`). For example, the achievement "Sleepover" at difficulty 2 groups five goals .

Achievements do not have other properties; topic(s) and difficulty level(s) are inherited from included goals. For example, "Sleepover" takes from "carbs2" the topics "Nutrition" and

```
pom:sleepover om:includes pom:carbs2
pom:sleepover om:includes pom:askHelpKnow2
pom:sleepover om:includes pom:FunStressKnow2
pom:sleepover om:includes pom:InsulinForCarbsKnow2
pom:sleepover om:includes pom:RegSnackKnow2
```

Listing 7: Example for achievement relational properties.

"Carbs" and "Insulin" from "insulinForCarbs2." An achievement (goal set) is created on each difficulty present in included goals. For example, "Sleepover" includes goals on difficulty levels 1–4, subsets are created on each level, and the child is required to complete "Sleepover1" (i.e., attain all included goals) before he/she is able to train the goals included in "Sleepover2" and so on.

3) Task: Individuals of the class `LearningTaskType` represent a structured set of activities to train goals based on the ITS application(s). Each task is related to a specific learning goal, formalized in the object property `om:hasGoal` (inverse of `om:attainedBy`). Goals can be trained by more than one task, but each task is designed to train one goal.

```
pom:memoryAH om:hasGoal pom:carbs2
pom:sortCEH om:hasGoal pom:carbs2
pom:sortNEH om:hasGoal pom:carbs2
```

Listing 8: Example for task relational property.

There are no requirements such as precessors. Tasks can be location specific (`om:needsLocation`). For example, a video introducing diabetes is directed at Dutch children, while a video on the gluca pen is in Italian:

```
pom:watchVDDB om:needsLocation pom:hospital_2
pom:watchVDDB om:needsLocation pom:hospital_3
pom:watchVGP om:needsLocation pom:hospital_4
```

Listing 9: Example for task object property.

In addition, subclasses for specific activities within the ITS may have extra properties. In our case, `QuizTask` has a property `om:numOfQuestion` to store the number of questions fitting a specific task.

```
pom:answerMBG3 om:numOfQuestion "12"^^xsd:int
```

Listing 10: Example for task subclass property.

4) Object Properties: Object properties carry additional information about a learning objective. Information presentation can be more rich compared to data properties because the object properties can have data properties themselves. For example, setting an icon for a topic presenting the topic in the user interfaces. Learning goal type has four object properties: topic, difficulty, skill, and insulin device. Task type has one object property: hospital. These objects are discussed as follows.

a) Topic: Topic is the most straightforward property indicating the general subject of a learning goal; for example, nutrition, carbohydrates, or insulin (The topics of an achievement are derived from the set of topics of included goals). It carries a label, selected from a predefined set, and a reference to a stored icon representing the topic.

```
pom:carbs2 om:hasTopic pom:nutrition
pom:nutrition om:label "Nutrition"^^xsd:string
pom:nutrition om:hasIcon "images/food.png"^^xsd:anyURI
```

Listing 11: Object property topic example.

b) Difficulty: Difficulty level indicates the general level of expertise of a learning goal on a scale [1–6].

The concept is based on skill trees used in games and provides a directed hierarchical (from easy to more advanced per subject) data structure. In the case of learning goals, it prescribes the assumed proficiency level of the child on a given topic allowing for personalized scaffolding; objectives are selected through assessment and authoring, constructing an individual learning path.

Each goal is assigned a difficulty level from a fixed set (i.e., rdf individuals: novice, advanced beginner, competent, proficient, expert, master). Each difficulty has a label and color for presentation in the user interfaces and a suggested age range (see Listing 12). The age range indicates the approximate age group of children for whom a specified goal is likely suitable. For example, a young child is expected to have goals on the first or second level rather on the final, depending on their diabetes onset, and developmental level. Important is that difficulty is not an age restriction, meaning that a child can progress on different difficulty levels depending on topic.

```
pom:carbs2 om:hasDifficulty om:difficulty2
om:difficulty2 om:label "Advanced beginner"^^xsd:string
om:difficulty2 om:hasColor "ff7f50"^^xsd:string
om:difficulty2 om:ageRange "7-9"^^xsd:string
```

Listing 12: Object property difficulty example.

c) Skill: Skill level specifies the complexity and specificity of each learning goal based on Bloom's taxonomy of learning objectives (see Section II-A).

Skill is used to organize preceding and succeeding goals (within or over difficulty levels); a goal aimed to apply a specific piece of knowledge is typically learned only after, respectively, understanding and remembering this knowledge.

Each goal is assigned a skill level from a fixed set of 16 levels (over three classes, i.e., knowledge, attitude, and psychomotor; see Listing 13, lines 1–3, for an example) defined in the bloom ontology. The OM imports this ontology of skill levels based on the on the taxonomy of Bloom [16], where `om:Skill` $\equiv$ `bloom:LearningObjective`. The current (diabetes) domain knowledge base includes mainly cognitive goals to remember or understand and goals to execute a physical action requiring application of cognitive skills. The chosen implementation allows for extension and makes this limitation transparent.

d) Device: Device specifies any technological artifact involved in a learning goal. In the PAL case, an insulin device is required to complete either an injection or bolus (see Listing 14, lines 1 and 2), where `pom:InsulinDevice` is a subclass of

```
pom:carbs2 om:hasSkill bloom:remember
bloom:remember rdf:type bloom:Remembering
bloom:Remembering rdf:subClassOf bloom:Knowledge
bloom:remember bloom:hasColor "2d3092"^^xsd:string
bloom:remember bloom:hasIcon "images/brain-blue.png"
    ^^xsd:anyURI
```

Listing 13: Object property skill example.

om:Device, and the insulin device is used to personalize learning goals by relevance. Device has data properties om:label and om:hasIcon for interface presentation.

```
pom:bolus1 om:needsDevice pom:pump
pom:injectInsulinHelp2 om:needsDevice pom:pen
pom:pump om:label "Pump"^^xsd:string
pom:pump om:hasIcon "f566"^^xsd:string
pom:pen om:label "Pen"^^xsd:string
pom:pen om:hasIcon "<i class='fas fa-syringe'></i>"
    ^^xsd:string
```

Listing 14: Object property device example.

e) Location: Location specifies a physical place where the child is located at or related to. In the PAL case, the location refers to the hospital where the child is a patient (see Listing 15), where pom:Hospital is a subclass of om:Location, and the hospital location is used to personalize learning content in language and facilitate treatment plan differences between hospitals. For example, whereas the participating Italian hospital preferred dietary restrictions, the Dutch hospitals prefer insulin corrections. Location has a om:label and om:address.

```
pom:watchVDDB om:needsLocation pom:hospital_2
pom:hospital_2 om:address "Ede, Netherland"^^xsd:
    string
pom:hospital_2 om:label "Gelderse Valei"^^xsd:string
```

Listing 15: Object property device example.

B. User Model

The user model is an overlay model representing the current state of domain knowledge of a particular learner. In other words, the user's individual objectives with for each objective the progress value and a reference to an domain specific objective (i.e., an knowledge element of one of the classes *Type, with the relation between goals and achievements, and goals and tasks is inherited from this type). The key principle of the overlay model is that for each objective in the domain model, an individual user knowledge model stores the progress of the user on this objective. This model functions as an internal memory for the system to know what tasks the user did or did not do and what knowledge is or is not attained.

The user model (see Fig. 4) includes the Child with some data type properties that describe the child (e.g., name, birth date, and gender) and some object properties useful for personalized goal setting. Furthermore, there are three object properties relating to three objectives classes within the user model (i.e., LearningGoal, Achievement, and LearningTask).

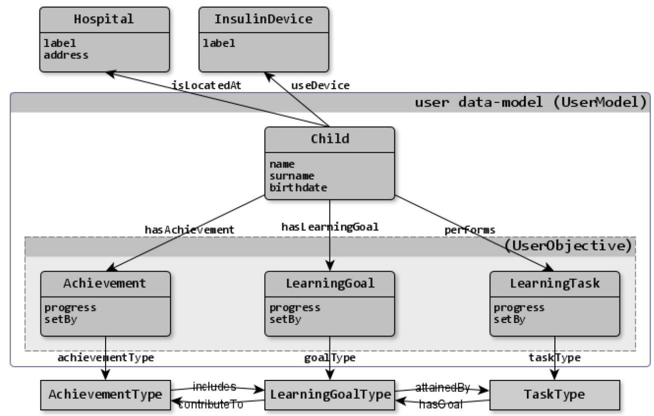


Fig. 4. Objectives model upper user model classes, properties, and relations.

1) Child: Child defines a specific child. An individual of the class Child presents the characteristics of the child used by the system, including progress on each of the objectives. Child is defined in the PALO domain ontology (dom); individuals are given the prefix pal. The *OM* uses data type properties that are crucial for personalization of the educational path such as age (derived from birth date) to select a starting point for suggested skill level and, for the diabetes domain relevant property, gender to include or exclude specific goals such as those related to period. Furthermore, the *OM* uses object properties dom:useDevice and dom:isLocatedAt for personalization (see Listing 16). These data are added via the PALControl application in consultation with the health-care professional (HCP). Other characteristics defined in the PALO domain ontology such as relatives and preferences for color are omitted because they are irrelevant for personalized learning.

```
pal:child1 rdf:type dom:Child
pal:child1 dom:hasName "Henk"^^xsd:string
pal:child1 dom:hasSurname "Jansen"^^xsd:string
pal:child1 dom:birthdate "2006-04-12"^^xsd:date
pal:child1 dom:hasGender dom:male
pal:child1 dom:useDevice dom:pen
pal:child1 dom:isLocatedAt pal:hospital_2
```

Listing 16: Data and object properties that define a child and may be used for personalization example.

The *OM* user model extends the class Child from PALO with three objects properties (om:hasLearningGoal, om:performs, and om:hasAchievement) to store child-specific progress data on learning goals, learning tasks, and achievements (see Listing 17).

```
pal:child1 om:hasLearningGoal pom:henksgoal1
pal:child1 om:hasLearningGoal pom:henksgoal2
pal:child1 om:hasAchievement pom:achievement283
pal:child1 om:performs pom:answerI1henk
```

Listing 17: Child data and object properties example.

2) User Objectives: User objectives are included in the user model as well (i.e., Achievement, LearningGoal, and

`LearningTask`). They are related to the child by the object properties presented above. A child can have this personal objectives for each of the DKE's in the knowledge base. Each of the three classes has the data properties `om:progress` and `om:setBy`, which, respectively, store the child's percentage of attainment of the objective and who last updated the progress value (see Listing 18). The latter can have values either "system" or "user," where system means the child–system interaction resulted in progress update and user means the value was last manually updated via an authoring tool.

```
pom:henksgoal1 om:hasProgress "0.8"^^xsd:float
pom:henksgoal1 om:setBy "system"^^xsd:string
pom:insulingoalhenk om:hasProgress "0.0"^^xsd:float
pom:insulingoalhenk om:setBy "user"^^xsd:string
```

Listing 18: Data properties of objectives example.

Furthermore, each class has its own object property relating it to the specific knowledge elements, individuals in the domain ontology (see Listing 19).

```
pom:henksgoal1 om:goalType pom:carbs2
pom:answerI1henk om:taskType pom:answerI1
pom:achievement283 om:achievementType pom:sleepover
```

Listing 19: Object properties of objectives example.

C. Content Model

The content model includes a concept-based organization of educational materials such as the actual videos and questions for quizzes. Individuals of the class `Material` list all relevant materials and connect specific content with one or more learning goals via tasks with object property `om:forTask` (see Listing 20).

```
pom:video1 rdf:type om:Material
pom:video1 om:forTask pom:WatchVVDB
pom:WatchVVDB om:hasGoal pom:BasicPrinciplesKnow2
```

Listing 20: Materials related to learning goals example.

`Material` has two data properties: `om:medium` and `om:link`. The first defines the sort of material and can be any of "web," "video," or "app," with app meaning that the material is included in the ITS system and web or video is linking to any source on the world wide web. Dependent on the value for `om:medium`, `om:link` is either an external hyperlink or in-app location, as shown in Listing 21.

```
pom:video1 om:medium "video"^^xsd:string
pom:video1 om:link "https://www.youtube.com/watch?v=
    K4UsrTNz37I"^^xsd:anyURI
pom:answerI1 om:medium "app"^^xsd:string
pom:answerI1 om:link "QUIZ$I1"^^xsd:anyURI
pom:BS_PlayNEH om:medium "app"^^xsd:string
pom:BS_PlayNEH om:link pom:"
    GAME$BREAKSORT$EASY$NUTRIENTS"^^xsd:anyURI
```

Listing 21: Material data type property example.

IV. Case: Diabetes Self-Management Education

The *OM* of the previous Section III provides the foundation of the personalized self-regulated learning support for children with diabetes that has been developed in the PAL project. This project has followed a human-centered iterative development process, in which the upper ontology was instantiated in a *POM* (see Listings in Section III) for diabetes self-management in the Netherlands and Italy. Both *POM* and *OM* have been integrated in the so-called PALO ontology—a set of ontologies specific to the PAL system. In the rest of this section, we describe the domain knowledge acquisition and modeling, the domain ontology development, and the deployment of this ontology within the PAL system.

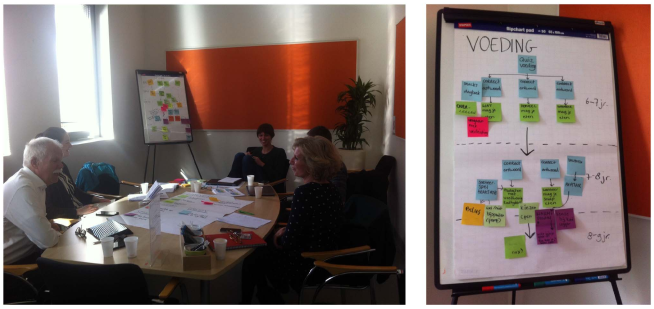
A. Content Creation

Acquiring the domain knowledge and deriving a coherent and concise set of learning objectives out of it consisted of complementary activities. First, we formalized diabetes self-management training goals based on an existing list of guidelines of knowledge and skills. Second, we organized expert sessions to introduce the conceptualization of the knowledge structure in the upper ontology and to collaboratively construct individuals for achievements and tasks. Then, we developed and provided an authoring tool to facilitate experts in creating their own goals, achievements, and tasks for diabetes self-management education in a format that matches with the overall ontology structure.

1) Formalization of Existing Guidelines: To support personalized learning of diabetes self-management skills, we needed to know the relevant DKEs, the education paths toward mastering these, and a structured way to include the DKEs in an ITS. We constructed a first set of DKEs from a topic-based list of knowledge and skill goals (Weet- & doe-doelen; see Section II-C). We formalized the goals in *POM* as `LearningGoalType` and added the `om:description` and the Bloom level (`om:hasSkill`) among the other properties listed in Section III. Furthermore, we made explicit the underlying hierarchical structure in `om:requires`. We added achievements in `AchievementType` as substantial goals that can only be attained by first attaining a set of smaller goals. A first version of the "goal-tree" was implemented in OWL/RDF and integrated in the PALO ontology, which was connected to the PAL system [36], [38], [39]

2) Expert Sessions: To improve the content of the domain model (*POM*), we consulted domain experts. The aim of these sessions was to get them acquainted with the model and to elicit content. Pediatricians specialized in diabetes type 1 introduced the upper ontology (*OM*). In this phase, a doctor and nurses from two Dutch hospitals were involved. In brainstorm sessions, domain-specific achievements, goals, and tasks were gathered. The HCPs that participated in these sessions were already involved in previous iterations of the PAL project and had some experience on describing achievements, goals, and tasks on diabetes self-management for children.

The process of these sessions was as follows: The researcher would suggest a topic, such as nutrition. The HCPs were asked to think of as many learning goals as possible. Next, these goals



(a) Session in progress (b) Result for one topic

Fig. 5. Expert session for domain content creation. (a) Session in progress. (b) Result for one topic.

would be grouped by age range (the age a child is likely to be able to comprehend the knowledge or skill). Within each age range, clusters were formed, and given a description, an achievement was defined. Then experts were asked to think of possible activities to train each knowledge element. Fig. 5(b) shows the resulting objectives structure for the topic of nutrition.

3) Collaborative Construction of Ontologies: Subsequently, we had several intensive sessions with experts from the two Dutch hospitals as well as the Dutch diabetes association, followed by sessions with experts from an Italian hospital to check and refine the model and documentation. The PAL project used the Wiki Socio-Cognitive Engineering (WiSCE) tool[7] for the project team to collaborate and share knowledge efficiently. It is built and maintained in Atlassian Confluence,[8] providing guidance and structure for the different stakeholders involved in the research and development process. To make the knowledge of HCPs explicit and to facilitate authoring of the domain model by HCPs, WiSCE provided shared templates in which they could add and edit the content directly (see Fig. 6). The templates facilitated joined content creation and refinement, assuring compliance with the ontology structure (consistency and completeness).

The information gathering process was as follows: the HCPs were asked to think about achievements that are relevant in the daily lives of children with type 1 diabetes. After that, we went through each achievement asking the HCPs which knowledge and skills a child should have to attain this achievement and attach a corresponding development level per skill. Gathered content was recorded in the Confluence templates and used in later sessions for refinement and definition of new content. When no new achievements and goals were raised, the sessions were devoted on selecting appropriate (learning) tasks for each goal per skill level. These were also documenting in Confluence. The resulting content was reviewed by other HCPs.

While linking goals and achievements, we experienced that most achievements require a similar subset of goals representing basic principles of diabetes self-management. Therefore, a new achievement type was created holding the basic principles for each development level. Children are required to complete these basic principle achievements as a prerequisite for other achievements on that developmental level. This way, an appropriate foundation on disease self-management knowledge can be assured. Furthermore, the construction of achievements centered around daily life challenges resulted in multiple achievements with similar type of subjects (e.g., cycle to school and play outside, both addressing physical activity) and, therefore, having shared goals. As a result, a child can attain a goal that results in progress on multiple achievements, a skill that can be applied in different contexts.

[7][Online]. Available: https://scetool.ewi.tudelft.nl
[8][Online]. Available: https://www.atlassian.com/software/confluence

Goal: Interpret a blood glucose measurement

<table>
<tr><td>ID</td><td>IM1.1, IM2.1, IM2.1, IM3.1, IM3.2, IM4.1, IM4.2</td></tr>
<tr><td>Description</td><td>The child needs to learn how to interpret the blood glucose measurement.

The child needs to know what acceptable values are (skill level 1), need to understand the effect of a value (skill level 2), the child needs to learn how to interpret the values with help (skill level 3) and eventually needs to learn how to interpret the values on its own (skill level 4).

I can interpret a blood glucose measurement with help - I can interpret a blood glucose measurement myself</td></tr>
<tr><td>Skill</td><td>
<table>
<tr><th>Development level</th><th>Skill level</th><th>Tasks</th></tr>
<tr><td>1</td><td>I know

I know which glycemic values are fine for me.</td><td>QUIZ (QIM1)

B&S (BIM1)</td></tr>
<tr><td>2</td><td>I know

I know the which action to perform for each glycemic value.</td><td>QUIZ (QIM2)

MEMORY(MIM2)</td></tr>
<tr><td>2</td><td>I can do myself

I can detect when an action is needed based on my glycemic value myself.</td><td>REAL WORLD TASK (RIM2)</td></tr>
<tr><td>3</td><td>I understand

I understand which action action to perform for each glycemic value.</td><td>QUIZ (QIM3)

MEMORY (MIM3)</td></tr>
<tr><td>3</td><td>I can do with help

I can perform the correct action to improve my glycemic value with help.</td><td>TIMELINE TASK (TLG3)</td></tr>
<tr><td>4</td><td>I know

I know what to take into account while reasoning about my glycemic value.</td><td>QUIZ (QIM4)</td></tr>
<tr><td>4</td><td>I can do myself

I can perform the correct action to improve my glycemic value myself.</td><td>REAL WORLD TASK (RIM4)</td></tr>
</table>
</td></tr>
<tr><td>Achievements</td><td>Swimming, Sleepover, Go on Holiday, School camp, Being ill, Sports game, Sports training, Cycle to school, Go on Holiday, Dinner in restaurant, Roller coaster, Lazy weekend</td></tr>
<tr><td>Topic(s)</td><td>Glucose measurement</td></tr>
<tr><td>Dependencies</td><td>The skills in this goal are only dependent on each other in terms of development level (i.e. one should posses all skill on a lower development level before it is able to achieve a skill on a higher development level)</td></tr>
</table>

Fig. 6. WiSCE's Confluence template example filled for a specific goal.

B. PAL System

This section highlights the usage of the *POM* as implemented in the PAL system, a diabetes education framework aimed to provide goal-based personalized guidance in self-management training. The PAL system is an mHealth application to support children of 7–14 years old with diabetes mellitus type 1 in their self-management training. Kaptein et al. [39] provide a comprehensive overview of the system components; in this section, we will elaborate on the role of *POM* in goal setting and adaptation of system interaction.

The *POM* is integrated in the core of the PAL system and three user interfaces. The interfaces visualize the relations within *POM*, facilitate the selection of personal goals and tasks, and show feedback on learning progress to three different user groups (i.e., children, informal caregivers, and HCPs). The PAL core—the "brain" of the system—calculates progress based on task performance, generates verbal feedback on progress, and adapts the nonverbal behavior of the robotic PAL based on task selection.

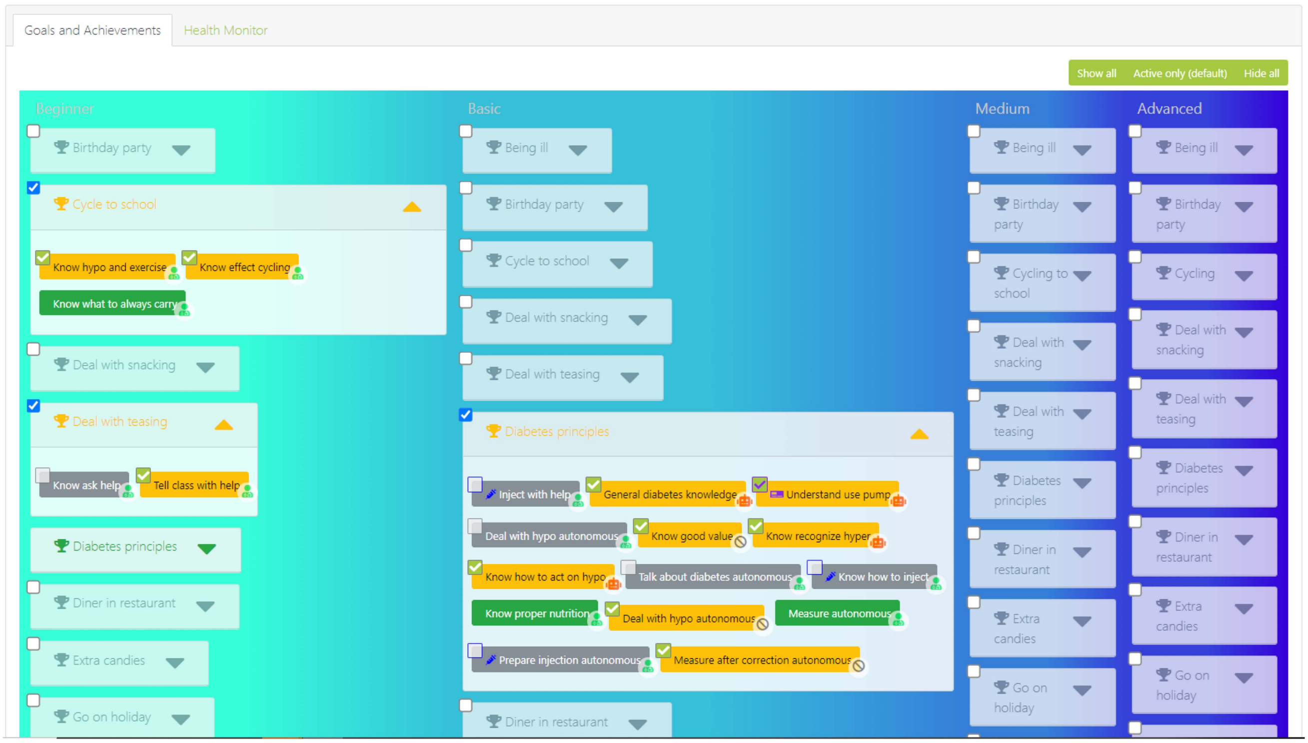


Fig. 7. Visualization and goal selection in *palControl&Inform*.

1) Authoring: For authoring of personal learning goals, a web-based application, *palControl*, was developed (see Fig. 7). The user interface visualized the goal hierarchy and relations with achievements encoded in `om:requires`, `om:includes`, and `om:contributeTo`. The logic within the *POM* ontology supported goal selection. Besides, the tool was used to add new children and provide information to the user model such as the `InsulinDevice` and `Hospital` of the child and the child's personal `LearningGoal` and `Achievement` selection. An earlier version of the tool was described and evaluated in [36]. Next, we will briefly describe the process of goal selection.

a) Visualization of achievements and goal hierarchy: The interface shows all the achievements and an option to (un)fold specific or all achievements to show/hide the goals within each achievement. The achievements are ordered from left to right based on their `Difficulty`. Both achievements and goals are color-coded indicating their status (derived from the value in `om:progress`): gray depicts an inactive objective, yellow an active one, and green an attained one. Goals may hold icons for who last updated the status (system or user; `om:setBy`) and to indicate their relevance for either pen or pump users (`om:needsDevice`; derived from the associated `LearningGoalType`). A detailed description of each goal is given upon mouseover.

b) Collaborative goal setting: In a face-to-face meeting, the HCP and child will discuss challenges the child faces in his/her daily life. Based on this, they will manually select a relevant achievement (`AchievementType`) or individual goals (`LearningGoalType`). A future development would be to select the first goals based on the outcome of an assessment upon intake of the child. For each selected objective, individuals (`Achievement` and `LearningGoal`) that track the `om:progress` of the child are created in accordance.

c) Automated selection: Based on the manual selection, an automated selection mechanism will ensure that all required achievements, goals, and tasks are selected based on relations in the *POM* ontology. The selection mechanism activates an achievement whenever one or more goals within are activated (`om:contributeTo`), and it activates all goals within an achievement whenever an achievement is selected (`om:includes`). When activating goals, it will take into consideration relevance of these goals for the specific child based on the child's `Device` (i.e., pen or pump). And for each activated goal, the related tasks are activated (`om:attainedBy`) taking into consideration their relevance based on the `Location` (i.e., the Netherlands or Italy) of the child. Furthermore, for each items, the hierarchy is traced, selecting prerequisite objectives as well (`om:requires`).

2) Learning Activities: Children can develop their knowledge and skills by performing activities in the mobile *myPALapp*. The application contains various games, a medical diary, and the *myPALObjectives Dashboard* (*cPOD*, Fig. 8). The *cPOD* is described and evaluated before in [40]. Here, we will present the important functionality of the *POM* in the *myPALapp*, specifically the *cPOD*.

a) Awareness of objectives: The first aim was to visualize the objectives to the children and create awareness of their personal learning path. The interface shows all active achievements, the

Fig. 8. Progress tracking and task selection for children in *myPALObjectives Dashboard*.

goals within the achievement, and tasks to attain each goal. Inactive goals (`LearningGoal` with `om:progress` 0.0) are shown as well, so the child knows what more is to come to complete the achievement in the near future. Also, the child can see more advanced `Difficulty` levels of the achievement that can be made available later on in live. For each achievement and goal, a clear progress indicator is shown, so the child knows how much activity is required for attainment. Thus, the interface combines information from both the domain model and the user model.

c) Task selection: From the *cPOD*, the child can directly start the learning activity required to progress toward goal attainment. Next to each goal is a button opening a new screen that displays all related and relevant (based on `Device`) tasks. Whenever a goal—and thus related tasks—is selected on multiple difficulty levels (i.e., prerequisites are not yet met), the system will offer the tasks on the lowest, active level.

d) Stylized behaviors: Based on the selection of a specific task, the behavior of the PAL actor (robot or virtual) is adapted to act more as an instructor or more peer-like behavior. The robot roles suitable for each activity are defined in the PiNS ontology, and a behavior manager is responsible for the selection of according nonverbal behaviors from a predefined and annotated set of behaviors.

d) Feedback on performance: The PAL actor will also provide feedback on goal attainment and progress to motivate the child. An episodic memory module and ontology are connected to the *POM*; they use the progress data to make remarks about recent attained goals and ones that showed no progress for some time.

3) Attainment Visualization: An important function of usage of *POM* is to give insight into the progress of the child toward autonomous self-management to the child itself (as described above), to the informal caregiver (in PALInform), and to HCPs (in PALControl).

a) Goal calculation: First, progress on each objective is calculated based on the relations between objectives defined in the ontology and a child's task performance. Upon the activation of objectives, an individual is created in the child model with `om:hasProgress` 0.0. Whenever a task is completed successfully, the progress is updated to 1.0. Also, the progress for the related goal is recomputed; a percentage of completion is calculated based on the total number of *available* tasks for this goal and the number of these tasks that are completed. Finally, the achievement progress is updated for all active achievements in which the goal is included. Here, the percentage of completion is calculated based on the total number of *relevant* goals in a specific achievement and the number of them that are attained. (Whenever a previous inactive achievement is activated, the progress will be set dependent on included goals already attained.)

b) Visualization: HCPs can view the child's progress in *palControl* (see Fig. 7). Green nodes indicate completed achievements and goals (`om:progress` is 1.0). Upon mouseover on the setter icon, the completion date is shown. This view gives a full overview of the status of all achievements. The HCPs can also access *palInform* (see Fig. 9) to view a timeline indicating dates on which the child attained a new goal or achievement. The view can focus on weekly, monthly, or quarterly overview. *palInform* is also accessible to authorized informal caregiver such as parents. The goal tree is visible for parents as well, but opposed to HCPs, they cannot edit goal status.

C. Evaluation Results and Discussion

We evaluated the model with three user groups: 1) children with T1DM; 2) their parents as informal caregiver; and 3) HCPs (i.e., pediatric diabetes nurses and/or doctors). With each user group, we evaluated the *OM* data model (i.e., the class definitions: concepts and relations), the domain-specific content defined in *POM* (i.e., the individuals that are concrete achievements, goals, tasks, and accompanying learning materials), the user interfaces (respectively, the *myPALObjectives Dashboard* for children (*cPOD*) and *palControl&Inform*), and the functionality in the *PALsystem* depending on the *POM* (i.e., how it supported motivation, personalization, and adaptation of the system). These evaluations were part of the integrated PAL 3.x evaluation, divided in two cycles (3.0 and 3.5) running May 2018–January 2019. In the rest of this section, we present the results per user group and briefly discuss these results.

1) Children: The *POM* data model and content, as well as the *cPOD*, were evaluated with children both structured in a usability study and "in the wild" as part of the integrated PAL 3.x experiment cycles.

The usability study ($N = 12$) has been reported extensively in earlier work [40]. From this study, we learned that the *cPOD* was insufficiently clarifying the different concepts and relations in the *POM* (i.e., the achievement, learning goal, and task as well as properties such as difficulty levels and the difference between global and personal data). Therefore, we were unable to evaluate whether the *POM* data model and content were fitting user needs. This user interface was improved upon for the integrated experiments.

During the PAL 3.x experiment, 24 Dutch children used the *myPALapp* for several weeks. In the 3.0 cycle, children ($n = 14$) did have personal learning objectives, but generic activity content, in the 3.5 cycle ($n = 10$), activity content was

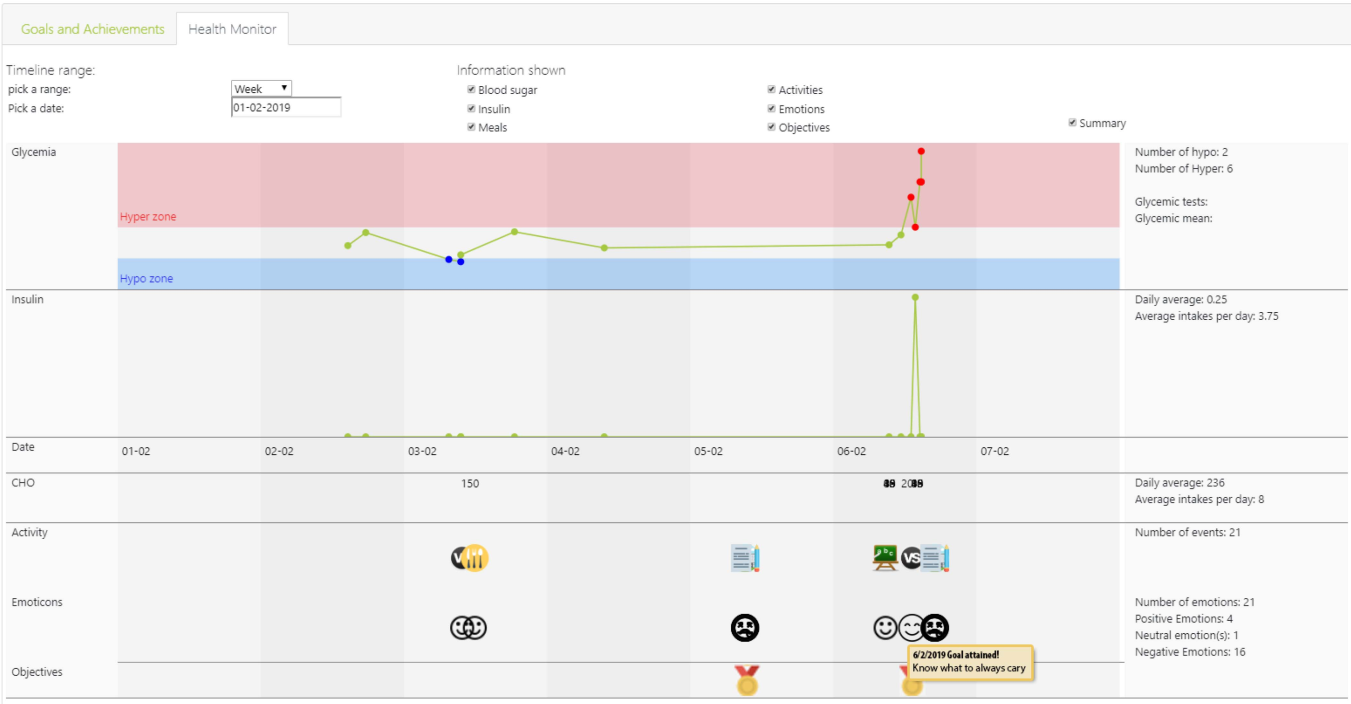


Fig. 9. Progress tracking for (informal) caregivers in *palControl&Inform*.

adapted toward personal objectives. Afterward, children filled a questionnaire about the *myPALapp* with a number of questions dedicated to the *cPOD*. The first seven questions were statements to be rated on a five-point scale: either fully agree, partially agree, don't know, partially disagree, or fully disagree. This was followed by four open-ended questions.

The majority of children (3.0 $n = 11$, 3.5 $n = 9$) agreed (partially) that it is important to have personal goals for disease management, and that the goals were helpful in disease management (3.0 $n = 7$, 3.5 $n = 9$). Furthermore, children believed tasks to support goal attainment (3.0 $n = 10$, 3.5 $n = 5$). Self-reported goal attainment and task completion varied between children: In the 3.0 cycle, six children (partially) agreed to have attained all their goals, opposed to five children that (partially) disagreed. In the 3.5 cycle, three children reported to have attained their goals and four reported not to have done so. In the 3.0 cycle, five children (partially) agreed to have completed all their tasks, opposed to six that (partially) disagreed. In the 3.5 cycle, four children reported to have completed their tasks and three (partially) disagreed to have done so. Children (partially) agreed that the activities in the *myPALapp* were supporting training (3.0 $n = 12$, 3.5 $n = 8$) and attaining (3.0 $n = 10$, 3.5 $n = 7$) of their goals.

For the open questions, eight children were excluded because they either did not answer ($n = 6$) or answered about the *myPALapp* in general and not specifically the *myPALObjectives Dashboard* (*cPOD*, $n = 2$). Children liked about the *cPOD* that they could set their personal goals and that they could undertake activities and attain a goal/achievement, being able to see when they attained a goal/achievement, seeing their progressing toward attainment, being rewarded with points (allowing them to customize the game background and robot dance). Children reported less dislikes; five children disliked nothing. Incidental dislikes were: not being able to set new goals, not attaining a goal for each activity done, and "mean feedback" given by the robot. Seven children self-reported to have understood everything about the *cPOD*. Three children did not understand everything but did not specify what was unclear. Specific responses were that it was unclear where to find the achievements, how progress was reached, and why some achievements were not attained. None of the children provided suggestion to improve the *cPOD*, only for additional rewards, such as new games and robot clothing.

It is doubtful whether children consciously used the *myPALObjectives Dashboard* and as a consequence experienced increased motivation to attain their goals and achievements. Three children admittedly never looked at the *cPOD*. Researcher observation suggests that young children never used the *cPOD*, whereas some older children did look at and consciously attempted to attain their objectives; however, this was not reflected in the questionnaire responses that did, on average, not vary between the older and younger children. Nonetheless, children did play games and reached goal attainment (unconscious). In cycle 3.5, the *POM* facilitated personalized content (opposed to cycle 3.0 where generic content was available to all children). For example, children were presented only those quiz questions related to and contributing toward the attainment of selected goals. This personalization of content was noted by some of the children. However, it did not yield significant differences in questionnaire responses (although children proved to learn more with the PAL system in general [11]). It should be noted that a relatively small number of children participated in this evaluation. The results give rise to better integrate the dashboard in the overall PAL usage (including the provision of guiding avatar dialogues) and, subsequently, test the effectiveness of the *myPALapp* in a controlled experiment with and without the *cPOD*.

2) Parents: A total of ten parents, who participated in the PAL3.x study, filled a questionnaire and four of them participated in a semistructured interview about *palControl&Inform*. The questionnaire included questions about their usage and satisfaction of the *palControl&Inform*; the interview further elaborated on this and asked them about future expectations.

None of the parents ever used *palControl&Inform* due to various reasons: parents forgot its existence ($n = 2$), or were unaware of its existence ($n = 3$), checked progress in the *myPALapp* ($n = 1$), or reported their child never to have used the *myPALapp* ($n = 2$). Two parents did not provide a reason for never using *palControl&Inform*. Due to the lack of usage of *palControl&Inform*, we could not collect information about the parents understanding of the *POM* and implementation thereof in the user interface. Albeit never using the app, parents did report to expect added value of objectives in the child's self-management development (3 partial agree, 4 fully agree), agreed that insight in their child's progress is valuable (5 fully, 2 partial), and hold a strong preference for being able to author their child's objectives (5 fully agree, 2 partial agree). One parent commented that goal setting via consultation with the HCP (as is the designed practice) was the favorable process.

Upon further questioning, while looking at *palControl&Inform*, parents elaborated further on their opinions. Parents expressed personal preferences, such as updates via e-mail and goals including personal values. One of the four interviewed parents reported their child to purposely pursue goal attainment; two other parents believed that the personalized game content facilitated relevant learner experience and, consequently, contributed toward self-management learning.

Even though parents agree that insight into their child's self-management goals is important, none of them ever used the tool offered. This indicated that the tool did not meet their needs. The fact that suggestions were given to send an e-mail about it indicated that parents are too distracted to remember to check the *palControl&Inform*. A follow-up study should acquire further requirements to improve the strategies and user interfaces to support this user group for their role in child's self-management learning.

3) Health-Care Professional: HCPs were heavily involved in the development of the *OM* and *POM*; therefore, we did not further evaluate the model with these professionals. To check where the model was understandable for others as well, we showed the *palControl&Inform* to a pediatric diabetes nurse that was involved in the PAL project before but not familiar with this version of the ontology. This session resulted in the following remarks: the achievements and learning goals are formulated and visualized clearly. Also, the difficulty levels were clear and fitting the objectives. Goal setting works quicker compared to the earlier version; the HCP was able to select goals for a child within a few minutes. At first, the HCP did not notice the automated selection of prerequisites; this was due to some lag in the visualization of updated goal/achievement status. This lag also resulted in the HCP multiple times clicking the same node causing it to deactivate unintended. The domain-specific goals were evaluated as complete, clear, relevant, and fitting yet challenging for children aged 7–14. Formulation of achievements as daily live challenged was appreciated. The HCP did mention similarity between goals, indicating that it was unclear that goals could be "repeated" on a different difficulty level.

V. Discussion

To support personalizing of a learning path in self-regulated learning, we created a knowledge base of domain objects and objectives. This ontology is different from existing works, such as [33] and [34], that personalize content but do not take into account learning objectives and the learners real-time state thereof, while educational theories state the importance of the learning objectives as starting point for the personalization of education (see, e.g., [13]).

We have created a domain-independent *OM* and individuals specific to the diabetes self-management domain. The upper ontology provided the foundation of a comprehensive and coherent set of achievements, goals, and tasks, as well as the storage of child data necessary for personalized goal setting. The model was successfully integrated in the PAL learning support system for children (aged 7–12) with diabetes. The inclusion of the *POM* showed valuable in creating the awareness of personal goals, monitoring of progress, offering of relevant tasks, and tailored feedback. In the evaluation of the overall system [11], the children proved to learn more with the PAL system compared to care as usual. However, the specific contribution of the *POM* on the observed learning gain needs to be assessed in future studies.

The current domain-specific individuals have been created in collaboration with diabetes care experts from both the Netherlands and Italy (which have a different view on the diabetes regime and self-management support), resulting in an ontology that allows for the location-based customization of tasks. To establish the goal set and learning content was a demanding (multidisciplinary) job. Upscaling of the PAL system to other hospitals and countries might require further extensions of the domain-specific models. For this, the availability of a knowledge base to choose objectives and an authoring tool to make some adjustments would be very helpful.

More specifically, the role of the "topic" description and assignment of "difficulty" has been subject of discussion throughout the development of *POM*. Each goal has a difficulty level and one or more topics. Achievements are "empty" shells to structure these goals; they do not define knowledge but structure DKEs. Achievements are important concepts for the HCP to summarize goals and have an overview of the knowledge base. This raises the question whether achievements should get a property of topic and difficulty that is filled with the information inherited from included goals. This functionality is not supported in RDF; thus, it would require an external script scraping this information. Alternatively, an application programming interface (API) function can query for this information and show it in the interface without it being written into the knowledge base. This functionality would support topic-based achievement searching.

For the specifications of a specific goal, it is important to first define the desired end knowledge state and then trace back on the required knowledge. The preceding knowledge levels proved to span across the three different Bloom pyramids (i.e., cognitive, affective, and psychomotor skills), and consequently, we had to apply the Bloom taxonomy in a flexible way. For example, to know when to administer insulin, the child should first understand the effect of insulin. And to be able to administer insulin (skill), a child first needs to know where injections can be given (factual knowledge). More generally, the learning goals that the PAL system can accommodate will be constrained by the technological capabilities of the interactive agent, the content of the educative games, and the monitoring possibilities. The learning with the PAL system is intended to *supplement* or *augment* the teaching of the caregivers and parents, who supervise the process and complement where needed.

The *OM* provides the foundation of an interactive learning environment that supports teachers and learners in a comprehensive and coherent way with the personalized goal setting and task planning, and the progress monitoring and agent-based feedback. The ontology was developed for this purpose and, consequently, differs from other ontologies that have been developed for different purposes and applications. Corresponding to the ontology development for automated scenario-based training [41], our *POM* distinguishes an upper and lower ontologies supporting the learning of "frames" or "mental models" with less emphasis on the modeling of the physical environment and game dynamics. Similar to the ontology for serious games [42], our *POM* assesses the individual performance for personalization (however with less emphasis on learner style). Furthermore, our model bares resemblance to the domain and user models in the domain of adaptive hypermedia (AH) (see for an overview [6]) with less emphasis on the modality of the content and knowledge types. We started to construct a concise model, focusing on the learning objectives and tasks for personalization and human–agent communication. The ontologies for scenario-based training, serious games, and AH could provide elements to refine the user and domain models of the *OM*.

VI. Conclusion

A. Summary

To answer the first research question ("How to model learning objectives?"), we looked into how learning goals are formulated in pedagogy and ontologies for education. Effective learning goals must attune to the appropriate level; a way to structure this is Bloom's taxonomy [16], [19]. Furthermore, a learning goal must have attributes presenting relations and descriptions [13], [14], [27], [30]. Formalization in an ontology supports shared understanding and automated reasoning [12]. We modeled educational objectives (i.e., achievements, learning goals, and accompanying tasks). The upper ontology structured the classes and relations and defined domain-independent constructs (i.e., levels and topic). The domain model specified diabetes self-management training objectives for young children. These objectives were defined with experts (pediatric doctors and diabetes nurses) and considered relevant to and covering the diabetes domain to a considerable extend. From this, we conclude that our upper model adequately supported the formalization of implicit knowledge of HCPs on diabetes self-management training. To answer the second research question ("How to deploy the model in a learning environment?"), we took a more practical approach by implementing the model successfully in an multimodal (robot, tablet, and web applications) system providing personalized learning content. A field study with children with type 1 diabetes in the Netherlands and Italy showed our model to contribute in support of children's basic needs (autonomy, competence, and relatedness) [11]. However, there is room for improvement in user interface of accompanying materials for progress toward goal attainment and a more advanced personalized guidance selection mechanism.

In the rest of this section, we will highlight important lessons learned in the form of guidelines for the formalization of learning objectives and future challenges and opportunities for this line of work.

B. Design Guidelines

Recommendations for formalization of domain-specific learning goals, achievements, tasks, and materials in the *OM* knowledge structure, and integration thereof in a multimodal ITS are as follows.

1) Consult domain and pedagogic experts to make an inventory of important learning goals and define learning activities (tasks and accompanying materials) to train goals.
2) Formulate achievements from logical learning units (e.g., daily challenges) that require a subset of the knowledge and skills encapsulated in the goals to improve relevance.
3) Formulate achievements and goals from the perspective of the child. This facilitates a sense of ownership and increases experienced relevance.
4) Provide instruction and explanation to the child on how achievements, goals, and task are selected and can be attained (i.e., that progress on a goal is gained by task completion, and benefits earned by this).
5) Define user characteristics and relate them to goals and/or tasks to indicate personal relevance.
6) Embed the objectives in the ITS application by making easily accessible to the (child) user and integrate them in other system functionality such as feedback provided by an pedagogical agent.

C. Challenges and Opportunities

Suggestions for future work in terms of considerate additions to and usage of the *OM*are as follows.

1) Children showed difficulty understanding the concepts (achievement, goal, and task) and the way these items were

intended to guide their learning path. It is worthwhile to further explore and understand these difficulties in order to update the model and improve the instructions and explanations for better understanding.

2) The acquisition and formalization of objectives is a time-consuming effort, which might be supported by better tooling (the WiSCE Confluence page appeared still rather "technical" for experts to directly create the domain model content).
3) An increased number of learning activities (i.e., tasks and accompanying materials, such as questions and carddecks) is necessary to maintain interest and engagement. In the current implementation, often, the number of possible tasks for selected goals was minimal and contributed to motivation loss. Recommender systems for online resources can be connected to find more content (see [33]).
4) Especially for children who do not consciously use the *cPOD*, it is important to have an intelligent task selection mechanism in place that guides the child toward the completion of those tasks that contribute to goal attainment.
5) Add to the user model an estimated knowledge level based on age and/or pretest results (and in the PAL case diabetes onset) to lessen the burden of initial goal setting. Or add a probabilistic model to estimate the child's knowledge based on other criteria such as interests and knowledge on other topics. This kind of model can be used for more fine-grained recommendations of the most appropriate next activity, for example, to watch an explainer video if estimated knowledge is low and play a quiz when estimated level of proficiency is good.
6) Expand the leaner model with domain-independent knowledge measures such as a learning aptitude test (see [34]).
7) Expand the diabetes knowledge base with cognitive goals on higher level's within Bloom's taxonomy and formalize the attitude goals stated to be important in diabetes self-management training but not yet modeled.
8) Add to the user model a buggy model (i.e., user's structural mistakes) to indicate possible misconceptions. Use these data to reactivate specific learning goals for retraining.

References

[1] D. H. Schunk and B. J. Zimmerman, *Motivation and Self-Regulated Learning: Theory, Research, and Applications*. Evanston, IL, USA: Routledge, 2012.

[2] C. Breazeal, K. Dautenhahn, and T. Kanda, "Social robotics," in *Springer Handbook of Robotics*. New York, NY, USA: Springer, 2016, pp. 1935–1972.

[3] J. F. Bryan and E. A. Locke, "Goal setting as a means of increasing motivation," *J. Appl. Psychol.*, vol. 51, no. 3, 1967, Art. no. 274.

[4] P. Black and D. Wiliam, "Assessment and classroom learning," *Assessment Educ.: Princ., Policy Pract.*, vol. 5, no. 1, pp. 7–74, 1998.

[5] R. Kleinrahm, F. Keller, K. Lutz, M. Kölch, and J. M. Fegert, "Assessing change in the behavior of children and adolescents in youth welfare institutions using goal attainment scaling," *Child. Adolesc. Psychiatry Ment. Health*, vol. 7, no. 1, pp. 1–11, 2013.

[6] P. Brusilovsky, "Domain modeling for personalized guidance," in *Design Recommendations for Intelligent Tutoring Systems: Domain Modeling*, vol. 4, R. A. Sottilare, A. C. Graesser, X. Hu, A. Olney, B. Nye, and A. M. Sinatra, Eds., Orlando, FL, USA: US Army Research Laboratory, 2016, ch. 15, pp. 165–2184.

[7] L. S. Vygotsky, *Mind in Society: The Development of Higher Psychological Processes*. Cambridge, MA, USA: Harvard Univ. Press, 1980.

[8] L. Berk and A. Winsler, "Vygotsky: His life and works and Vygotsky's approach to development," in *Scaffolding Children's Learn.: Vygotsky and Early Childhood Learn.*. Washington, DC, USA: NAEYC, 1995, pp. 25–34.

[9] B. R. Schadenberg, M. A. Neerincx, F. Cnossen, and R. Looije, "Personalising game difficulty to keep children motivated to play with a social robot: A Bayesian approach," *Cogn. Syst. Res.*, vol. 43, pp. 222–231, 2017.

[10] D. L. McGuinness and F. Van Harmelen, "OWL web ontology language overview," *W3C Recommendation*, vol. 10, no. 10, 2004, Art. no. 2004.

[11] M. Neerincx et al., "Socio-cognitive engineering of a robotic partner for child's diabetes self-management," *Front. Robot. AI*, vol. 6, 2019, Art. no. 118.

[12] S. Staab and R. Studer, *Handbook on Ontologies*. Berlin, Germany: Springer, 2010.

[13] S. Ebbens and S. Ettekoven, *Effectief Leren*. Groningen, The Netherlands: Noordhof Uitgevers BV, 2005.

[14] R. F. Mager, *Preparing Instructional Objectives*, 2nd ed. Belmont, CA, USA: Fearon Publishers, 1984.

[15] R. J. Marzano, D. Pickering, D. E. Arredondo, G. J. Blackburn, R. S. Brandt, and C. A. Moffett, *Dimensions of Learning*. Alexandria, VA, USA: Assoc. Supervision Curriculum Develop., 1992.

[16] B. S. Bloom, M. D. Engelhart, E. J. Furst, W. H. Hill, and D. R. Krathwohl, "*Taxonomy of Educ. Objectives: The Classification of Educ. Goals: Handbook I: Cogn. Domain*. New York, NY, USA: David McKay, 1956.

[17] L. S. Vygotsky, "Mind in society: The development of higher psychological processes," in *Interaction Between Learning and Development*. Cambridge, MA, USA: Harvard Univ. Press, 1980, ch. 6, pp. 79–91.

[18] M. E. Hoque, "Three domains of learning: Cognitive, affective and psychomotor," *J. EFL Educ. Res.*, vol. 2, no. 2, pp. 45–52, 2016.

[19] D. R. Krathwohl, "A revision of Bloom's taxonomy: An overview," *Theory Pract.*, vol. 41, no. 4, pp. 212–218, 2002.

[20] R. Dave, "Psychomotor levels in developing and writing objectives," 1970.

[21] E. J. Simpson, "The classification of educational objectives, psychomotor domain," 1966.

[22] D. R. Krathwohl, B. S. Bloom, and B. B. Masia, *Taxonomy of Educational Objectives, Handbook II: Affective Domain*. Philadelphia, PA, USA: David McKay Company, Inc., 1956.

[23] K. Stancin, P. Poscic, and D. Jaksic, "Ontologies in education—State of the art," *Educ. Inf. Technol.*, vol. 25, pp. 5301–5320, 2020.

[24] S. Wang, "Ontology of learning objects repository for pedagogical knowledge sharing," *Interdiscipl. J. E-Learn. Learn. Objects*, vol. 4, no. 1, pp. 1–12, 2008.

[25] T. Kasai, H. Yamaguchi, K. Nagano, and R. Mizoguchi, "Building an ontology of it education goals," *Int. J. Continuing Eng. Educ. Life Long Learn.*, vol. 16, nos. 1/2, pp. 1–17, 2006.

[26] S. M. Rashid and D. L. McGuinness, "Creating and using an education standards ontology to improve education," 2018.

[27] H. Chung and J. Kim, "An ontological approach for semantic modeling of curriculum and syllabus in higher education," *Int. J. Inf. Educ. Technol.*, vol. 6, no. 5, 2016, Art. no. 365.

[28] J. David, A. Lobov, and M. Lanz, "Attaining learning objectives by ontological reasoning using digital twins," *Procedia Manuf.*, vol. 31, pp. 349–355, 2019.

[29] E. L. Baker, G. Chung, and J. Herman, *Ontology-Based Educational Design: Seeing is Believing*. Los Angeles, CA, USA: Crest Publishers, 2009.

[30] L. Ng, "Ontological model for representation of learning objectives," Ph.D. dissertation, Dept. Interact. Arts Technol., Simon Fraser Univ., Burnaby, BC, Canada, 2005.

[31] A. Inaba, T. Supnithi, M. Ikeda, R. Mizoguchi, and J. Toyoda, "An overview of learning goal ontology," in *Proc. Workshop Anal. Model. Collaborative Learn. Interact.*, 2000, pp. 23–30.

[32] S. Assami, N. Daoudi, and R. Ajhoun, "A semantic recommendation system for learning personalization in massive open online courses," *Int. J. Recent Contributions Eng., Sci.*, vol. 8, 2020, Art. no. 71.

[33] P. Wongthongtham, K. Y. Chan, V. Potdar, B. Abu-Salih, S. Gaikwad, and P. Jain, "State-of-the-art ontology annotation for personalised teaching and learning and prospects for smart learning recommender based on multiple intelligence and fuzzy ontology," *Int. J. Fuzzy Syst.*, vol. 20, pp. 1357–1372, 2018.

[34] S. Sarwar, Z. U. Qayyum, R. García-Castro, M. Safyan, and R. F. Munir, "Ontology based E-learning framework: A personalized, adaptive and context aware model," *Multimedia Tools Appl.*, vol. 78, pp. 34745–34771, 2019.

[35] D. Freeborn, T. Dyches, S. O. Roper, and B. Mandleco, "Identifying challenges of living with type 1 diabetes: Child and youth perspectives," *J. Clin. Nurs.*, vol. 22, nos. 13/14, pp. 1890–1898, 2013.

[36] R. Peters, J. Broekens, and M. A. Neerincx, "Guidelines for tree-based collaborative goal setting," in *Proc. 22nd Int. Conf. Intell. User Interfaces*, 2017, pp. 401–405.

[37] M. V. Welie, "Task-based user interface design,"2001. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.90.3368&rep=rep1&type=pdf

[38] H.-U. Krieger, R. Peters, B. Kiefer, M. A. Van Bekkum, F. Kaptein, and M. A. Neerincx, "The federated ontology of the PAL project interfacing ontologies and integrating time-dependent data," in *Proc. 8th Int. Conf. Knowl. Discov., Knowl. Eng. Knowl. Manage.*, 2016, pp. 67–73.

[39] F. Kaptein et al., "A cloud-based robot system for long-term interaction: Principles, implementation, lessons-learned," *Trans. Human–Robot Interact.*, 2021, pp 1–27.

[40] R. Peters, E. Oleari, F. Sardu, and M. A. Neerincx, "Usability of the pal objectives dashboard for children's diabetes self-management education," in *Proc. 5th Int. Conf. E-Soc., E-Learn. E-Technol.*, 2019, pp. 22–28.

[41] M. M. Peeters, K. V. D. Bosch, M. A. Neerincx, and J.-J. C. Meyer, "An ontology for automated scenario–based training," *Int. J. Technol. Enhanced Learn.*, vol. 6, no. 3, pp. 195–211, 2014.

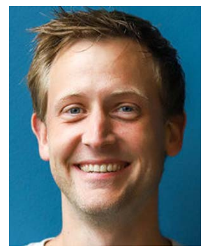
[42] B. Said, L. Cheniti-Belcadhi, and G. El Khayat, "An ontology for personalization in serious games for assessment," in *Proc. IEEE 2nd Int. Conf. Artif. Intell. Knowl. Eng.*, 2019, pp. 148–154.

Rifca Rijgersberg-Peters received the dual master's degrees in human computer interaction and science education from the University of Twente, Enschede, The Netherlands, in 2014 and 2018, respectively.

She is currently a Ph.D. Researcher of child–robot interaction with the Delft University of Technology, Delft, The Netherlands. She is also a Senior Lecturer of Interaction Design with the HAN University of Applied Sciences, Arnhem, The Netherlands. She works with the Personal Assistant on the Healthy Lifestyle Project on both the design of the robot's nonverbal behavior to express a strategic style and thereby support verbal behavior, and the development of children's learning goals that guide the robot in this style selection.

Willeke van Vught received the M.Sc. degree in artificial intelligence from Vrije Universiteit Amsterdam, Amsterdam, The Netherlands, in 2017.

She is currently a Research Scientist with the Department of Human Machine Teaming, TNO, Soesterberg, The Netherlands. Her research interests include observing, predicting, and directing team members from hybrid teams (containing humans and nonhuman intelligent agents) with the use of ontologies, models, reasoning, and by facilitating interactions within and between team members. As a Personal Assistant for the Healthy Lifestyle Project, she worked on the development of the children's learning goals, and facilitated workshops and experiments with health-care professionals, children, and parents.



Joost Broekens received the M.Sc. degree in computer science from the University of Delft, Delft, The Netherlands, in 2001, and the Ph.D. degree in computer science from the University of Leiden, Leiden, The Netherlands, in 2007, in the area of computational modeling of emotion in relation to learning processes.

He has authored or coauthored papers in the area of computational models of emotion (ranging from theoretical approaches to more applied ones), developed master-level courses and course material on the topic, and given several invited lectures as well as less formal talks for the larger public related to affective computing. His research interests include reinforcement learning, affective computing, human–robot and human–computer interaction, and gaming research.

Mark A. Neerincx received the M.Sc. degree in cognitive psychology from the University of Leiden, Leiden, The Netherlands, in 1997, and the Ph.D. degree in cognitive psychology from the University of Groningen, Groningen, The Netherlands.

He is a Full Professor of Human-Centered Computing with the Delft University of Technology, Delft, The Netherlands, and a Principal Scientist with TNO (Netherlands organization of applied research), Soesterberg, The Netherlands. His Ph.D. research was centered on task allocation and computer support for high-demanding cognitive work. In national and international research programs, he initiated and led the development and application of a sociocognitive engineering methodology for the design of artificial, virtual or physical, agents (ePartners) that show social, cognitive, and affective behaviors to enhance performance, resilience, health, and/or well-being. Examples include robotic and virtual assistants that help patients cope with their chronic disease (such as diabetics) in different self-managements activities, and robots that help older adults with dementia and caregivers to establish positive experiences of life in care centers.