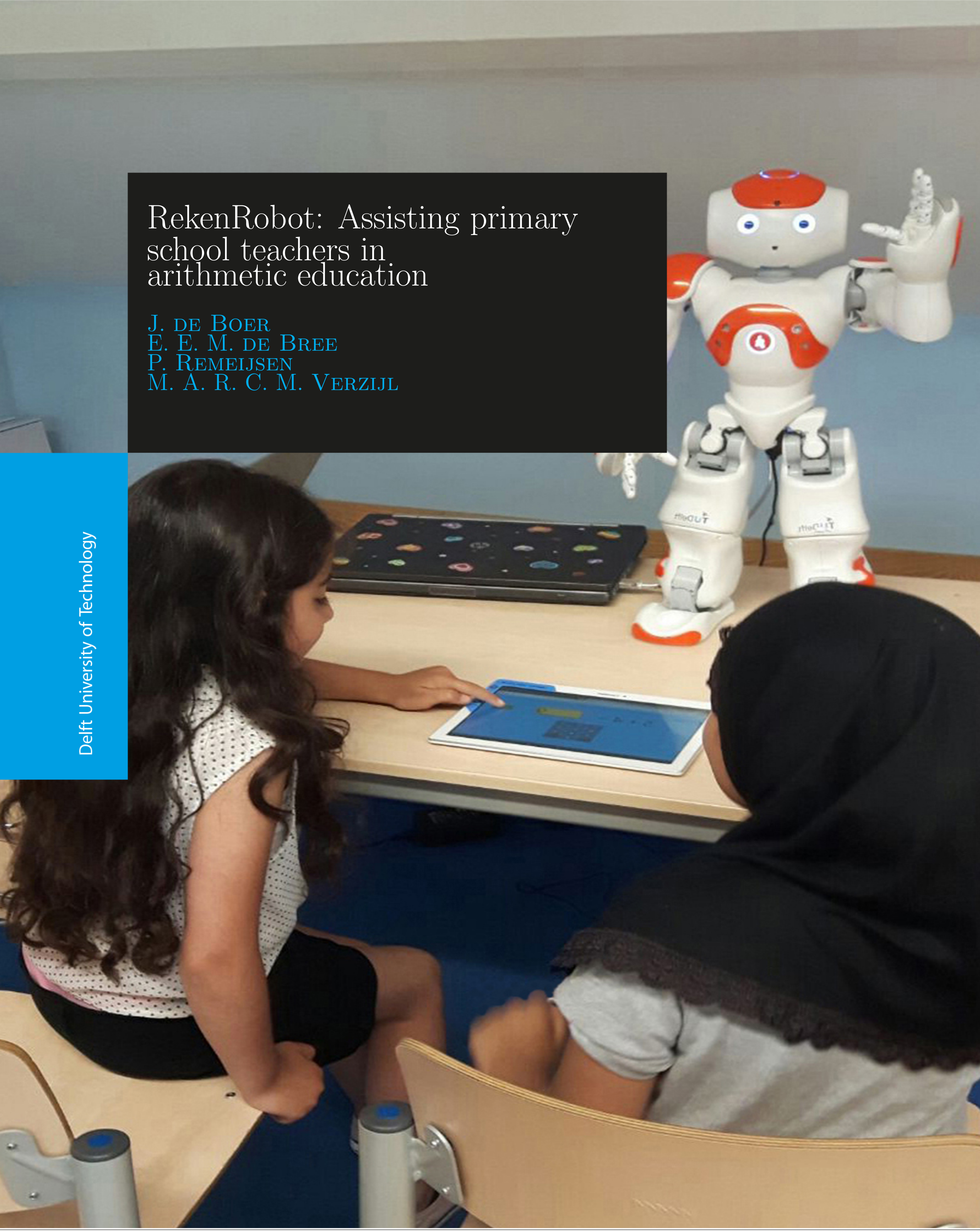


RekenRobot: Assisting primary school teachers in arithmetic education

J. de Boer
E. E. M. de Bree
P. Remeijsen
M. A. R. C. M. Verzijl

Delft University of Technology

Delft University of Technology

Challenge the future

Interactive Robotics©

RekenRobot: Assisting primary school teachers in arithmetic education

J. de Boer
E. E. M. de Bree
P. Remeijsen
M. A. R. C. M. Verzijl

Bachelor's Thesis
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology

TU Delft coach:
Dr. K. V. Hindriks
Company contact:
Dr. Ir. D. J. Broekens
Bachelor Project Coordinators:
Ir. O. W. Visser
H. Wang

July 14, 2017

This paper is under embargo for 6 months.

Foreword

This Thesis is the final report for the RekenRobot, written as part of our Bachelor Project for the course TI3806 at the Delft University of Technology. The RekenRobot was an assignment given by the Delft based company, Interactive Robotics©.

The aim of Interactive Robotics© is to bring robots into the classroom, with the RekenRobot being a robot that specifically helps with arithmetic. During the course of this report, we will describe the process which was undergone to develop and create the RekenRobot, from the research phase up to the final product.

We would like to thank Dr. Koen Hinkriks and Dr. Ir. Joost Broekens for their support and guidance with the RekenRobot, your ideas and suggestions were invaluable to us. We'd also like to thank Ruud de Jong for always being able to get us a robot when we needed one and for helping us get everything set up initially.

J. de Boer, E. de Bree, P. Remeijsen and M. Verzijl
June 2017

Interactive Robotics©

Summary

A large problem that primary schools face is that the ratio of pupils to teachers is too high, the class sizes are too large and this makes it difficult for a single teacher to have a good oversight of how the development of a given child is going. The aim of Interactive Robotics© is to tackle this problem by bringing robots into the classroom to aid teachers. They aim to have a single robot in a classroom that has the ability to teach different lessons and subjects; the RekenRobot being specifically for basic arithmetic. During the research phase, ideas were gathered regarding how to create teaching methods that are motivating and stimulating. For instance, personalisation, humanising the robot and adaptability of the teaching material were desired functions.

The software for the RekenRobot was built from scratch, using the programming language GOAL[1], JavaScript[2], CSS[3], HTML[4] and JSP[5]. The original target audience of the project were children between the ages of 6 and 8. Later this was changed to cover different school years: 3-4, 5-6 and 7-8, making use of levels with different degrees of difficulty. The robot can work one-on-one with a child, being able to practice addition, subtraction, times tables and telling time, as well as 2 forms of explanations can be given: making use of a bus and a number line. Using no explanation to rather focus on automation is also an option.

The idea of the project was to lay the groundwork for the later development of the Reken-Robot, as this will be an ongoing project for Interactive Robotics. The application designed in this project will be adapted to become part of the Interactive Robotics system.

The first user tests at primary schools yielded a largely positive result. The children were excited and motivated to work with the product. The system is simple enough to require very little explanation. This project was never meant to realise a product that can be deployed tomorrow, but the result is a very solid basis for further improvements.

Keywords: arithmetic, children, teaching, robot, robotics, primary school, teacher

[1] https://goalapl.atlassian.net/wiki/
[2] https://en.wikipedia.org/wiki/JavaScript
[3] https://en.wikipedia.org/wiki/Cascading_Style_Sheets
[4] https://en.wikipedia.org/wiki/HTML
[5] https://en.wikipedia.org/wiki/JavaServer_Pages

Contents

Chapter 1

Introduction

This paper will give an overview of the research and software development done over the course of ten weeks to create the 'RekenRobot'-project for Interactive Robotics©(IR). The purpose of this project is to develop a software solution using the NAO robots[1] and the IR platform to educate young children in arithmetic. The goal is to construct an educational system that is both exciting and engaging for the children, as well as easy to deploy for teachers without extensive technical knowledge.

IR is a company that seeks to use social robotics to enrich human experiences in for example education. IR makes use of multiple types of robots. For the purposes of this project, the NAO robot will be used. The NAO is one of the most used and recognised social robots available on the market. A humanoid shape, with a large range of movement capabilities as well as various sensing capabilities, makes NAO a versatile robot. NAO is capable of detecting touch on the hands and head and has physical buttons on the feet. The NAO can also recognise speech in 20 languages. Two small cameras are located on the front of the head. During this project, research will be done to find out which factors are important in this domain. Which features will have to be added to enable primary school teacher to improve the way they teach arithmetic, will also be of interest.

First in chapter 2, a problem definition and analysis will be given. Secondly the Research phase, found in chapter 3, will be discussed, during which papers were collected on the subject matter of robots in education, as well as for arithmetic education in general. Additionally, interviews were held with experts and the primary school teachers for whom this product was made. Next, an 'Ideal Robot' will be described in chapter 4, determined using the information gathered during the research phase. This 'Ideal Robot' description was then split into feasible user stories and the non-functional requirements, as explained in chapter 5. The process will be described in chapter 6. Chapter 7 will give some background information about the different levels of the Dutch educational system. The product design and implementation is discussed in chapter 8 and after this, there is a chapter on testing found in chapter 9. Chapter 10 contains the product evaluation, followed by the conclusion in chapter 11. Lastly, the discussion of the project is given in chapter 12, which also contains the ethical issues that could occur with the RekenRobot, the recommendations and any possible future research.

[1] `https://www.ald.softbankrobotics.com/en/cool-robots/nao`

Chapter 2

Problem definition and analysis

Dr. Koen Hindriks was the TU Delft coach for this project, while Dr. Ir. Joost Broekens was the company contact for the company Interactive Robotics©. The initial project description can be found in Appendix A.

The assignment given was to design and implement a robot that is able to assist teachers in the classroom when it comes to teaching children basis arithmetic; written in the programming language GOAL[1]. The idea behind the robot is not to teach the children new material, but instead help to revise the material that has already been learnt and to allow the children the chance to practice.

The original problem description did not go into a lot of detail, therefore there was a large amount of flexibility when it came to what was to be implemented. Therefore going into the research phase allowed for a large degree of freedom. The research phase is described in detail in chapter 3.

Before even the research phase had begun, the client gave the advice to image and write down what the 'Ideal Robot' would be, the result of which can be found in chapter 4. This could then be adjusted based on the outcomes of the research phase. The idea behind doing this was to give a 'point on the horizon', something to work towards, as well as allow for creativity in designing a robot that would be impossible to finish in the given time but would allow priorities to be set; deciding which functionalities are considered to be the most important.

The advice was also given to begin with clean code, only building the components that are needed for this project. In short, to use a bottom-up approach. Besides the standard actions and percepts of GOAL, it is possible to build everything else. Due to the fact that this product will eventually be part of IR system, it was very important to keep discussing and exchanging ideas, so that everyone, including the client and the TU Delft coach were all on the same page. The idea is that the product will not be linked to the system during the project, but instead afterwards.

Due to the fact that the product will later on be linked to the rest of the system, it was important to keep this in mind. All components had to be discussed with both the client and the TU Delft coach; the messages that were used had to be accepted by client as well as any ideas about how to link to the existing system in the future.

To give an idea of what the RekenRobot could become, the client believed that it would be very beneficial if there were at least three or four different type of tasks in the RekenRobot system. One of these should also be specifically for a older age group, for example year 8, to keep the project as generic as possible and also to keep in mind that the RekenRobot should not be completely based on one age group. In the end, the aim is for children of all ages at primary school to be able to work with the RekenRobot. To get a glimpse of what the future can hold for the RekenRobot. This also keeps the system from feeling too childish.

Seeing that the product will grow in the future to contain many different types of tasks and explanations, the system needs to be as generic as possible. The idea is that new methods will be easy to add to the system, preferably without having to change the basic system. Therefore it is also important to be able to determine from the project which types of tasks and explanation types are being used and where they originate from.

[1]`https://goalapl.atlassian.net/wiki/`

Chapter 3

Research

3.1 Introduction

As technology advances and becomes more common, the age at which children come in contact with these technologies keeps decreasing. It is not uncommon nowadays, to see children playing games on a smartphone or tablet before they are even able to walk. This means that also in education the use of electronics will start to play an even more important role. Robots have made huge leaps and bounds in previous years. There has already been a lot of research that has focused on the use of ICT and robotics to improve or enhance the learning process or experience of children.

One of the main aims of the RekenRobot is that it needs to be appealing to children, that they want to work with the NAO and that it does not become boring. The research phase was meant to determine what works well when it comes to robots in the classroom, and can keep the children motivated.

The research process will be described in the Methodology section 3.2. Here the process is described; how the research papers were found and how they were chosen. The aim was to find information about robots and education, in addition to trying to find out what the best techniques for teaching, specifically in the field of arithmetic for younger children. Research was done to give an idea of what is possible and what is considered important when it comes to robot-human interaction. The initial planning for the two research weeks was as follows:

- **Week 1**
  - The 'Ideal Robot'
  - Literature research
    - Educational side
      - Interviews with teachers / educational expert
      - Lesson methods research
      - Simulate and motivate pupils
    - Technological side
      - What already exists?
      - What works well?
  - Build a basic robot
    1. That can generate addition questions that have answer < 20
    2. That can wait for an answer
    3. That can react to the answer
       - Correct
       - Incorrect
    4. That can return to step 1.
  - Contact schools
  - Think about the link to the IR system that already exists
- **Week 2**
  - Interview teachers

    - How to keep the robot interesting for children
    - Gamification, a good idea?
    - ...
  - Continue work on the basic robot
  - Think about the functionalities that the RekenRobot could have
  - Write the user stories
  - Discuss components needed

IR already has pilot schools, some of which have a NAO, that were very willing to be interviewed and open to testing the RekenRobot with their children. These schools all saw robots in the classroom as being a very possible future and therefore were happy to help. They also see the advantages of having a robot in the classroom, the main problem many teachers face being that they feel they do not have enough time to help each child.

During the research, it quickly became apparent that there were two main aspects to consider when designing a robot that will be used to help children do arithmetic: the educational side and the technological side. The educational side's research tried to find out what the methods are that are being used in classrooms in the Netherlands, to teach young children, around the age of 6-12, simple arithmetic. It became apparent that researching the teaching methods would be very important, as the robot should be able to explain a question in a form that the children have already had before.

The technological aspect's research aim was to try and find out how to bring the material across to the pupils and the more practical side to teaching. This could be divided into three categories, what do children react well to when interacting with robots?, what are the advantages and disadvantages to using a robot over using a teacher?, and what is the NAO capable of? The first question's aim was to research the behavioural aspect of interactions with robots, aiming to find out what forms of teaching a pupil likes or dislikes. This section also tries to find out which methods are effective in bringing across information, as well as what a robot can do to help a pupil to recall information. The second research question, from the technical side of the research, aimed to uncover in what ways a robot can be useful in a classroom. With current technology it is impossible to fully replace a teacher with a robot, but the robot can be very helpful to a teacher. Why is the use of a robot in the classroom beneficial and in what ways is a teacher better? The last question is a description of what the NAO can do and why it is a good robot to work with children.

3.2 Methodology

To start the process of researching the topics, first the main keywords were defined: *children* and *robot*. In the case of children, only primary school children were of interest. This meant that the term *primary school* was included in the search queries. All papers about autism, disabilities, children with health problems or special needs were excluded as it was decided that, given the time frame, there was not enough time to add the features that would support these children. Children and primary schools are still very broad concepts, therefore *learning* and *calculating* were included in the search queries. The robot should help the children do arithmetic. This leads directly to the including criteria. The included papers should (1) present a robotics way of (2) teaching (3) especially for primary school children. All other papers were excluded. For example papers about mobile applications for preschoolers.

Papers were collected from three databases: Scopus[1], Web of Science[2] and IEEE Xplore[3]. For each of the databases, the search queries had to be adapted due to search constraints. The exact search queries can be found in Appendix B: Search queries. IEEE Xplore (Appendix A: IEEE XPlore Digital Library) has an advanced search features, which allows for long search queries. The search query for Scopus (Appendix A: Scopus) could be shorter as it quickly filtered down to fewer papers. Web of Science (Appendix A: Web of Science) did require long search queries, but has a filtering feature. To get the papers that were found, the 'Web of Science Categories'

[1] `www.scopus.com`

[2] `http://apps.webofknowledge.com/WOS_GeneralSearch_input.do?product=WOS&search_mode=GeneralSearch&SID=U2necp5zBag5dXz9jP2&preferencesSaved=`

[3] `http://ieeexplore.ieee.org/Xplore/dynhome.jsp?tag=1`

was set to robotics, for the 'Document Type', article, proceeding papers, editorial material and reviews were checked and the 'Research Areas' were to put to Robotics and Computer Science.

After combining the papers obtained from the three databases, there were a total of 785 papers. Additionally, two papers were received by mail, as these papers were not yet on the internet and two sample papers were given as a guideline as to what type of papers should be searched for. After removing the duplicates there were 505 remaining. These papers were then filtered based on the include and exclude criteria. The papers were all screened manually by reading the titles, keeping the including criteria in mind. If the title had the potential to be relevant, the paper was included. Any papers that were obviously not relevant were removed. After this round of filtering, there were 85 papers left. The abstracts of these 85 papers were read to determine whether the papers were relevant, resulting in 35 papers. The papers were divided in three different groups. The first group contained papers that were focussed on the behaviour of the robot towards children. The second group contained papers that described ways of teaching children with a robot. The last group contained papers that specifically described ICT-inspired ways of teaching arithmetic. Every group had a corresponding research question, which will be discussed later in this chapter. These remaining 35 papers were all read and papers were excluded if no relevant information could be obtained or if they contained overlapping information. The meant that 17 papers were used to answer the different research questions. The PRISMA Diagram of the research process can be found in Appendix C: PRISMA-diagram.

3.3 Educational aspect

3.3.1 Literature research

This section will take a look at how arithmetic is taught today. The goals of current educational programs will be covered and those chosen to be used in the RekenRobot will be described. The problem in current Dutch school system is that there is a lack of standardisation. The result is that there are many similar but all slightly different methods of teaching basic arithmetic. This paper will only look at a few aspect to give a picture of what the aims of the final product will be.

The goals and guidelines for education in the Netherlands are defined by the SLO (Dutch: Stichting Leerplan Ontwikkeling, English: Foundation for Curriculum Development). These guidelines will be kept in mind when developing the solution, as these are the common goals and practises that are required in Dutch education.

Therefore, the application will be designed to make sure the children can pass these requirements as fast and well, as possible. The initial target for this project is the third year of primary school, making use of the SLO guidelines for that age group. One of the first goals on the list is to give meaning to 'addition' and 'subtraction' through tangible examples. For example, explaining addition and subtraction by viewing it as people entering and exiting a bus. This way, children develop an intuitive meaning for both terms. Next children will have to be able to do addition under 20, while using visual number boxes and aids such as a number line. Children will also have to learn about the inverse nature of operations such as addition and subtraction. Next, the children are taught about 'language' of arithmetic, such as operators. One of the most important items on the list is giving the child strategies to make the arithmetic easier for themselves, such as:

- $3 + 6 = 6 + 3$ (switching)
- $6 + 5 = 5 + 5 + 1$ (almost the same)
- $4 + 6 = 5 + 5$ (transform)
- $5 + 8 = 5 + 10 - 2$ (compensate)
- $6 + 8 = 6 + 4 + 4$ (completing the 10)
- $12 - 6 = 6$ and $6 + 6 = 12$ (inverse)

Finally the children have to memorise additions and subtraction under 20 and be able to instantly answer such questions.

As an example of a Dutch arithmetic methods, 'WizWijs'(WW) will be looked at. In the WW-method the creators have chosen to make the entire method very visual-based. By using simple drawings with bright colors, they hope to excite the children and make it appealing for them. One of the preferred ways to comply with the first SLO guideline, is to use the metaphor of a bus. They ask the child: ‘When a stopped bus is occupied by 5 people and it drives away with 8, what happened?’ See figure 3.1 It gets the child to think of addition as the tangible concept of people entering a bus. Alternatively, they use easily identifiable concepts such as circles or hearts to do the ‘counting’. This could be seen as the first level of the method. After the children are more proficient at these task, the method builds towards more complicated exercises. For example using three buses instead of two, or combining addition and subtractions in one exercise. Additionally, the WW-method often makes use of games to present the material. For example, children are given images of two dartboards with two arrows in them each. They then have to add the values of the darts and determine which of the dartboard represents the highest value. This way it is exciting because they want to know who won the game.

There are also a lot of online solutions for practising arithmetic. Most use some form of ‘gamification’. These online games keep in mind that most children of this particular age group have very short attention spans. Therefore, at most 10 questions are asked in one game. Multiple levels of difficulty allow all children to be challenged on such platforms. Finally, being able to specify whether to practice addition, subtraction or both make these online platforms a very adaptable solution.

In current research it is generally considered a good idea to use some form of gamification. It often leads to a better ‘learning experience’ among children [1]. In the study, correct answers were indicated by a ‘positive’ expression, for example a certain movement. Wrong answers were indicated by a sound effect and extra explanation was immediately provided. The study indicated that this, in combination with gamification, had positive effects on pupils. Although gamification results in higher motivation, it does not always improve cognitive learning [2]. This illustrates that fun should not overshadow the actual learning. Another study showed that ‘fun, immersion, fantasy and sensation’ have the greatest impact on the continuous use and intent of children [3]. Besides the solution being easy to use for children, the software solution must also be easy for teachers without technical knowledge to deploy [4].

To make the robot more interesting to children, a narrative could be used. For example, the robot could have a problem and the child could help the robot solve that problem by doing arithmetic exercises. By using a narrative, the child is engaged and will be more willing to solve the exercises. By using the robot as a teaching-assistant, pupils could be more motivated than with human teachers [5].

One should keep in mind that a child being able to solve a problem in one scenario, does not automatically mean that the child can transfer the knowledge to another scenario. That is why variation in both exercises and narratives is preferred. Where ‘$2 + 2$’ is the arithmetic side of the sum and ‘2’ sweeties ‘+2’ sweeties is the ‘real life situation’. A young child will sooner understand a real life concept than symbols. That’s why it is important to demonstrate the connection between the symbolism [6].

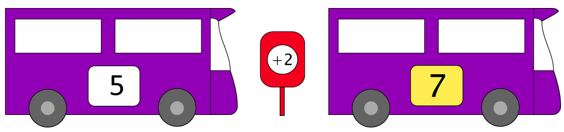


Figure 3.1: A representation of what a ‘bus’ exercise looks like. The yellow box is where the answer can be filled in on the tablet.

As well as the research done through papers and the Dutch educational system, interviews were also held, found in Appendix D. In total four interviews were held, the first with an educational expert (Appendix D.1) who had a larger range of knowledge about primary school education. The second interview was with a pilot school of IR called ‘De Voorsprong’ (Appendix D.2), the teachers who were interviewed being teachers who later on will be working with the RekenRobot. The third interview was with teacher from ‘De Gantel’ (Appendix D.3), who was asked very similar questions as ‘De Voorsprong’, to verify information that was learned before. The last interview was with a teacher from ‘De Notenbalk’ (Appendix D.4), who taught children of year 8. This interview was held to learn the teaching methods of children who are slightly older, to try and keep in mind that the RekenRobot needs to stay versatile.

3.4 Technological aspect

3.4.1 What do children react well to when interacting with robots?

When designing the interaction between the robot and the children, generalisations should be used that allow the interaction to be both motivating and stimulating to a child.

Clabaugh, Ragusa, Sha & Matarić (2015) [5] described "visual, verbal, and active"([5], 2015, para. 9) as the 3 main dimensions for interactions between robots and preschool children. They believe that keeping the interaction personal, is critical for the development of the child and to help them be able to reach their potential. They make a distinction between a simple robot that only determines if the child has answered a question correctly and a system that is able to determine differences in needs, styles of learning, has different levels of difficulty, etc. Playing is also a very important aspect when learning. Clabaugh et al. (2015) [5] used a Socially Assistive Robot (SAR) to work with the children, aiming for the SAR to be seen as a peer with a child-like voice, that can help the child to learn. The SAR system had 2 components, an Aldebaran NAO and a tablet. To be able to determine the child's learning styles a questionnaire was used, filled in by parents and teachers, to give an idea of which type of learning style works for each child, using the 3 above mentioned dimensions for learning. Exercises where then chosen based on these questionnaires. The exercises that were given ranged between 20 and 60 seconds, and also had a range of difficulty levels. During the sessions on average 24 questions were asked, for each question the child was given 3 chances to answer the question correctly, otherwise the next question would be asked. Each question had a different level of difficulty. There was a positive correlation between the correct answering of the questions and the time taken to answer them, though the response time was considered to be the more interesting factor as it could allow the system to predict whether the child will answer the question correctly. The main conclusion determined from the results showed that the personalisation of the SAR helped the children to learn, making use of the data gathered by observing the child, seeing what their learning styles are and their behaviour around the system.

In the research done by Kanda, Hirano, Eaton & Ishiguro (2003) [7], the main topic was the idea of identification of the child. A wireless sensor was given to each of the children, allowing them to be identified. Two robots were used, and the children could freely interact with them. Initially the robots gained a large amount of attention, where most of the 119 pupils with which the robots were tested wanted to interact with the robots. After a week interest in the robots started dwindling and quickly there were times when no one wanted to play with the robots. This is important to keep in mind; the robots have to stay interesting to the children for them to want to continue working with them. What was also discovered while testing with the robots was that children started developing sympathetic emotions for the robot when they saw that they were alone, this is believed to be a first step in forming a longer lasting bond with the robots. There was also an instance where a child initially did not have any interest in the robots, but when it called her name she started interacting with it. This shows that by using names, therefore by giving the child the idea that the robot knows them, can make the robot seem more interesting to a child and therefore increases the chance that they will want to interact with the robot. In general, the use of names is considered a good way of engaging a child when working with a robot; it gives a more personal touch to the interaction. Using the wrong name when interacting with a child will have a negative impact, same as in human interact. Another interesting point found was during a situation where one child saw another child playing with the robot, and therefore also wanted to join in. This means that a child playing with a robot could help to increase the interest for the other children, either through curiousness or jealousy. If the robot is considered to be interesting, then it may bridge the gap to using the robot for educational learning. When learning, pupils form a bond with their teachers, to get this same idea with robots ways need to be determined to be able to form a similar bond. A big problem that came out of this study was that long term interest in the robot is hard to achieve, therefore new ways of engaging a child need to be created.

The robot's ability to recognise emotions is also considered to be important according to Ahmad, Mubin & Orlando (2016) [8]. They argue that perception of emotions can help to increase the effectiveness of the sessions with a child while they are learning. They say that it is very important to create fitting responses to what is happening during the interactions, though this is considered challenging to do. Being able to interact in a human-like way is difficult to achieve for a robot. Robots have to be able to respond in a natural way. Ahmad et al. (2016) [8] also made use of a NAO, having it play 'Snakes and Ladders'[4] with 12 children. The NAO checked the child's facial expressions every 30 seconds, while also checking how the child is doing in the game. Based on the emotions sensed by the NAO, its speech and gestures where determined accordingly. Memory of the previous sessions was key to some of the phases used; by referencing to a previous game the robot can also come over as more human-like as it is remembering what happened in the past, which is a step towards building a relationship with the child. One of the main requirements that came out of the research was that the robot needs to be able to adapt its voice based on the situation that it is in. The tone and pitch of the robot's voice should be adaptable, while interacting with a child, to give the conversation a more real feeling. Overall in this study, it was determined that children liked it when the robot changed its reactions based on the changes in the child's emotions, made use of the previous interactions and adapted based on personality.

Also in the study done by Ahmad et al. (2016) [8], the children were asked what their opinions were when it came to NAO and the use of robots in the classroom and some very useful information was obtained. The children who had tested the robots considered sustainability to be an issue and also said that they believed the teacher still to be a very important part of the classroom. The children had different views when it comes to robots, some believing that a robot has the ability to crash and therefore should not be used in the classroom. This could also lead to children feeling frightened of the robot, especially if the robot starts doing unexpected things or performing unexpected actions. When it came to the actual use of a robot, children wanted the NAO to help but not reveal an answer straight away, instead give hints as to how to reach an answer. All the children believed gestures to be a very important part of teaching, and would like the robot to depict emotions using gestures. They also preferred it when the robot had the ability to remember previous interaction, to help build a history with them. Some children even believed that to make a robot more natural, it should remember some events incorrectly, make mistakes and should sometimes be corrected by the child. Others believed this to be false and that a robot should always remember everything correctly, as any machine should. Another point that the children made was that the robot should understand emotions and be able to describe their own emotions to the child, allowing the child to interact with the robot on a deeper level. Of course the robot does not actually have emotions, but instead should show the socially correct emotions to, for example, losing a game. There should be a large range of emotions, including confidence as some believe that if the robot appears to be confident, they too will feel more confident. The emotion of sadness should only be used when the child themselves is, for example, hurt. The main emotion to be used should be happiness though. There was a divide as to whether the robot should adapt its personality per child to a similar personality or to an opposite personality. Some believed that a shy child could benefit from a confident robot. The robot should be able to identify what type of personality the child has, and know what a child likes and does not like. According to the children the NAO should speak slowly in a classroom, and adapt its pitch based on the situation it is in. A concern for the safety of the robot, for example against viruses, was also made therefore this should be taken into account. The teacher should also be in the classroom when a child is interacting with the robot, to keep an eye on what is happening and be able to prevent to child destroying or running away with the robot. A last point that the children made was that the robot should update its dialogue and actions, as otherwise the interaction can become boring.

When it came to gender when interacting with robots, Sandygulova, Dragone & O'Hare (2014) [9] believe that for children up to the age of 7 or 8, the gender stereotypes are in general quite applicable. After this, these stereotypes start to break down. Therefore for younger age groups the robot could make a distinction for different genders. In this research it also came to light that children up to the age of 10 preferred voices that sounds the same age as they themselves are, while after this age the children tended to prefer a female adult voice.

Huang & Mutlu (2014) [10] found that gestures are especially important when teaching. They help to increase the user's experience and improve the human-robot interaction. Gestures that call attention to an object help a person to recall information, as it is considered the best type of gesture to help a person learn. The use of beats help a person to understand what the robot is trying to explain. Depicting objects can make a robot's communication seem more natural, and also help a person recall information. Huang & Mutlu (2014) [10] believed that gestures that give

[4] `https://en.wikipedia.org/wiki/Snakes_and_Ladders`

an idea of a moment in time, with for example forwards and backwards movements for future and past, was in general not well received as the test subjects believed them to be distracting. This was due to the large amount to movement that was used, though these movements did contribute to higher ability to recall what the robot had said. Therefore, gestures can be seen as an important part of teaching, and also allow the robot to come over as more human-like. Overall a mix of different types of gestures is advised, the type being determined by the context and content that the robot is teaching.

Lin, Liu, Chang & Yeh (2009) [11] did a study in Taiwan on a child's perceptions of robots. In the study, 26.5% of the children believed that robots are there to help them and do things for them, while another 26.5% believed that robots should be there to be their companions. For many of the children, their expectations of what a robot can do were far higher. 66% of the children thought that robots were interesting, but believed them to be able to do things that are not yet possible. Some believed that robots are able to transform and are much smarter than humans. Therefore, before children are introduced to robots, the limitations should be explained in an attempt to lower expectations. If this is not done, there could be a negative impact and disappointment when a child is working with a robot. What also came out of this study was that children have different opinions as to which subjects they would like a robot to teach them, the largest percentage (26%) was for the physics subject Nature Science, as some believed it to be the hardest subject. From this research, 59% of the children would like to learn from a robot. The main reason for wanting a robot tutor would be to ask the robot to repeat content already learned, either due to the child missing the content or as they did not understand the teacher's initial explanation. The main reason for not wanting to have a robot tutor would be that it could be distracting. Some children also said that they prefer learning with a human teacher. Therefore teacher should be careful when they introduce a robot to a classroom, and conscious of when the best times are to do so. It must also be kept in mind that some of the children do not like robots, and will be less inclined to interact with them, though overall there is a positive outlook from children when it comes to robots.

3.4.2 What are the advantages and disadvantages of using a robot over a teacher?

Robots offer new ways of teaching and present the possibility to relieve the workload of teachers and supplement the existing curriculum. One of the consistent themes among the papers found for this research question was that robots are not a replacement for human teachers. Not only because the robots are not advanced enough to take over all responsibilities of the human teachers, but also because robots and humans can fulfil different and complementary roles. For example, one of the papers talked about having the NAO not as a teacher, but as a peer who is also 'learning' [12]. In the papers that were found on the subject, there was little to no information about using robots for education in relation to using human teachers, for example tasks that a robot cannot do but a teacher can and vice versa.

One of the challenges that robots could help with is that nowadays classes are of a considerable size, leading to problems for teachers. Ideally, teachers would like to give each of the pupils adequate amounts of attention. With large groups, teachers have to divide their attention and are hindered by the fact that large groups are harder to control and have problems focusing on the material [13]. Technological solutions might help solving such problems. One of the possibilities is splitting off a small group of the pupils to work with the robot, while the teacher is busy with the rest of the group. Another option is having the robot single out one pupil who is either lagging behind or is exceeding the rest of the class. Robots in education are often not the one-size-fits-all solution that the human teachers are. However, human teachers have difficulties differentiating pupils and teaching them at different paces [14]. Robots and other technological solutions might help in overcoming such difficulties. Human teachers have to keep track of the progress of many children. By letting the robot do such tasks, the human teacher can easily differentiate between children and tailor the education to their needs. A group of advanced children could be challenged by the robot, while the human teacher focusses on the struggling children. In short, many configurations are possible to help relieve the human teacher.

When children interact with robots instead of with human teachers, a lot of excitement about learning can be generated and stimulated. Something that keeps coming forward are the positive effects of combining teaching with physical exercise. For example, using a mat with numbers on it, on which children jump to select the correct answer when practising multiplication tables [15]. For education to be effective, the robot should also be able to facilitate 'natural' interaction. However, one of the main problems with robots in education is the cost this requirement introduces. Using robot-based education could provide an ability to customise education to certain children who need supplementary teaching next to the teacher. Besides the cost of using robots in education, the technological knowledge that teachers are required to have to use such a robot is still a hurdle to large scale adaptation of the technology. Ideally, the human teacher does not have to acquire programming or other technical skills to operate educational robots [16]. The first solutions that will find large scale adoption will undoubtedly be plug-and-play systems.

3.4.3 What is the NAO capable of?

The research mentioned in the subsections above gives a good insight into the way children interact with robots. However, this project will be making use of the Aldebaran NAO robot.

The NAO is connected to the network and has the ability to talk to the child. It is 58cm high, and has a few human attributes; a head, arms, legs, a body, feet and each hand has 3 fingers. The NAO has speak recognition using a microphone and speakers where its voice comes out of. The NAO also has the ability to move its head, body, arms and legs, which allow the robot to have a more human-like appearance.

As with all robots, a robot has its own capabilities and limitations. When combined with this project, the NAO has a few advantages, which include; portability - teachers can easily move it and put it in the correct position, and a smaller size - pupils find it easier to connect with a robot which is the same or small in height. The NAO also has a good range of motion which can be used to perform versatile movements. Besides the user input by voice or what the robot sees, the NAO also has a few touch sensors on its body. These are found on its hands and on its feet, the latter in the form of bumpers which can be pressed.

There are also some disadvantages when using the NAO; physical interaction with children is important but the NAO cannot really move swiftly around on its own, meaning that the interaction is limited to a specific space. Due to the anatomy of its hands it cannot really pick up or move items around. No proper facial recognition, limited battery power and engines which can overheat (when fixed in a position) are on the list as well. Some of these disadvantages, such as the battery power and the lack of moving around, did not have to be a problem for this project because of the way the robot is used. The NAO is put on a table in front of a pupil, therefore it does not walk around, meaning that the NAO can stay plugged into a power outlet. Another problem that was encountered was that the voice recognition was not always accurate, the NAO cannot pick up every voice type, some it struggles with and therefore does not always 'hear' the correct answer, even when it has been spoken and occasionally does pick up the correct answer even when the incorrect one was said.

Looking at these these capabilities and limitations, it quickly became clear that using only the NAO for the RekenRobot would be difficult. It was decided, also based on the research, to include a tablet alongside the NAO. This tablet can be used as an extra visual stimulant to explain certain questions, as well as a device for the pupils to input the answer to a question.

Chapter 4

The Ideal Robot

In this chapter, an indication is given of what arguably the 'ideal robot' would look like. This 'ideal robot' has been design based on both initial ideas and ideas that were gathered during the research phase. The reality is that this project spans only ten weeks during which the software solution has to be developed. Therefore, not every desirable feature will fit in the scope of this project. The purpose of this section is to set a goal and give initial shape to the project.

The ideal robot should of course be able to do arithmetic exercises with the children. This will be possible through both speech and using a tablet. Random numbers should be generated according to the parameters set by the teacher. Additionally, the teacher should be able to set and adjust the robot to different levels of difficulty. It is essential that a reliable and clear interface should be available to the teacher, where they can change settings, add new children and receive feedback about how the children are doing. The progress of the pupils should be recorded in a database, such as the amount of question already asked and how long the interactions are. Ideally, the robot should be able to detect when the pupil walks away. The login system for teachers as well as the pupils should be as easy and fast as possible. For example, the pupils could enter a short number code on a tablet by the robot, after the teacher generates that code specifically for that child. The same could be applied to group exercises. The robot should be ready to go immediately after logging in and the teachers should be able to pause the interactions. The entire system should be designed in such a way that new modules can easily be added and modified. The addition of new types of questions or teaching methods should require little effort. If possible, the teacher should be able to pause the lesson.

One of the biggest selling points of the robot is that it helps to excite and retain the attention of the children. To accomplish this the robot should be able to display positivity through acts such as dancing or gestures, after a certain number of correct answers. The questions and answers should be dynamic and not be noticeably repetitive. Additionally, the robot should be able to make the interaction personal, for example by using the name of the pupil. To avoid frustration in the interaction, the robot should adjust difficulty based on how the pupil answered the previous questions, as well as provide explanation when wrong answers are being given. The same goes for complimenting the pupils when they are doing well. In the same spirit, the robot could measure how long it takes the pupils to answer an answer, to determine how difficult it is. The difficulty of a question could be modified by changing the time the pupil has to answer. Asking some of the same questions as in previous interactions gives an ability to gauge whether a pupil remembered the material. The robot should help explain the tasks through narrative or contextualising, but also help visualise it by using blocks, tablet, QR-codes etc. Tangible examples, for instance, people entering and leaving a bus could be used by the robot to make addition and subtraction simpler. Besides these requirements there are additional features that are desirable:

- Various forms of gamification, for example a quiz.
- Group exercises.
- Learning that involve physical exercise.
- Answering through touch or the bumpers.
- Adapting for special needs children.
- Free-choice period where the pupil can choose to keep on working, or do something different before returning to the robot. [17]

- Displaying different emotions and depicting numbers with its body.
- Class presentations (or robot and child presenting what they have learned together).
- The robot should detect that someone is in front of the robot, and is able to detect when the child walks away. Ideally even recognise when another child answers a question out of turn.
- The ICT department should be able to deal with potential problems without too many issues.

The ideal location for the robot would be in a separate space, away from other children so neither the child working with the robot can be distracted by the other children or the other children can be distracting for the child working with the robot. But at the same time the ideal location should also be somewhere where the teacher can keep an eye on what is happening with the child working with the robot. These two desired requirements can in some cases be contradictory, therefore the actual location will need to be discussed per classroom and per school as to where the teachers believe the best place is to have the NAO. Whether the robot is placed in a classroom or in a separate room, the location should be at an appropriate height and provide easy access to electricity and internet connection.

Chapter 5

Functionalities

Based on the chapter 4: The Ideal Robot, the user stories could be created, based on what was considered to be feasible. Parts of the ideal robot will not be possible, either due to the constraints of the NAO or due to the time constraints.

5.1 User stories

The User stories will be given in the form of a MoSCoW method[1]. With the MoSCoW method, the user stories are divided in four different categories: 'Must have', 'Should have', 'Could have' and 'Won't have'. The 'Must have' requirements of the product are requirements that are critical for the product, without the 'Must have' criteria the product cannot be successful. The 'Should have' requirements are considered important and should be in the final product, but are not critical. The 'Could have' requirements are nice to have in the final product, but will only be done if there is time. The 'Won't have' requirements will not be done. The user stories give the functional requirements of the project.

- *Must Have:*
  - The robot must be able to create and verbalise simple arithmetic problems and ask a pupil for an answer.
  - The pupil must be able to answer questions given by the robot using speech.
  - The pupil must be able to answer questions given by the robot using a tablet. A web-based program must be created, with the option to show the arithmetic problem and the pupil must be able to input an answer.
  - The robot must give feedback to the answer that the pupil has given:
    - If the answer is correct, give the child a compliment.
    - If the answer is wrong, allow the pupil to try again if they got it wrong. The robot must be able to keep a track of the number of wrong answers a pupil has given to the same question, after 2 wrong answers the robot helps the child to work out the correct answer.
  - The robot must give a response to an answer, making use of multiple different text phrases, sounds and gestures; with difference between correct and not correct answers.
  - The teacher must be able to give the robot a task. Therefore an interface must be made, in which the teacher chooses the task (addition, subtraction, etc).
  - The robot must keep track of how long a pupil takes to answer a question.
  - The teacher must be able to pause or stop the robot with use of the interface.
- *Should Have:*
  - The robot should keep track of how long it has been interacting with a pupil and how many questions it has asked.
  - The teacher should be able to tell the robot which level it should use for a given pupil by using the interface.

[1] `https://en.wikipedia.org/wiki/MoSCoW_method`

- The robot should be able to personalise. The teacher should set the name of the pupil in the teacher interface and the robot will use that name.
- The robot should be able to recognise if a pupil is finding the questions easy or difficult and adapt the level accordingly.
- The robot should be able to explain a question with the use of visualisation or a tablet.
- The tablet should be able to visualise the question that the robot asks, for example by using: sweets, buses, animals etc.
- The pupil should be able to log in on the tablet.
- The robot should have active / physical way of teaching the pupil.
- The robot should be able to keep track of how many questions a child has gotten correct in a row, and if with for example 5 correct questions in a row give the child a high-five or dance.

- *Could Have:*
  - The robot could have the ability to ask questions in a fun and interesting way, making use of gamification. Accordingly to section 3.3.1, for example using a narrative or using dartboard to decide who is the winner.
  - The robot could be able to adapt the time given to a pupil to answer the question asked.
  - The robot could ask questions from previous sessions to see if the child has remembered what they have learned and adapt the questions asked based on the results.
  - The robot could change its emotions based on whether the correct answer has been given or the actions of the pupil.
  - The pupil could answer questions given by the robot using the bumpers found on the robot.
  - The pupil could answer questions given by the robot using physical activity, for example running to a correct point in a room.
  - The pupil could answer questions given by the robot using cards.
  - The database could keep a track of data per pupils, teachers and schools.
  - The robot could recognise that the child that it is teaching is nearby.
  - The robot could have a quiz function.
  - The robot could work with multiple children at the same time and be able to distinguish which child has answered a question.

- *Won't Have:*
  - The robot gives the pupil the option to have a free-choice period where they can choose to continue to work with the robot or do something else before going back to the robot.
  - The robot has the ability to differentiate between pupils with disabilities or special needs and teaches them accordingly.
  - The pupil has the ability to answer questions given by the robot using blocks.
  - The robot has the ability to depict objects or numbers with its body.

5.2 Non-functional requirements

The non-functional requirements are the requirements of the system itself, instead of the behaviour and functions of the system. Many of the non-functional requirements will be explained in further detail in chapter 6. The non-functional requirements for this project are as follows:

- The back-end code will be written in the programming language GOAL[2].
- The front-end will be written in JavaScript[3], JSP[4], and CSS[5].
- GitLab[6] will be used to keep a track of branches.
- The NAO robot will be used to communicate with the children.

[2]https://en.wikipedia.org/wiki/GOAL_agent_programming_language
[3]https://en.wikipedia.org/wiki/JavaScript
[4]https://en.wikipedia.org/wiki/JavaServer_Pages
[5]https://en.wikipedia.org/wiki/Cascading_Style_Sheets
[6]https://about.gitlab.com/

- A tablet will be used as a second way of submitting answers.
- Scrum[7] will be used to organise the project.

Besides the technical requirements that were provided by the clients, there were additional requirements that the system should fulfil. Because the application is going to be used by primary school teachers and young children, the user interface should be intuitive and no technical knowledge should be required. The system should also be pleasant to use. If there are huge delays in the interactions, the user will get frustrated. Another requirement is that the system should be designed in such a way, that it can be easily adapted for and deployed on the IR platform.

[7] https://en.wikipedia.org/wiki/Scrum_(software_development)

Chapter 6

Process

This section will give an overview of the process, including the project method, the planning and development tools that were used and the initial roadmap of the project.

6.1 Project method

At the start of the project, a room in close proximity to the office of the company contacts was assigned to the group in the EWI building. The use of this room was very useful as it allowed the project members to sit and work together, being able to physically discuss problems with ease and ask questions without convoluted means. A confidentiality contract was signed, meaning that the code written belongs to IR.

To keep track of the progress and to be able to plan what needs to be done when, the Scrum[1] approach was used. The main idea behind Scrum is that a project is split into sprints, in this project the sprints were a week long. Tasks are determined and assigned to team members, and the aim is to have these tasks finished within that sprint. Each task is given a priority, meaning that the tasks that have a high priority are expected to be finished first, while those with a lower priority will be done later in the week, but are in theory also expected to be done. JIRA was used to keep track of the sprints, JIRA[2] will be explained in more depth in Section 6.2. The Scrum Master was J. de Boer, while the Product Owner was M. Verzijl. Each week the role of secretary rotated among the team members.

Every day a meeting was held, called a short daily Scrum meeting, to see what task are being done that day. On the Monday of the following week (to allow for work to be done during the weekend), a Scrum meeting was held to review the week. A retrospective was written with everything that had gone well during the week, and with things that could have gone better. Then new tasks for the new week were decided and divided, following which the new sprint was started. If any tasks were not finished during the previous week, they are shifted to the new sprint.

The amount of workload was also adapted slightly every week, to try and find a situation where all the members are given enough work, but not so much that they are not able to finish their tasks for that week. Of course there are always tasks that are expected to take longer or shorter than originally planned. If a team member finished their work too early, then it was always possible to look at the backlog of tasks to find other work to do, keeping the roadmap in mind.

A task was considered to be finished, and therefore a merge request could only be accepted, if another member of the team has reviewed the code. As a general rule, no one was allowed to merge their own merge request.

6.2 Planning, development tools and frameworks

IR already uses a lot of tools to assist with project development, therefore the same tools were used in the development of the RekenRobot. The libraries chosen for the web front end were

[1]https://en.wikipedia.org/wiki/Scrum_(software_development)
[2]https://www.atlassian.com/software/jira

mainly chosen based on what was needed for the user interfaces.

JIRA is a software development tool that can keep track of sprints. JIRA allows for a backlog of all the tasks that still need to be done, and an active sprint to be shown. JIRA's sprints allows for different types of issues - stories, tasks and bugs - to be assigned.

Confluence[3] is a piece of software that is specifically made to help software engineers keep a track of their projects. Confluence also has a work space where all documentation can be written. This is where all the meeting notes, product requirements, the calendar, architecture design and flow diagrams could be found.

SourceTree[4] is a software that makes use of Git, allowing code to be easily pushed and pulled. It clearly shows what code has been added, removed and changed.

Gitlab[5] was used to keep a track of the branches, and this is where merge requests were created, and could easily be adapted if need be.

Maven[6]is a package and dependency management tool.

JQuery[7] is a JavaScript library that makes, for example event handling easier.

Bootstrap[8] is a web development framework for HTML, CSS and JavaScript.

d3js[9] is a components library for HTML and CSS. The teacher interface makes use of the d3js library.

Socket.io[10] is a used for the sending and receiving of messages.

W3Schools[11] is a site that helps web developers, making use of tutorials, examples and has descriptions of functions. The clocks used in the clock questions have as basis the clock code that comes from W3Schools.

QUnit[12] is a testing framework used for testing JavaScript code.

Apache Tomcat[13] is a web-container which was used to develop the JSP-pages.

6.3 Roadmap

The first two weeks of the project were the research weeks. Based on these two weeks a better version of the roadmap could be made. The roadmap components are all stories that during the sprint were divided into tasks. Starting the project, the aim was to complete all the interviews during the research weeks, but due to the regional 'May holidays' that fell exactly in those two weeks, there were no reactions to emails until the 3rd week. This meant that interviews were also done during the work phase of the project. The planning can be found in Table 6.1.

[3] https://www.atlassian.com/software/confluence
[4] https://www.atlassian.com/software/sourcetree
[5] https://about.gitlab.com/
[6] https://maven.apache.org/
[7] https://jquery.com/
[8] http://getbootstrap.com/
[9] https://d3js.org/
[10] https://socket.io/
[11] https://www.w3schools.com/
[12] https://qunitjs.com/
[13] http://tomcat.apache.org/

Week	Tablet	NAO	Teacher Interface	Formulation	Paper	Extra Information
Week 3 (8^{th} May - 14^{th} May)	• Have the ability to answer a question given by the NAO	• NAO's reactions • Give more questions	• Choose a task, using parameters	• NAO should give feedback		
Week 4 (15^{th} May - 21^{st} May)	• Login system for a child	• Register time taken for a child to answer question • NAO can give a reaction if a series of questions is correct	• Can pause the interaction with the NAO	• NAO is active and physical		
Week 5 (22^{nd} May - 28^{th} May)	• Visualisation of the questions	• NAO can keep a track of sessions	• NAO personalisation • Implement different levels	• NAO differentiation • Explanation visualisation		
Week 6 (29^{th} May - 4^{th} June)	• Testing with children • Update based on feedback • Possible implementation of could have's	• Testing with children • Update based on feedback • Possible implementation of could have's	• Testing with children • Update based on feedback • Possible implementation of could have's	• Testing with children • Update based on feedback • Possible implementation of could have's	• Write the Product Design section	• Deadline 1st SIG on the 2nd June
Week 7 (5^{th} June - 11^{th} June)	• Testing with children • Update based on feedback • Possible implementation of could have's	• Testing with children • Update based on feedback • Possible implementation of could have's	• Testing with children • Update based on feedback • Possible implementation of could have's	• Testing with children • Update based on feedback • Possible implementation of could have's	• Write the Product Design section	
Week 8 (12^{th} June - 18^{th} June)	• Testing with children • Update based on feedback • Possible implementation of could have's	• Testing with children • Update based on feedback • Possible implementation of could have's	• Testing with children • Update based on feedback • Possible implementation of could have's	• Testing with children • Update based on feedback • Possible implementation of could have's	• Write the Product Design section	
Week 9 (19^{th} June - 25^{th} June)	• Testing with children • Update based on feedback • Possible implementation of could have's	• Testing with children • Update based on feedback • Possible implementation of could have's	• Testing with children • Update based on feedback • Possible implementation of could have's	• Testing with children • Update based on feedback • Possible implementation of could have's	• Write the Conclusion • Write the Discussion • Write the Abstract • Write the Foreword	• Deadline 2nd SIG on the 26^{th} June • Deadline Report on the 26^{th} June • Deadline Infosheet on the 26^{th} June

Table 6.1: The plan of action for the RekenRobot, for the work phase of the project.

Chapter 7

Background information

As discussed in the section about education research, there are many different methods of explaining a question. This project focused mainly on the methods used at the schools in The Hague and which where mentioned by SLO.

In this chapter, a description of each of the tasks used for the RekenRobot is given. These descriptions include what a child of a certain age group should be able to do given the task, as well as the ontology of the RekenRobot that accompanies the task. A generic ontology can be found in Figure 7.1.

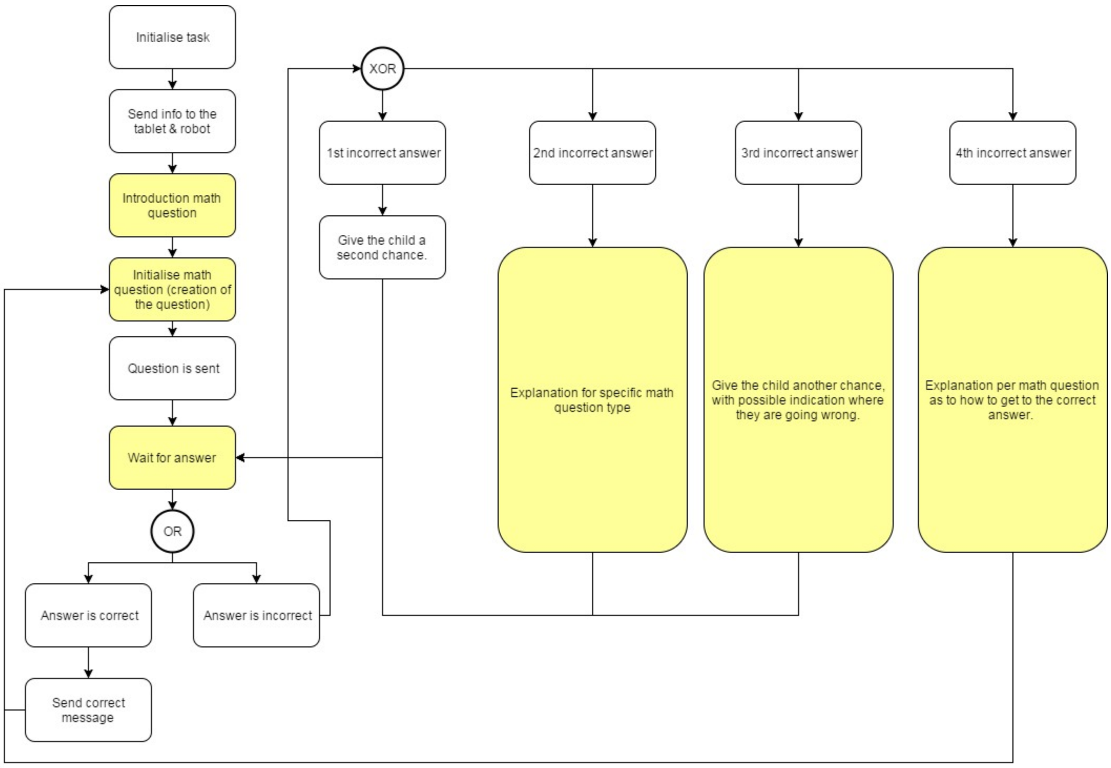


Figure 7.1: The generic GOAL agent flow diagram, the yellow boxes change based on the specific task.

7.1 Plus and Minus

For plus and minus questions, using the SLO core objective 27, the following holds per age group:

- Year 3/4
  - Addition and subtraction up to 10 and up to 20 using number images (five and ten structure), using help methods such as a number line.

- Automate and memorise additions and subtractions up to 10 and up to 20.
  - Addition and subtraction up to 100, making use of a number line and the idea of abacus to help them count.
  - The formation of a network of number relations when adding and subtracting up to 100.
- Year 5/6
  - Practise and application of the memorised knowledge of addition and subtraction up to 20.
  - Practise and application of the doing addition and subtraction up to 100 mentally.
- Year 7/8
  - Practise and application of the arithmetic addition and subtraction up to 100 mentally.

The specific ontology flow figures for plus questions and minus questions can be found in Figure 7.2 and Figure 7.3 respectively.

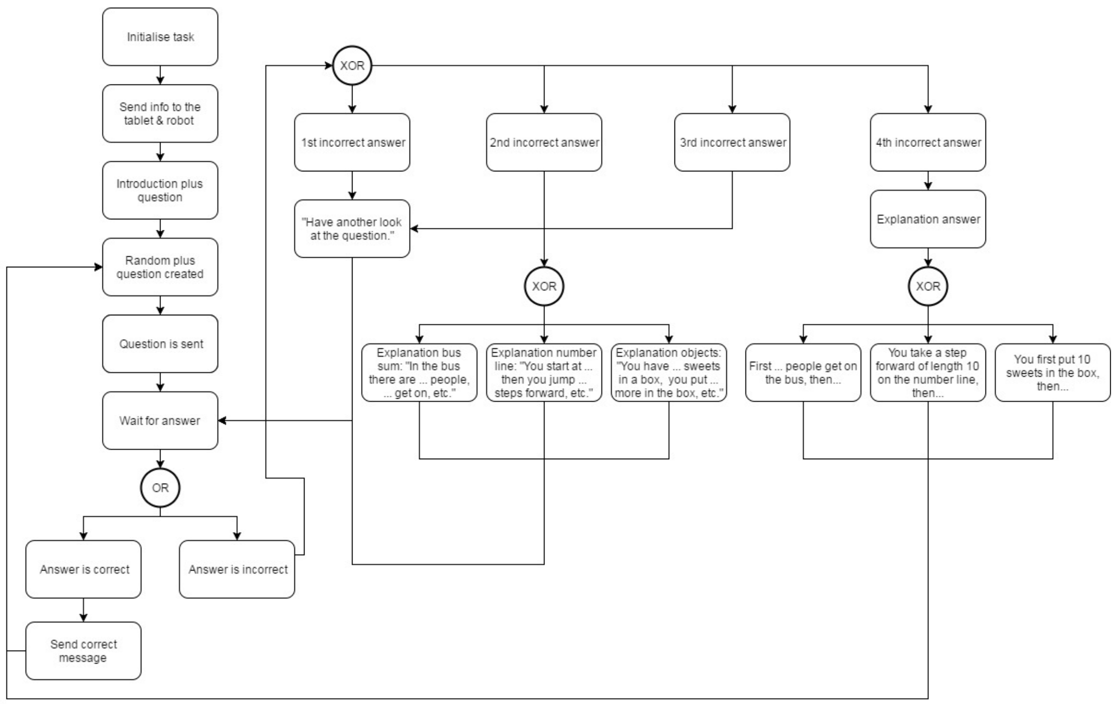


Figure 7.2: The GOAL agent flow diagram for a plus question

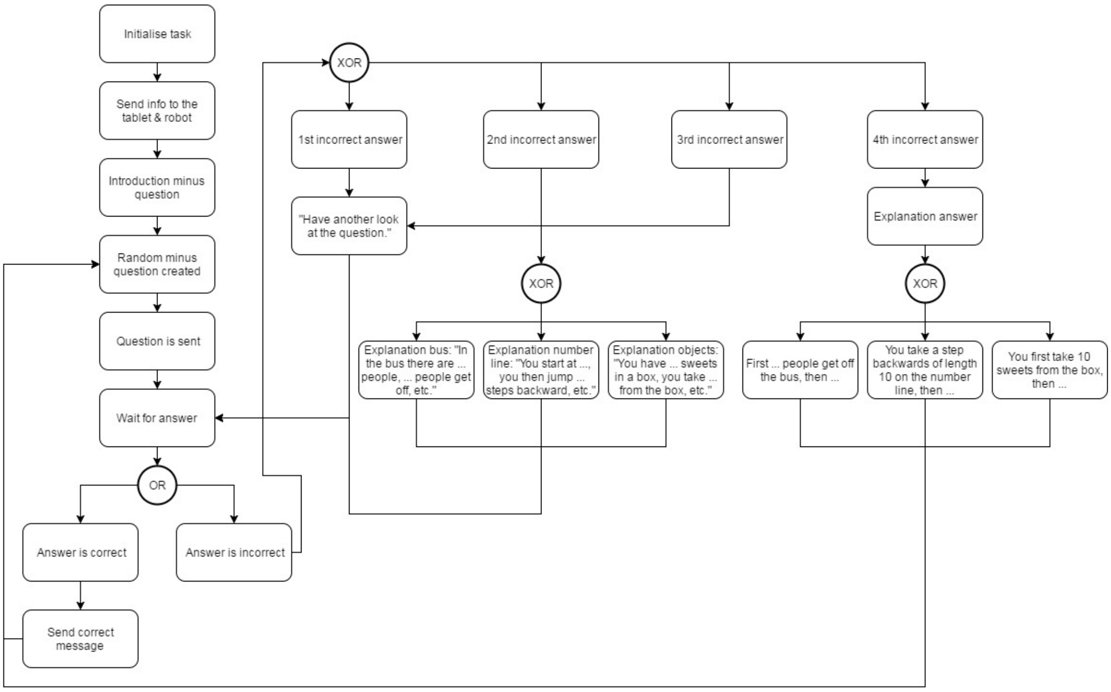


Figure 7.3: The GOAL agent flow diagram for a minus question

7.2 Times tables

For the times tables questions, using the SLO core objective 27, the following holds per age group:

- Year 3/4
  - Understanding the concept of times tables, by making use of situations where multiplication is used.
  - Making use of strategies such as:
    - $2 \times 3 = 3 \times 3$, where the two are interchangeable
    - $2 \times 3 = 1 \times 3 + 3$, where an addition is pulled out of the multiplication
    - $2 \times 3 = 3 \times 3 - 3$, where an addition is pulled into the multiplication
    - $2 \times 3 = 60$ and $4 \times 3 = 12$, where one is double the other
  - Being able to make use of the models, contexts and tables in which times tables can be found.
  - Automating and memorising times table products.
- Year 5/6
  - Expansion and practise of the memorisation of times tables.
- Year 7/8
  - Maintaining and application of the learnt knowledge of times tables.

The specific ontology flow figure for times tables questions can be found in Figure 7.4.

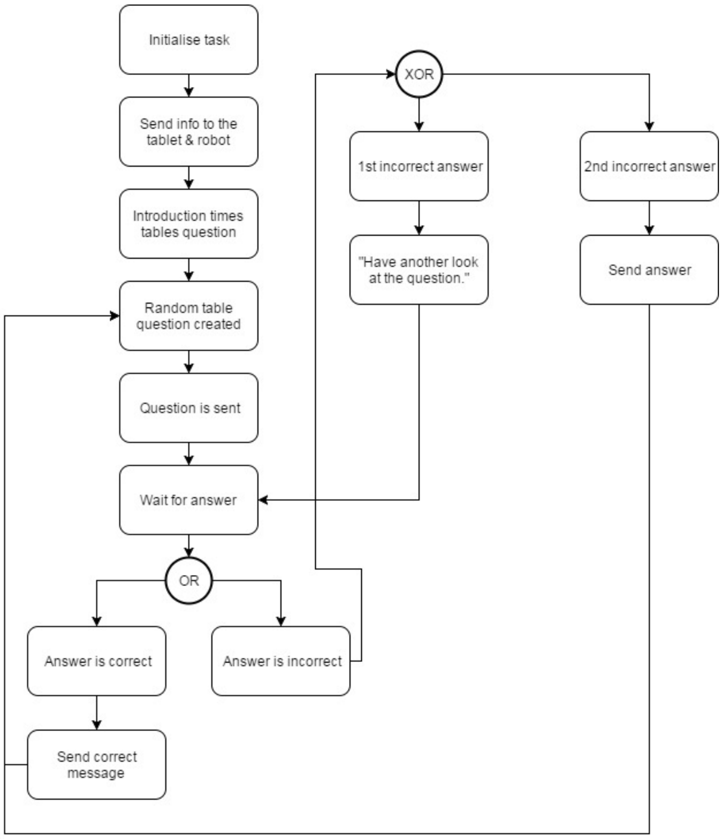


Figure 7.4: The GOAL agent flow diagram for a times tables question

7.3 Telling the time

When it comes to telling the time, using SLO core objective 33, the following holds:

- Year 3/4
  - Introduction of the analogue clock as a tool to tell the time.
  - Practise reading and determining the half and whole hours on an analogue clock, and eventually the quarters.
- Year 5/6
  - Expansion of the knowledge of telling time to all times that can occur on an analogue clock, and practised reading all the times possible.
  - Introduction to digital clocks, understanding the link between digital and analogue clocks and practising telling the time using a digital clock.

The specific ontology flow figure for a clock question can be found in Figure 7.5.

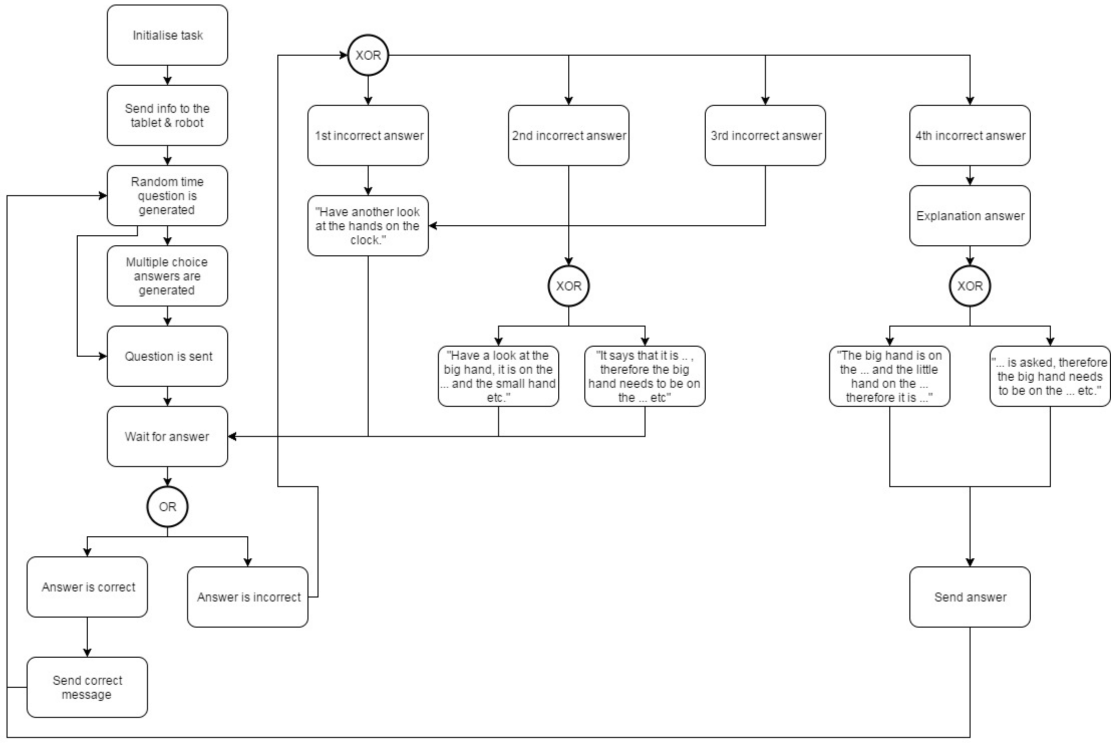


Figure 7.5: The GOAL agent flow diagram for a clock question

7.4 Long multiplication

Long Multiplication was a task that had been discussed but had a low priority; it was researched and flows were made in case there was enough time left over. Unfortunately, this task was not completed in this project, but the description using the SLO core objective 30 and the ontology can be found in Appendix E.

Chapter 8

Product design and implementation

The following chapter gives an overview of the specific interaction per type of question; the four types of tasks being plus and minus questions, practising multiplication tables and telling the time. The initialisation, the input and the explanations said by the NAO when the pupil gets the answer wrong are given. After this the individual components of the RekenRobot are discussed, followed by the Architecture design.

8.1 Overview

Initially, before the exercise begins, there is a welcome screen shown on the tablet, which waits for the teacher interface to decide which task the child will be working on. This welcome screen can be seen in Figure 8.1.

Figure 8.1: The UI which shows the Dutch 'Welcome' screen, which is the first thing shown on the tablet.

After this, the question which is to be answered will be shown. For the plus, minus and clock questions, the initial question screen will be shown again if the child has gotten an answer incorrect, giving them another chance. After this, explanations will be given, aiming to help the child to understand where they have gone wrong. For all the tasks, if the child gets an answer correct the 'Well done!' screen is shown, which can be seen in Figure 8.2.

Figure 8.2: The UI which shows the Dutch 'Well Done' screen, shown when a child gets the answer correct.

Seeing that this version of the RekenRobot has been designed and created for the Dutch educational system, the NAO will speak in Dutch. Any references to what the NAO says are translations of the NAO's speech. The components used in the final product will also be described, as well as the ontological flow.

8.2 Question type design

8.2.1 Plus and minus questions

Introduction

When it comes to the plus or minus questions, the NAO will originally start with: 'Today we will be practising addition' or 'Today we will be practising subtraction'. During the session, either only plus questions will be asked or only subtraction questions.

In the situation where the child is still learning addition or subtraction, the session will be 10 minutes long and the child can always ask the NAO for help. In the case that they are automating their addition or subtraction skills, the session will be 2 minutes long and they will not get any help from the NAO. The goal is to get as many questions right within those 2 minutes.

Initialisation

Two random numbers will be generated which the child will have to add or subtract. An example of the initial UI for a plus question is shown in Figure 8.3, while an example of a minus question can be found in Figure 8.4

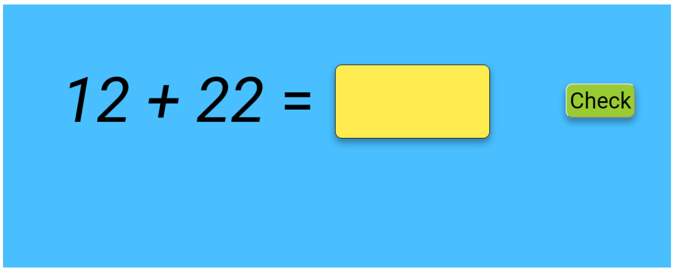


Figure 8.3: The UI which shows the visualisation of the question, in this case a plus question.

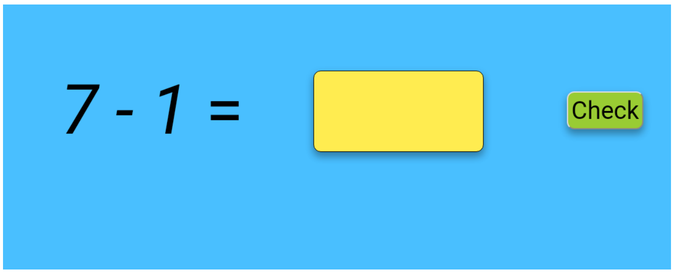


Figure 8.4: The UI which shows the visualisation of the question, in this case a minus question.

The teacher can specify which level the questions will be on, which in turn dictates in what range the answer of the question will fall. They can also set what type of explanation the child will get if the child gets the answer wrong: either using a bus or using a number line. Both the level and the explanation type is chosen initially, when the teacher is filling in the teacher interface. The levels that can be chosen in the teacher interface for the addition questions can be found in Table 8.1.

Level	Boundary	Example
1	Maximum answer is 10	$4 + 6 = 10$
2	Answer between 10 and 20	$8 + 12 = 20$
3	Answer between 20 and 50	$22 + 26 = 48$
4	Answer between 50 and 100	$42 + 29 = 71$
5	Answer between 100 and 150	$103 + 29 = 132$

Table 8.1: The different levels for the plus questions.

The teacher can set to what increment the final answer will be. The question cannot go through the next multiple of '10', so the question '$32 - 16$' will not be asked as the '6' causes the answer to go into the next 10's and this is considered to be challenging for children of this age. The levels that can occur for the subtraction questions can be found in Table 8.2.

Level	Boundary	Example	Notes
1	Largest number is maximum 10	$10 - 8 = 2$	
2	Largest number is maximum 20	$19 - 11 = 8$	
3	Largest number is maximum 50	$48 - 7 = 41$	The second number is between 0 and 9, and the first digit of the answer will be the same as the first digit of the first number, therefore $37 - 9$ is not asked in this level.
4	Largest number is maximum 50	$42 - 21 = 21$	The answer does not go into the previous multiple of 10, therefore $32 - 19$ is not asked in this level.
5	Largest number is maximum 50	$42 - 15 = 27$	

Table 8.2: The different levels for the minus questions.

If the teacher has chosen the number line, during the initialisation the GOAL agent will have to split the second number into increments. If the question asked is, for example '$12 + 19$', then the '19' will have to be split up into '10', '8' and '1'. The logic behind this split is that first steps of '10' are taken, if there are no more steps of '10' that can be taken, the next increment will be a number that added to the running total will give the next multiple of '10', so for '$12 + 19$', first the '+10', which gives '22', then '+8' to give a total of '30'. Finally, the last '1 is added to give the overall total of '31'. The same is true for the minus question, if for example the question asked is '$27 - 19$', then the '19' will have to be split up into '10', '7' and '2'.

Initially the tablet does not make use of these increments, but the GOAL agent does have to determine what these steps will be in case the child has given the wrong answer three times and the third explanation will be given.

Input

After the question has been asked, the GOAL agent will have to wait for the input of the child. This can either be done by saying the answer or by using the keypad on the tablet, which can be seen in Figure 8.5. The child should always be able to answer the question, even if the NAO is not yet finished with its explanation.

Explanation step 2

In the scenario that the child is still learning to add or subtract, thus not automating their given knowledge, the NAO needs to give a specific explanation if the child got the question wrong twice. The explanations that can be given are:

- Addition
  - The bus explanation: 'If there are x people in the bus, and y more get on, how many people are there now in the bus?'.
  - The number line explanation: 'We are now at point x on the number line, if we take a jump of length y, where are we now on the number line?'.
- Subtraction
  - The bus explanation: 'If there are x people in the bus, and y get off, how many people are there now in the bus?'.
  - The number line explanation: 'We are now at point x on the number line, if we take a jump backwards of length y, where are we now on the number line?'.

In the case that the teacher had chosen the bus explanation, two buses will be shown on the tablet. If the question that has been asked is, for example '7 − 1', then the first bus will contain '7', the bus stop will have a '−1' on it, and the last bus will be an empty box. The child has to determine what they believe the number should be in the last bus, in this case '6' should be filled in. The NAO will say the accompanying text that belongs to the bus explanation. An example of the UI with the bus is shown in Figure 8.5. After an answer has been filled in, the green check button can be clicked to send the answer to the GOAL agent.

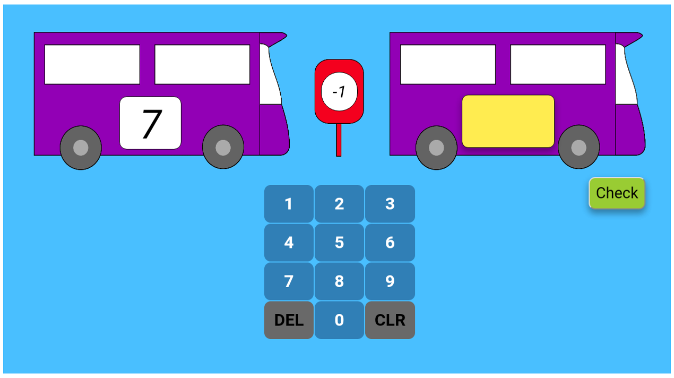


Figure 8.5: The UI which shows the visualisation of the bus explanation.

In the case that the teacher chose a number line, the first value will be shown on the number line, as well as all the increments of 5 within the range. An example of the UI with the number line is shown in Figure 8.6

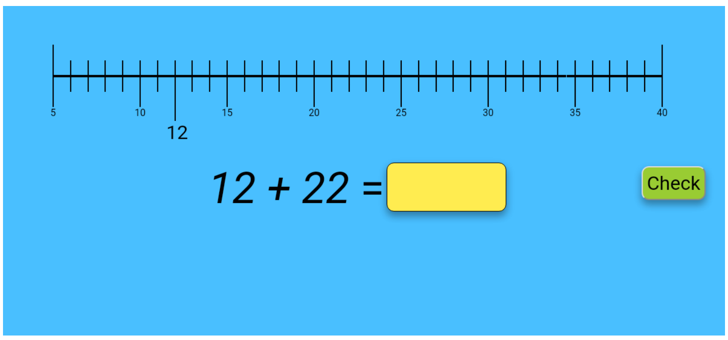


Figure 8.6: The UI which shows the visualisation of the number line shown when the child has gotten the answer wrong twice.

Explanation step 3

For both methods, the child should be given the chance to look at the question again. For the number line, during the 3rd explanation, intervals will be shown above the number line to help the child splitting the number up into smaller numbers. An example of the UI with the number line, showing the jumps above is shown in Figure 8.7.

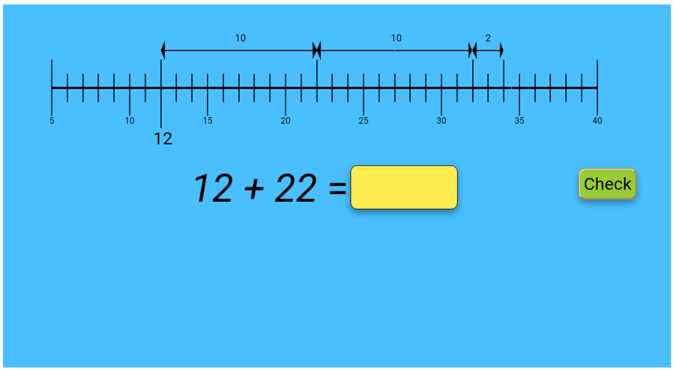


Figure 8.7: The UI which shows the visualisation of the number line explanation, with the increments above the number line.

Explanation step 4

When the child has gotten the answer wrong a forth time, the NAO will give the correct answer and explain what needed to be done to get to that answer. An example of the final number line with the answer can be found in Figure 8.8.

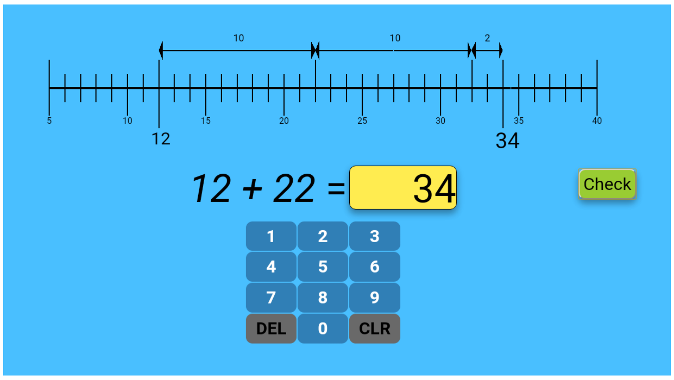


Figure 8.8: The UI which shows the visualisation of the number line explanation, with the increments above the number line and the final answer shown.

8.2.2 Times tables

Introduction

When it comes to the times tables questions, the NAO will originally start with: 'Today we will be practising your times tables'. The tables sessions are always two minutes long, and the child cannot ask the NAO for help as this is purely for automatising their tables knowledge.

Initialisation

Which level a child is at determines which tables will be asked. The second value, the value which the first number is multiplied against, will be a random number between 1 and 10. An example of the UI for the tables can be see in Figure 8.9, the number pad can be seen which means that the yellow button has been pressed, ready to fill in an answer.

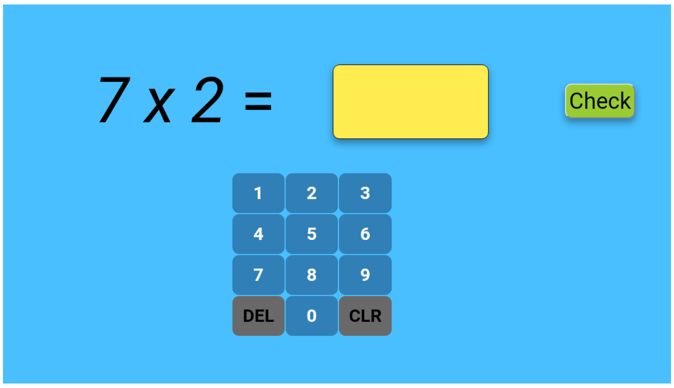


Figure 8.9: The UI which shows a times tables question.

For now, the tables will not have any form of explanation. If the teacher has chosen for tables to be asked, then the idea is that the child is practising and automatising what they have already learned. There will be a series of tables questions asked during the session.

The levels that can occur for the times table questions can be found in Table 8.3.

Level	Tables	Example	Notes
1	1, 2 & 10	$2 \times 7 = 14$	
2	0, 1, 2, 5 & 10	$0 \times 9 = 0$	Tables 0 and 5 have an 80% chance of being asked.
3	0, 1, 2, 3, 4, 5 & 10	$3 \times 3 = 9$	Tables 3 and 4 have an 80% chance of being asked.
4	0, 1, 2, 3, 4, 5, 6, 9 &10	$9 \times 3 = 27$	Tables 6 and 9 have an 80% chance of being asked.
5	0, 1, 2, 3, 4, 5, 6, 7, 8, 9 & 10	$7 \times 8 = 56$	Tables 7 and 8 have an 80% chance of being asked.

Table 8.3: The different levels for the times tables questions.

Response

The answer is always correct or incorrect. If the answer is correct, a new question will be created by the GOAL agent and then given to the child. If the answer is incorrect, then the correct answer is shown and a new question will be created by the GOAL agent.

8.2.3 Clock questions

Introduction

When it comes to the clock questions, the NAO will originally start with: ‘Today we will be practising how to tell time’. The session will last 10 minutes, and the pupil can always ask for help from the NAO.

Initialisation

The teacher chooses the level beforehand. Based on their choice, a random time will be chosen. There are 2 types of question that can be asked; analogue to word and word to analogue. In both these cases, the question that is generated is also the answer to the question that NAO asks. Three other times are also generated, so that the question can be asked in multiple choice form. The pupil then needs to decide which of the four given times they believe to be correct.

When telling time in the Netherlands, the time is said slightly different than in the British variation. As in English, the full hours, quarter to and quarter past are said the same; the time around the half hour are said differently. In British, the time ‘2:30’ would be said ‘half past 2’, while in Dutch a direct translation would give ‘half 3’, which implies half an hour until 3 o’clock. On top of this, there is also a slight variation in how the times between quarter past and quarter to are said, in Dutch, ‘2:40’ would be ‘10 past half 3’. A list of all the 5 minute intervals is given in Table 8.4 for the hour ‘7:00’, with the direct translations as they would be in Dutch. Of course the same applies for (e.g.) ‘3:17’, this would be ‘3 before half 4’ in Dutch.

Digital Time	British Analogue	Direct translation Analogue
7:00	7 o'clock	7 hour
7:05	5 past 7	5 over 7
7:10	10 past 7	10 over 7
7:15	quarter past 7	quarter over 7
7:20	20 past 7	10 before half 8
7:25	25 past 7	5 before half 8
7:30	half past 7	half 8
7:35	25 to 8	5 over half 8
7:40	20 to 8	10 over half 8
7:45	quarter to 8	quarter before 8
7:50	10 to 8	10 before 8
7:55	5 to 8	5 before 8
8:00	8 o'clock	8 hour

Table 8.4: The direct translations of different times between British and Dutch

This difference in telling the time means that the RekenRobot has to say slightly different speech than if it were to be programmed in English. The extra forms need to be taken account of when questions are being generated.

The types of questions that can occur in the time telling series can be found in Table 8.5. When half past 3 is written, it is meant in the Dutch sense ('2:30'), as the RekenRobot uses this form, see Table 8.4 for the translations.

Digital (XX:YY)	Function	Analogue (XX:YY)	Function	Word form	Boundaries
12:00	XX % 12 + 12	12:00	-	It is 12 hour.	YY == 00
13:11	XX % 12	01:11	-	It is 11 over 1.	00 < YY < 15
15:15	XX % 12	03:15	-	It is quarter over 3.	YY == 15
07:17	XX % 12	07:17	30 - YY & XX % 12 + 1	It is 13 before half 8.	15 < YY < 30
14:30	XX % 12	02:30	XX % 12 + 1	It is half 3.	YY == 30
20:35	XX % 12	08:35	YY - 30 & XX % 12 + 1	It is 5 over half 9.	30 < YY < 45
11:45	XX % 12	11:45	XX % 12 + 1	It is quarter before 12.	YY == 45
23:55	XX % 12	11:55	60 - YY & XX % 12 + 1	It is 5 before 12.	45 < YY < 60

Table 8.5: The different types of questions that can be asked for the clock questions.

Input

The input is the digital form of the time that the pupil has chosen. In an example where the text 'It is 2 o'clock' is asked and 4 analogue clocks are shown with different times, the digital version of the clock that the child has chosen will be sent to the GOAL agent. A word to analogue UI can be seen in Figure 8.10, while an analogue to word exercise can be seen in Figure 8.11.

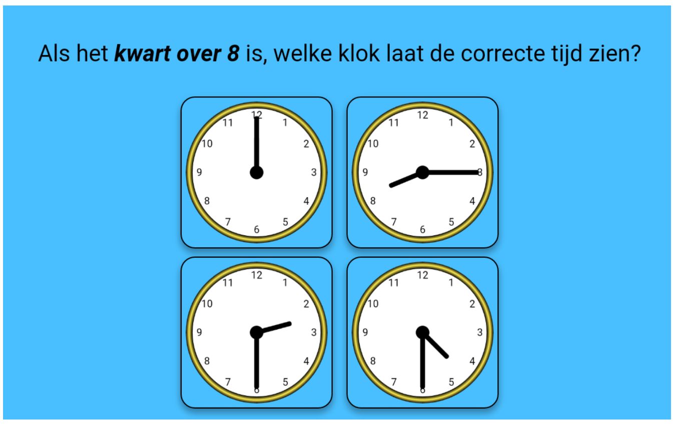


Figure 8.10: The UI which shows the visualisation of a clock question, the question being asked in word form and the four multiple choice answers are analogue clocks.

Figure 8.11: The UI which shows the visualisation of a clock question, the question being asked in analogue form and the four multiple choice answers text.

Explanation step 2

When the child has gotten the answer wrong twice, the NAO will give an explanation that puts emphasis on the clock hands. Two examples of what the NAO might say are: ‘It is quarter past 8, where does the big hand go when it is quarter past?’, and also ‘Have a look at the small hand, it is found between 8 and 9 and the big hand is on the 6, so what is the time?’

Explanation step 3

Here a hint is given. An example of this could be ‘If the big hand is found to be on the 6, then it must be half past something. Have another look at where the small hand is’ or another example ‘The small hand is almost at the 10, so it is a time just before 10’.

Explanation step 4

Here the answer is given. The NAO explains step by step how the answer as reached. The meaning of the big hand is explained, for example if the big hand is on 3, then the time is quarter past. The rules for the small hand are also revised; the fact the number that the small hand is closest to is the number that is often said when telling the time.

8.2.4 Long multiplication question

The Long multiplication was also designed, though unfortunately not implemented. The structure of the design can be found in Appendix F.

8.3 Components

The RekenRobot was divided into 4 different components, the GOAL agent, the formulation, the teacher interface and the tablet.

8.3.1 GOAL agent

The GOAL agent keeps track of all the messages that are sent back and forth between the NAO, the teacher interface and the tablet. It is also responsible for generating the questions to be asked, making use of the data given via the teacher interface, as well as the answers that the child gives either through speech or through the tablet.

8.3.2 The Formulation

The formulation of the questions, the types and the explanations also had to be done. This meant researching and deciding what the best ways of setting the levels were, having the questions being automatically generated and thinking of ways of rewarding the child when they have gotten answers correct.

8.3.3 Teacher Interface

The teacher interface is used so that the teacher can have control over the system. Using the teacher interface, the teacher can fill in which child will be working with the NAO next, choose what type of task the child is going to be doing, decide which explanation type should be used if the child has gotten the answer wrong and also the level that the questions should be. The teacher can also pause the NAO, but this can only be done with the IR system. The teacher interface is at the moment independent of the IR site, but will eventually be linked together; coupled to the web-page on the server. The UI for the teacher interface can be seen in Figure 8.12.

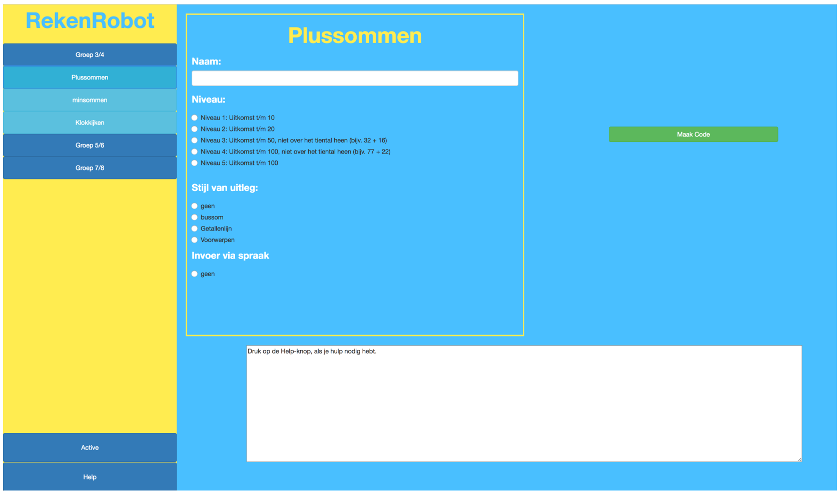


Figure 8.12: The UI for the teacher interface.

8.3.4 Tablet

The tablet is used as a way to visualise the questions that the NAO asks as well as a 2nd way of answering the question, seeing that the voice recognition does not always pick up on what the child has said. The tablet also shows the visual explanations that the NAO can give when the child has gotten an answer wrong multiple times.

8.4 Architecture design

The Architecture design can be found in Figure 8.13. The green box shows the section that was added for the RekenRobot, the rest of the system already exists. The real Architecture design is much more complicated than the one seen in Figure 8.13, but due to the confidentiality agreement signed with IR, more cannot be shown.

The GOAL agent will always have the lead in the program. When the customer starts a program, the agent will be started with initial data. The agent will decide what behaviours the robot should do and what the tablet or GUI should display. Messages will be transported through this architecture. The robot is also able to send messages back to the GOAL agent, which could include a message that it is done with a behaviour. The tablet is used to input answers, so these messages will also be sent to the GOAL agent. The GOAL agent will interpret the messages and decides what will be done next.

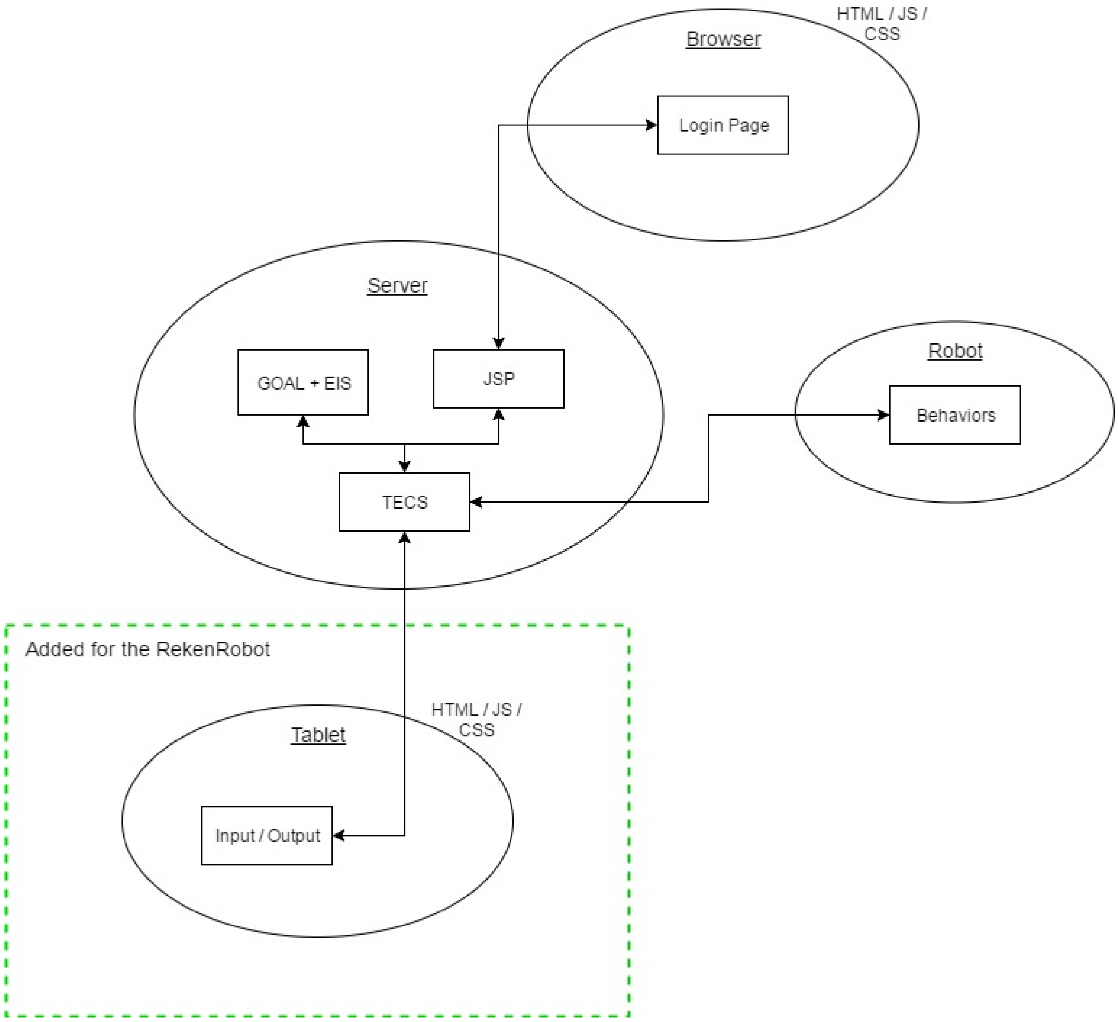


Figure 8.13: The Architecture Design for the RekenRobot.

Chapter 9

Testing, SIG and user feedback

This chapter will discuss how the code written for this project was tested, as well as how the product was subjected to user tests at two primary schools.

9.1 Testing the GOAL agent

The NAO was tested in two ways, with the actual NAO and with a feedback faker. The feedback faker was a small JAVA module designed to simulate the messages from the NAO. It was created specifically for the purpose of testing, meaning that the NAO did not have to be set up every time the GOAL code needed to be tested. Given that test cases cannot be written for GOAL, the main form of testing when it came to the GOAL agent was robustness testing, therefore the GOAL agent was tested continuously. The product was used as if an actual user would, to see if the system did what it was expected to do. The tablet was also tested in this way, to see whether the tablet showed the correct pages when the robot was speaking.

9.2 Testing JavaScript

The JavaScript code that was used for the tablet and the teacher interface was tested while doing the robustness tests for the whole product. While using the product, it was verified that the interfaces displayed the correct behaviour. Where possible, QUnit tests were used to test the correct parsing of the messages that the interfaces received.

9.3 Testing Prolog

GOAL heavily relies on Prolog in its workings. In the GOAL code written for this project, there was additional Prolog code in the form of helping functions, to facilitate simpler GOAL code. Therefore, these Prolog helping functions were also tested. This testing was done in an external file, as the GOAL agent could not run the test file itself. The tests in Prolog use the predicates or functions to verify a known outcome.

9.4 User testing

9.4.1 Testing at 'De Gantel'

Towards the end of the project, the system was tested at the primary school 'De Gantel' in The Hague. The test included the robot and tablet, but did not yet include the teacher interface. The reason for this is that the teacher interface will most likely be largely changed when it is integrated into the platform of IR. The test was conducted with children from the third grade of primary school, as they were the primary target. For this test, the robot was located in the teachers' lounge instead of the class room. This way, the focus could be on the user experience and the details of the system.

First the whole class came to see the robot and received a short introduction to the system. Due to time constraints, pairs of pupils came to work with the robot and did one or two exercises. The robot was met with general enthusiasm, as was to be expected. The robot seemed to spark curiosity and most children wanted to use it immediately. The teachers were also very enthusiastic about the robot and were curious about the reaction of the children. Although pairs of children were sent to the robot, the test was essentially limited to one-on-one interactions. Fortunately, the children only needed to hear the one-sentence explanation of the tablet interaction once. They immediately understood how to use the tablet.

The children were curious about what the robot could do. Considering this was the first test of the prototype, their expectations were not realistic, though this is to be expected of such young children. During the test there were minor internet connectivity problems. The speech and commands were slightly delayed, but the system reacted in the desired way.

In the introductory speech, the robot asked the children if they were ready. The robot does not listen to a respond, but some children did answer and did not seem to notice that the robot was not listening. The names of the students were pronounced rather well, often saying the correct name or at least one that sounded similar.

The test included pupils of multiple levels of understanding of the arithmetic material. The first child was really advanced and did not make a single mistake. The observation with all pupils was that they all took their time with an exercise, instead of guessing the answer. With the first child, it became apparent that just answering a barrage of questions correctly became too repetitive. This type of child should either receive even more incentive to keep going, or be excused from the rest of the exercise.

One of the bugs that was spotted was that when the difficulty level was increased after a certain number of questions were consecutively answered correctly, the system went from level 1 to a level higher than 2. Additionally there was a case where the same joke was told twice in a row. This might indicate a clean-up problem.

Among all the children it was indicated that they thought that the robot interactions were fun and clear and that they liked the jokes. A pair of girls said that they wanted the jokes sooner and more often. When looking at the children during the punchline, it was remarked that it seemed like they did not understand all jokes. To be sure, simpler jokes will be used in the future.

The ending speech, as it was during the testing, was too abrupt and did not mention that it is stopping because the time is up. This was improved for the following test. Additionally, the robot should not say the same for every number of correctly answered questions. This was also adapted, ready for the next test set.

Despite the expectations, all children were very delicate with the tablet. They were patient and did not start pressing buttons until they were sure the interface was ready. The children displayed this behaviour without first being instructed to have patience with the system.

There was an instance where the child answered the question wrong four times and the next question was asked immediately after without a pause. The robot was still giving the explanation, but the tablet already allowed the child to answer the next question on the tablet. This stress test was passed however, as when the check button was pressed on the tablet, the robot skipped to the correct stage. Introducing an extra message to let the tablet wait for the robot to finish will be a recommendation for the next iteration of the project.

One of the children remarked that is was odd that when a question was answered correctly, the tablet and robot did not always give the same compliment.

One interesting discovery was that when the children learnt that after their second mistake and the number line was shown on the tablet, they would do the first two attempts wrong or guess, so that they could make use of the number line. This lead to the idea of introducing a button to skip the current stage. This will however be a recommendation and not be included in this iteration of the project.

Overall, the first test went very well and the children and teachers responded positively to the system.

9.4.2 Testing at 'De Voorsprong'

After the first test with the 'De Gantel', a second user testing session was held at 'De Voorsprong' in The Hague. Children of both year 3 and year 4 tested the RekenRobot, which is the target audience for the product. The testing was done in an empty room, next to a year 3 classroom, meaning that the testing did not disrupt the rest of the class.

Initially, there were problems with the internet, meaning that all the libraries had to be downloaded via a hotspot to get the system running. This problem had not been taken into account. Consequently, a large amount of time was lost trying to get the system running. When the system was finally up and running, the testing went rather smoothly.

One of the few issues that occurred was that the NAO kept getting too hot. There were times when it froze completely, due to the heat. The weather was also very hot that day, which meant that the NAO heated up quicker than usual. A problem was that the NAO did not always say when it was getting too hot. A possible solution for this is to put a fan next to the NAO.

The initial idea of the NAO was that it would work one-on-one with the children, but they were often quite shy. Therefore, all the testing was done with pairs. Pairs of both advanced as well as 'weaker' children were given the chance to work with the NAO, to see how they perceived the experience. It also meant that, for some of the pairs, the children with the stronger grasp of arithmetic often answered the question; leaving the other child to sit and watch. Each set of children did a different task, meaning that all 4 tasks were tested.

Most of the children said that they liked working with the NAO. They often came into the room unsure of what they had to do, but quickly got the hang of it and actively answered the questions. The teacher of the year 3 class got the whole class to stand before the NAO, so that everyone could have a look at it. This means that when the children came to work with the NAO, they already knew what to expect. The year 4 pairs, being further away, were not able to do this. This meant that the children being led to the robot were often nervous and did not know what to expect.

There was one case where two girls from year 4, who had never seen the NAO before, found it very scary upon entering the room. After seeing how the NAO works and realising that it could not hurt them, they really enjoyed working with it, fighting for who got to answer the question on the tablet. They even wanted to work with it longer, begging to have another turn. This shows that, as initial contact with the NAO, it is better to first have the whole class be introduced with the NAO.

When it came to the jokes, the pupils did not always understand them. Sometimes they responded, saying what they believed the answer to the joke or riddle to be, which then clashed with the NAO saying the answer. This meant that they did not always pick up on what the answer was. When they did understand the joke, some did not find them funny and others looked like they dared not laugh. During the test, too many jokes were given during a session. There should be a maximum of two or three jokes given, no more. The pause after the joke should also be shorter, as the children were waiting for the next question. Not all children liked the jokes and got frustrated when a new joke was told. What could be adapted is that the child is given the choice as to whether they want to listen to a joke. Also a few children thought that the jokes were spoken too quickly, and they did not give the child a chance to think about the answer before the punchline was given.

What also became apparent was that the level system, in the form that is was in during the test, was not satisfactory. Initially, the levels also included the levels that came before. For example, with the plus questions; level 1 had the maximum answer as 10, level 2 had maximum level 20 etc. The original levels can be seen in Table 9.1. This meant that level 2 also asked question that could occur in level 1. If the numbers generated happened to all be small numbers, the child could easily progress to the next level. There was one child who said: 'We've gone up a level, but the questions have gotten easier'. When a harder question then came, the step up was too large. They had not been able to practise with questions that were actually representative of the previous level. This feedback has been taken into account for all the question types, and the current level systems can be found in chapter 8. There were also too few questions per level, more question should be asked before they go up to the next level.

Level	Boundary	Example
1	Maximum answer is 10	$4 + 6 = 10$
2	Maximum answer is 20	$8 + 12 = 20$
3	Maximum answer is 50	$22 + 26 = 48$
4	Maximum answer is 100	$42 + 29 = 71$
5	Maximum answer is 200	$123 + 49 = 172$

Table 9.1: The originaly levels for the plus questions.

The NAO did surprisingly well when saying names, though the names often had to be written as they would be said, instead of how they are spelt. There were a few names where it struggled, though it was not possible to experiment with different spelling of the name as the session had already begun.

The interactions with the NAO were fascinating to observe. When the robot asked: 'Are you ready for a question?', the children often nodded along. A few of the children wanted to touch the NAO, play with its hands and one child hoped that it would give her a handshake. Also many children believed that the NAO could see them, thinking that its eyes were following them.

What was very interesting to see was when one of the children got the answer wrong twice, and then a bus explanation was shown on the screen, he recognised it. He suddenly knew how he had to think about the question, and sat up more 'seriously'.

An observation made multiple times was that the text of the tablet, when the 'Well done!' screen appeared was not the same as what the NAO says. They would have preferred it if the two were synchronised.

For the clock questions, the levels increased but there was a definite division in what the children could do. They found the half, whole and quarters easy, but then they could no longer answer the questions as they had not yet learned the material, for example 10 past. Therefore it would be better if there was a cap on how high a child can go with the levels. The cap being the material that they have learned thus far. To make the clock questions more difficult, the multiple choice could be out of six. This means that the child has less chance of guessing the correct answer. With 4 clocks, they could use the process of elimination to get the correct answer.

The children did need to be instructed to click on the yellow rectangle on the tablet, to be able to fill in the answer. Additionally, a few of the children did not always understand what the NAO was saying and asked if a repeat button could be made, to repeat the last thing said.

One pair tested the voice recognition. The first of the two boys had a strong voice, so it could be recognised, while the second had a very timid. The NAO could therefore not hear his voice. The boys were then given the tablet to use, and told that they could also still answer the question vocally, but they preferred answering the question via the tablet.

Quite a few children did not like how the session finished. They were often still working on a question when the NAO said that the session was finished and then gave them the number of answers that they had gotten correct throughout the session. This meant that there was still a question on the tablet screen, which they could not give the answer to.

Seeing that many of the children wanted to have another go, this would imply that the session could be longer. What also could be an idea is to give the choice to the child, for example after 5 minutes ask them if they would like to continue working or stop.

When asked if the pupils liked working together with the NAO, there was a range of responses. Some would prefer working with it alone, others preferred working together and one boy said that he would like it most if the NAO could work with a whole group of people.

9.5 Feedback from the Software Improvement Group

The Software Improvement Group[1], also known as SIG, is a company that has strong ties with the TU Delft. Their aim is to improve code quality and to make students more aware of the maintainability of code.

Code was sent to SIG in the 6th week of the project, when around 75% of the code had already been written, as well as the final code at the end of the project.

The main points of improvement that came out of the first SIG report were code duplication and unit size. Neither of these came as a surprise and both were found mainly in the tablet code and were done to give a better overview of which objects were being created for the tablet. Software testing had not been done, given that GOAL code cannot be tested. JavaScript was used for both the teacher interface and the tablet, therefore most of the tests were stress tests and visual tests. Testing the connection between the GOAL agent and the teacher interface/tablet was also missing. The full Dutch text for the first feedback can be found in Appendix G.

After the commentary regarding the lack of testing, more time was spent researching testing methods for this types of languages. For the JavaScript code, the QUnit Testing Framework

[1]`https://www.sig.eu/`

was used to test the correct parsing of messages. For the agent code, only Prolog tests could be written.

Significant effort was spent to tackle the duplicate code problem. The splitting of the files and use of more functions allowed the reduction of similar code. The unit size problem has been decreased but not completely eliminated. However, the clock and number line JavaScript files are still rather large. This is because they were parts that are specific to the explanation methods. This is not code that is meant to be reusable. Attempting to enhance the re-usability would only result in unnecessary baggage. Splitting this these files up would not improve the readability.

The feedback for the final code showed that the improvements made had been picked up on. As expected, the Unit Size for a few files was considered to be too large, but the reason for this has already been described. The full Dutch text for the final code can be found in Appendix H.

Chapter 10

Product evaluation

This chapter will discuss the evaluation of the project and whether the final product can be considered a success. First, the amount of goals defined in the MoSCoW method that have been successfully implemented, will be examined. Second, an evaluation of the development process during this project. The last section will determine whether the project can be considered a success.

10.1 Discussion of the MoSCoW method

At the start of this project, the user stories were ordered using the MoSCoW method. Not only the desirability of the user stories were considered, but also the feasibility of a user story in the time that was allocated for this project. The goal was to choose the Must and Should Haves in such a way, that the majority of the project will be spent on those tasks. The Could Haves were tasks that could be done, depending on whether the Must and Should Haves had been completed. Because of the fluid product definition, it was difficult to create a realistic and satisfactory set of tasks. At the conclusion of the project, it turned out that the Must and Should Haves were largely completed.

The client indicated that the way in which the system is paused should be the same for all applications on the platform. As the specifications for this design are not yet in place, only the agent-specific capabilities for pausing have been implemented. The full capability to pause will be realised when the application is deployed on the platform.

One of the goals was to give the children rewards after specific milestones have been reached. At the start of the project this was envisioned to be a high-five or a dance. However, in the end product jokes are the reward.

The movements can be easily added after this project. This feature was omitted due to time restraints. The login system was envisioned at the beginning, but was scrapped after discussions with the client. The login system should be the same for all applications, therefore this will be implemented by the clients themselves. The active and physical exercises were also omitted due to time constraints. It was deemed more important to work on the basis of the application instead of specific lessons. The rest of the Must and Should Haves have all been completed according to the planning.

10.2 Process evaluation

In the past the projects at IR have been completed using an agile-development strategy. At the start of this project it was indicated that the same strategy should be adopted, supported by JIRA. However, because of the nature of this project, the issues were often changed or redistributed during the sprint. Estimation of the feasibility and duration of issues were hard to make, resulting in on-the-fly changes during each sprint. Weekly meetings were arranged to discuss the progression and show the developments. As work was done in a room that was close to the client, there were often unplanned short meetings and deliberations to make decisions. However, because the client had a very busy schedule, there have been instances where a meeting was needed to discuss the progress of a certain issue, but quite some time passed before a meeting was possible. During the course of the project this has led to the loss of efficiency and time.

10.3 Success criteria

At the beginning of this paper, there were several functional and non-functional requirements specified. The degree in which the functional requirements have been met, can be determined by looking at the previous section about the MoSCoW method. How well the non-functional requirements have been met, can be determined by looking at the results of the testing at the primary schools. As described in the section about user testing, the children and teachers were excited about the product. It was evident that the interface was easy enough to understand for the children and the use was largely enjoyable. How easy it was for the teachers to use the teacher interface has not been tested as it will change greatly in the future of this project, therefore considered unnecessary to test at present. Whether the system is easy to adapt and deployed on the platform can only be seen in the future, when the link is made. However, in discussions with the client, both parties have faith that this is indeed the case.

Chapter 11

Conclusion

At the start of this project, the goal was set to create an application that makes use of the NAO robot to assist teachers in basic arithmetic education. The challenge was to build this system in such a way, that it could later be integrated into the existing one of the client. The product of this project, will be one of the applications in the portal of the platform. As the platform is still in development, a lot of effort was put in designing the system to later be integrated with the other components, as well as to set the precedent for future applications.

The research phase was used to get an idea of how arithmetic is currently taught at primary schools, as well as looking at how robots have been used in education to date and how the NAO can be used in this project. In the research section lessons were drawn about what the system should be able to do. For instance, personalisation and adjusting the settings while doing exercises are important factors to consider when creating a system that is not just fun, but also effective. In addition, using the movements of the robots and jokes to humanise the robot, has been shown to increase engagement with the children. Making sure that the robot does not always say the same lines, helps with humanising, as well as, keep things exciting.

After the ten weeks of the project, a system was delivered that uses a tablet together with NAO. The interface that the teacher will use will in the future be largely altered to fit in the existing portal of the platform. The teacher can assign one of multiple types of exercises (Plus/Minus, Times Tables and Telling time) to a pupil, as well as set the difficulty and select a type of explanation. Whether the voice interface is enabled is also included in the interface. The alternative is using the tablet to answer the questions. The tablet is also used for visualisation and explanation. After certain milestones the difficulty will automatically adjust, while the children are rewarded with jokes.

After testing the product at two primary schools, the results were that children are excited and stimulated by the use of the NAO. The system is understandable enough to require very little explanation.

This project was never meant to realise a product that can be deployed tomorrow, but the result is a very solid basis for further improvements. For instance, adding more movement to the robot, adding vision capabilities, more types of exercises and more visualisation.

Overall the project went well, a product was delivered that contained the most important features for a solid basis of future developments. The testing was very positive, there was a large amount of positive feedback as well as great ideas for the future of the RekenRobot.

Chapter 12

Discussion

This chapter will discuss the process of the project as well as the ethical issues associated with robots in the classroom.

12.1 The Process

Overall the process of the project went quite well. Every week, a retrospective was written to keep track of what was going well in the project and what could be improved. The project started well. The interviews with the pilot schools were interesting and informative and gave a good indication as to which form the project should take. The teachers were enthusiastic about the idea of the RekenRobot, which was very uplifting.

The initial aim was to do the interviews during the research week, but unfortunately the two assigned research weeks were also during the 'May holidays' for primary schools. This meant that there were no replies until at least week 3, meaning that interviews with teachers were held during week 4 and 5.

From day one there were already many ideas as to what the RekenRobot should be able to do. The interviews helped to think of even more ideas. There were a lot of papers on the topic of robots in the classroom. Additionally, the education system in the Netherlands is very open about what they teach children. Therefore it was easy to access and gather the required information. There are also different examples online of websites that are being used to practice arithmetic.

Initially getting the NAO and the server working went slower than predicted. Getting NAO connected to the server was an issue that recurred throughout the whole course of the project, which took some effort at the start of the project. A solution to this was that a feedback faker was created, which was very useful in testing the product as it meant that the NAO did not have to be set up every time testing was to be done. This saved a lot of time.

There was also a lack of initial documentation and problems with setting up the server. Because the client and TU Delft coach were both extremely busy, it was not always possible to get their help. For the first few weeks there was also a bug which made the NAO say everything double, this appeared to be an issue with the server and the client was able to solve this problem in the third week of the project.

The project itself and the process went well though, there were no conflicts within the project group. The communication between the team members went very well; this was helped by the fact that all team members worked at the same location and could easily talk to each other.

An issue that became apparent rather late, is that much of the documentation was written down on paper, and not on Confluence, therefore it was difficult for the TU Delft coach to see what the Architecture Design and the Ontology flows were. This was resolved, but should have been done earlier on.

Another issue was that the teacher interface and the tablet both had a UI, which involves some form of creativity to make them look good. The priority however, was to get a working system. The visual side could always be adapted later on; to also give the same look and feel as the IR system.

IR had multiple teams doing different projects, one of which would also be linked to the IR system after their project was done. The RekenRobot project was at times dependent on this other team, for example the sockets that were needed to be able to locally run the RekenRobot was to be done by this team, which took longer than expected. Therefore further work on this had to be postponed until they were finished.

The initial idea was to add a database to our system, which could keep track of a child's progress. After discussions with the client it became apparent that the IR system would have the database, because the client wanted similar functionalities across all applications on the platform. This meant that linking the original plan to this system would be much more difficult than expected. Fortunately, the work for the database had not been done yet, otherwise this would have been wasted time. Instead, this will be a future functionality.

There should also have been a meeting much earlier in the project to discuss specifically the GOAL code with the TU Delft coach. This meant that later on during the project, a large part of the code had to be adapted; mainly improving the predicates. This was also true for the messages, as they needed to be changed to work with the IR system. This was not ideal, but did improve code quality and would make it easier to link the system to the IR system in the future.

12.2 Ethical issues

Using robots in education introduces some interesting ethical questions. In this section a short discussion about these relevant issues will be given. One of the first things that teachers worry about when discussing robots in education, is whether they will still have a role to play. It is no surprise that they immediately think about their livelihoods and their futures. This does not only raise the question whether teachers are replaceable, but also if it should happen. One of the key elements of good education, especially for young pupils, is the compassion and humanity the teachers provide. It could be argued that robots, no matter how advanced and cost effective, cannot imitate these attributes. A solution to this issue is that robots will only be used as tools, and never be able to fully replace teachers. If laws are passed that there must be a human teacher in a classroom, even if robots are being used to teach, this may remove the worry that teachers feel at being replaced. On the other hand, robots could help with the problem of standardisation. At the moment, many schools in the Netherlands have different ways of teaching, and standards of what a child must be able to go to the next level. Robots would be able solve this problem, if the same program is given to each school. Of course, each school could also ask for 'tailor made' programs, based on their way of teaching and the methods that they use. These are all issues that will need to be discussed and solutions will need to be found, which is beneficial for everyone.

Another ethical issue that arises often when discussing Information Technologies, is the problem of privacy. Children may say things that they do not understand or mean. However, the consequences cannot be too severe, precisely because children are still learning and finding the boundaries. A human teacher realises this and will deal with the child accordingly. The problem is that robots could record images and sounds, and thus record these slip-ups. The question is whether this is acceptable and how the children can still be free to discover their boundaries. Privacy is a very big issue, an adult does not want what they said as a child to return to haunt them. The whole point of going to school is to learn, not only subjects, but also social interactions; the process of growing up. If the child in question has famous or known parents, any recording of the child could be used against them. A solution to this is that if the child is filmed, that these records are deleted as soon as the robot has processed the data. It could be argued though that previous recordings can be used to see if the child has developed, to see how much they have learned since they started working with a robot.

The advantages of having a robot in the classroom, also comes with a set of disadvantages that pose some ethical dilemmas. One of the key features of the robots is the voice interaction. The ability to differentiate words is however not always perfect. This could be problematic for pupils with speech impediments. When the robot does not understand a pupil, but does understand others, this could be frustrating for the child and may lead to insecurities. The question that has to be answered is whether there is an obligation to make sure this type of situation does not occur. The same could be applied to children with accents. The best solution would be that voice recognition systems where updated and capable enough to recognise what any child has said, but unfortunately not all systems are adequately developed to be able to do this. Hopefully in the future this will be possible, but of course it will not be perfect. Frustrations may still occur, therefore it is important to tell a child, if they really want to work with the robot, that it may not be able to always understand what they are saying, and always allows the child to be stop whenever they want. A solution to the current systems is to have different ways of entering answers to question, not only with voice but also with movement or touch. A deaf child will not be able to hear the robot speak, therefore a tablet can be very useful to visually ask the child a question. There should be many different formats for different children, and it is equally important to have teachers keep an eye on the interactions, and intervene if necessary. A judgement call can be made by a teacher, whether it is wise to allow a certain child to attempt to interact with the robot, as they know best whether this will be possible.

As discussed before, using robots in education provides the teacher with opportunities to differentiate between the pupils and adjust the teaching material accordingly. This could be beneficial to the pupil, but also introduces new problems. When the children are split into groups based on how far they are with the teaching material, the struggling children are denied the opportunity to 'learn' from the ones that are more advanced. The struggling children can sometimes learn by example, when they see others doing the task at hand. Additionally, children might feel as though they are 'less' than the other children when they are put in a group of weaker pupils. Even if they themselves do not mind, the parents could have a problem with putting 'labels' on their offspring. A similar discussion is whether the robots should be 'gendered'. For example, if all robots look male, the girls might feel left out. These are all sensitive subjects, and should therefore be thought about in a sensitive manner. The issue of labels is a two sided coin, labels can make a child feel 'less worthy', as there are children who are better or smarter than them. It must be remembered that in a classroom there are children that are all in the same age group, but are not all the same age. There are children who are born September 1st, in the same school year as a child born August 31st of the following year. That is an enormous age gap and at a young age, when children are learning, this can be very apparent. Therefore, a child who is older will probably develop quicker than a child who is a year younger. It is not the fault of the child that they are younger, but they may feel unintelligent compared to the older children. Therefore labels may not be the best way to determine how well a child is doing. On the other hand, in the real world, there are always people who are older, smart, faster, richer, stronger etc. Therefore, it could be seen as beneficial if a child learns this from a young age. They will not always be the best, they will not be able to always get what they want out of life. The reality of life should be made apparent to them when they are young, and not when they start working for the first time. For the issue of gender, the robot should seem to be as neutral as possible. A girl will not necessarily prefer working with a 'female' robot and a boy may not prefer working a with a 'male' robot. If robots were 'gendered', it would of course be possible to allow a child to decide which robot they want to work with, but the best scenario would be to have a neutral robot, where gender would not be an issue.

Another issue that can occur when working with a robot is the amount of time a child is allowed to work with a robot. A child may get too attached to a robot, and start seeing it as a human or even a friend. This is of course not true, as the robot is an inanimate object that cannot think for itself. Therefore letting a child get too attached may have negative consequences; if a child is interacting more with a robot and less with other children, then they are not developing their social skills. They are not learning how to interact with other humans and therefore might not be able to develop the skills needed to have actual friends. The children should therefore be told what exactly the robot is there to do and teachers should not allow a given child to spend too much time with a robot. They should still participate in classes and learn to go about with other humans.

These ethical problems do not have easy solutions. When it comes to robotics, developers have to be sensitive to these, as well as other, issues. Seeing that robot-assisted education is starting to take off, it is important to keep these dilemmas in mind and try to find the best possible solutions.

12.3 Recommendations

Due to time constraints, not all envisioned types of lessons have been implemented. The final product did not contain long multiplication. Therefore a next step that could be taken would be to implement this. There are many different approaches to the long multiplication, but the one discussed had the child bring a sheet of paper or chalk board to do the calculations on. The tablet would have a box where only the answer can be filled in. If the pupil has answered incorrectly a few times then help boxes are added which they can fill in, which are also checked by the GOAL agent. The background information and product design for long multiplication can be found in Appendix E and Appendix F respectively. Another option discussed was being able to use half the tablet as a page, where the child can scribble their calculations using their finger. A problem than can occur though is space, the tablet is only so big therefore there is limited room for the child to draw.

During the interviews, it became apparent that the younger pupils need to stay active. The NAO could, for example, be put in a gym or classroom and the children have to run to different corners of the room, where each corner represents an answer to the question. The NAO could also make the child jump on the answer. This combined with gamification could be a good way of keeping the child interested and give the NAO added value.

Another feature that eventually should be in the NAO, and is also the aim of IR, is to link the RekenRobot to a database. This can then keep track of how each child is doing during their sessions with the NAO. It could allow the NAO to ask questions which the child has struggled with in previous sessions or ask questions again to check that the child can still answer the questions correctly.

If, for example, the times tables task is being asked, a way to make the session more interesting is to make the session slightly competitive. If the child is told how well they have done in previous sessions, how many answers they have gotten correct, they might have the extra incentive to try and beat their high score. It could also be possible to give, for example, the top 5 scores in the class, so the child is also trying to beat their classmates' scores; this should be done at the discretion of the teacher. Emotions could also be used as a way to increase the robot-human interaction. The NAO could use a larger range of movements to bring across certain emotions, having the NAO appear to be happier when the child has gotten the answer correct and maybe determined or less happy if they have gotten an answer wrong. As said in the interview with the education expert (Appendix D.1), with children of this age, an important factor is to keep away from the 2D plane. Using cards with, for example, QR-codes which the NAO can recognise, may be a fun way to interact with the child. They would then lift the card with the answer they believe to be correct, and the NAO would then be able to respond to the given card.

What also is needed is that more tasks and explanation types need to be made, one of the overall goals would be to have all the different SLO guidelines in the system. An example for a type of explanation type that could be added is making use of objects, such as sweets. Two boxes of sweets could be shown on the tablet, representing the question. If for example the question asked is '11 + 16', then there could first be a pile of '11' sweets, then a plus, then a pile of '16' sweets. During the initialisation, the GOAL agent would then have to split both numbers into what will later become smaller piles of sweets, in piles of 20s, 10s, 5s and 1s. These smaller piles of sweets could be shown later on during one of the explanation steps. Another option is to move (by swiping on the tablet screen) sweets from one box to the other, making use of animation. This could also be done with people for the bus explanation.

At the moment, the GOAL agent keeps track of how long a session is, and when the session is finished it goes into the End state. It then lets the child know how many questions they have gotten right and then turns off. This means that if the NAO was in the middle of a question, it can finish the question but will not respond to the answer that the pupil has given. This should actually be adapted so that the last question can still be answered by the child, and that it also counts towards how many questions they have answered correctly.

The product allows the teacher to pause and play the NAO, but this can only be done by the IR system when the two are linked together. Another option could be to let the child decide if they would like a little break from the NAO, or whether they would like to continue working. They could also be given the choice of doing a different task, for example they started with addition and can choose to change to subtraction.

Other reward elements could be added. In the current system a joke is told when either ten questions are answered correctly in a row, or five if they are answered correctly quickly in succession. It also claps or kisses its biceps. Other rewards that would still have to be made could include high-fives, doing a dance, as well as additional text, for example 'Wow, you are doing really well!'. Either way, it should be more physical than it is right now. By giving a high-five, doing a dance, calling the child's name it becomes more personal and the rewards more meaningful.

Seeing that the voice recognition is not always reliable, the teacher is given the choice to turn speech recognition off in the current RekenRobot. What could be added is that if a child vocally answers a question, the number that the NAO picks up is shown on the screen. If the number that the NAO heard is different to the number that the child actually said, then the child still has the possibility to change the number on the tablet before it is sent to the GOAL agent to be checked.

During the test phase one of the children said that they would prefer that when they got the answer correct, that the NAO said the same words as shown on the tablet. At the moment, the NAO has the ability to say different phrases to let the child know that the answer is correct, while the tablet only shows the 'Well done' screen. Different screens could be created, and synchronised with the NAO.

Another suggestion made by a child during the testing was that a skip button should be added, forcing the system to go straight to the explanation phase of the question. Some of the children cannot answer the question without having, for example, the number line as aid and therefore would prefer it if the system would go straight to the number line explanation.

At the moment, the URL bar can be seen at the top of the tablet when a child is working with the NAO. It would be preferable if the bar was not visible, therefore only the UI can be seen. During the testing the children only worked with the UI, but they could very easily have opened a new tab and do something else if they wanted to.

The NAO cannot distinguish between one and multiple; the NAO may say 'There are 1 people in the bus', while instead it should say: 'There is 1 person in the bus' for the singular. This should be adapted.

The clock questions could be extended to contain 6 types of questions that can be asked: analogue to word, word to analogue, analogue to digital, digital to analogue, word to digital and digital to word. At the moment, analogue to word and word to analogue are found in the RekenRobot.

12.4 Future research

The current form of the RekenRobot was designed for children at primary schools. A future challenge will be that more advanced mathematics will be very difficult to explain using the NAO. The explanation could become very long and higher levels of mathematics are often explained better by an actual teacher.

The NAO has trouble pronouncing words and often puts the emphasis on the wrong syllable or wrong part of the sentence. This will probably be improved in the future, together with the overall flow when the NAO speaks, considering that the robot sounds very 'robotic' at the moment. With these improvements will come better interactions with the child and allow for more personalisation.

During the research phase, it appeared that there were systems that were being made specifically for children who have a handicap, disability or are less-abled. The RekenRobot can hopefully be used by everyone in the future, making distinctions between children that might need different methods of learning, as well as being adaptable for every specific need.

Other research that could be done focusses on the specific movements that works best when teaching a child arithmetic. Mimicking movements that teachers make when teaching may give extra value, as it allows the NAO to seem more human.

What could also be added in the future is the use of headphones with a microphone, so the NAO can be used in a classroom with other children without being distracting. With a microphone, there is a larger chance that the NAO will pick up on the answer that is spoken and the headphones could be plugged into the tablet, which allows only the child to hear what the NAO is saying. With the use of headphones, the NAO itself will no longer be speaking, but instead its voice will come from the tablet.

Bibliography

[1] CW. Wei, IC. Hung, L. Lee, and NS. Chen. A joyful classroom learning system with robot learning companion for children to learn mathematics multiplication. *TOJET: The Turkish Online Journal of Educational Technology*, 10(2), 2011.

[2] F. Ke and B. Grabowski. Gameplaying for maths learning: cooperative or not? *British Journal of Educational Technology*, 38(2):249–259, 2007.

[3] M. Venter and L. de Wet. Continuance use intention of primary school learners towards mobile mathematical applications. In *2016 IEEE Frontiers in Education Conference (FIE)*, pages 1–9, 2016.

[4] K. Nagaraju and S. Jain. Use of Technology for Improving Math and Science Skills of Children of Classes Primary and Upper-Primary. In *2015 IEEE Seventh International Conference on Technology for Education (T4E)*, pages 111–117, 2015.

[5] C. Clabaugh, G. Ragusa, F. Sha, and M. Matarić. Designing a socially assistive robot for personalized number concepts learning in preschool children. In *2015 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pages 314–319, 2015.

[6] H. Gelderblom and P. Kotzé. Ten design lessons from the literature on child development and children's use of technology. In *8th International Conference on Interaction Design and Children, IDC 2009*, pages 52–60, Como, 2009.

[7] T. Kanda, T. Hirano, D. Eaton, and H. Ishiguro. Person Identification and Interaction of Social Robots by Using Wireless Tags. In *2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 2, pages 1657–1664, Las Vegas, NV, 2003.

[8] M. I. Ahmad, O. Mubin, and J. Orlando. Children views' on social Robot's adaptations in education. In *Proceedings of the 28th Australian Computer-Human Interaction Conference, OzCHI 2016*, pages 145–149, 2016.

[9] A. Sandygulova, M. Dragone, and G. O'Hare. Investigating the impact of gender development in Child-Robot interaction. In *AISB 2014 - 50th Annual Convention of the AISB*, 2014.

[10] CM. Huang and B. Mutlu. Multivariate evaluation of interactive robot systems. *Autonomous Robots*, 37(4):335–349, 2014.

[11] YC. Lin, TC. Liu, M. Chang, and SP. Yeh. Exploring children's perceptions of the robots, 2009.

[12] S. Meiirbekov, K. Balkibekov, Z. Jalankuzov, and A. Sandygulova. "You win, I lose": Towards adapting robot's teaching strategy. In *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 475–476, 2016.

[13] S. Nusir, I. Alsmadi, M. Al-Kabi, and F. Shardqah. Designing an interactive multimedia learning system for the children of primary schools in Jordan. In *2011 IEEE Global Engineering Education Conference (EDUCON)*, pages 45–51, 2011.

[14] M. J. Timms. Letting Artificial Intelligence in Education out of the box: Educational Cobots and Smart Classrooms. *International Journal of Artificial Intelligence in Education*, 26(2):701–712, 2016.

[15] S. Smith, A. W. D. T. Steele, J. du Toit, and M. Conning. Development of an educational tool to teach primary school pupils multiplication tables. In *2015 Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech)*, pages 19–23, 2015.

[16] S. Lallee, E. Yoshida, A. Mallet, F. Nori, L. Natale, G. Metta, F. Warneken, and P. F. Dominey. Human-Robot Cooperation Based on Interaction Learning. In O. Sigaud and J. Peters, editors, *From Motor Learning to Interaction Learning in Robots*, volume 264 of *Studies in Computational Intelligence*, pages 491–536. 2010.

[17] J. B. Janssen, C. C. van der Wal, M. A. Neerincx, and R. Looije. Motivating children to learn arithmetic with an adaptive robot game. In *International Conference on Social Robotics*, pages 153–162. Springer, 2011.

Appendices

Appendix A Initial project description

This description is the original project description as found on BEPSys.

Project description:
'For our product, robots in de klas, we are developing a robot application that can train calculation skills for primary school children. Key is a very robust interaction with the child as well as adaptive difficulty. Students wil focus on the implementation of the robot logic and child robot interaction using our existing platform. Behaviors/sensors etc. are dealt with, so the focus is on high-level interactions. Content for the reken robot will be develeoped by others in principle.'

Company description:
'Interactive Robotics is developing a Robot Interaction Engine, using cognitive technology. With robots quickly becoming more able to assist us, human robot interaction is the next big challenge that needs to be solved for robots to successively enter into our society. Robots not only need to perform tasks for us but need to do so in a way that makes sense to us.

This requires robots with the social intelligence to understand us, robots that have natural interaction capabilities to talk with us, and robots that are able to adapt to us. Our Robot Interaction Engine enables you to quickly develop interactive scenarios for your robot application. Whether it's a robot host, robot waiter, a care robot, or a robot teaching assistant, Interactive Robotics' solutions deliver an optimal interaction with people.'

Appendix B Search queries

B.1 Scopus

```
TITLE-ABS-KEY ((child OR children) AND (education OR learning OR calculating OR primary
    AND school) AND (robot OR robotics))
```

B.2 Web of Science

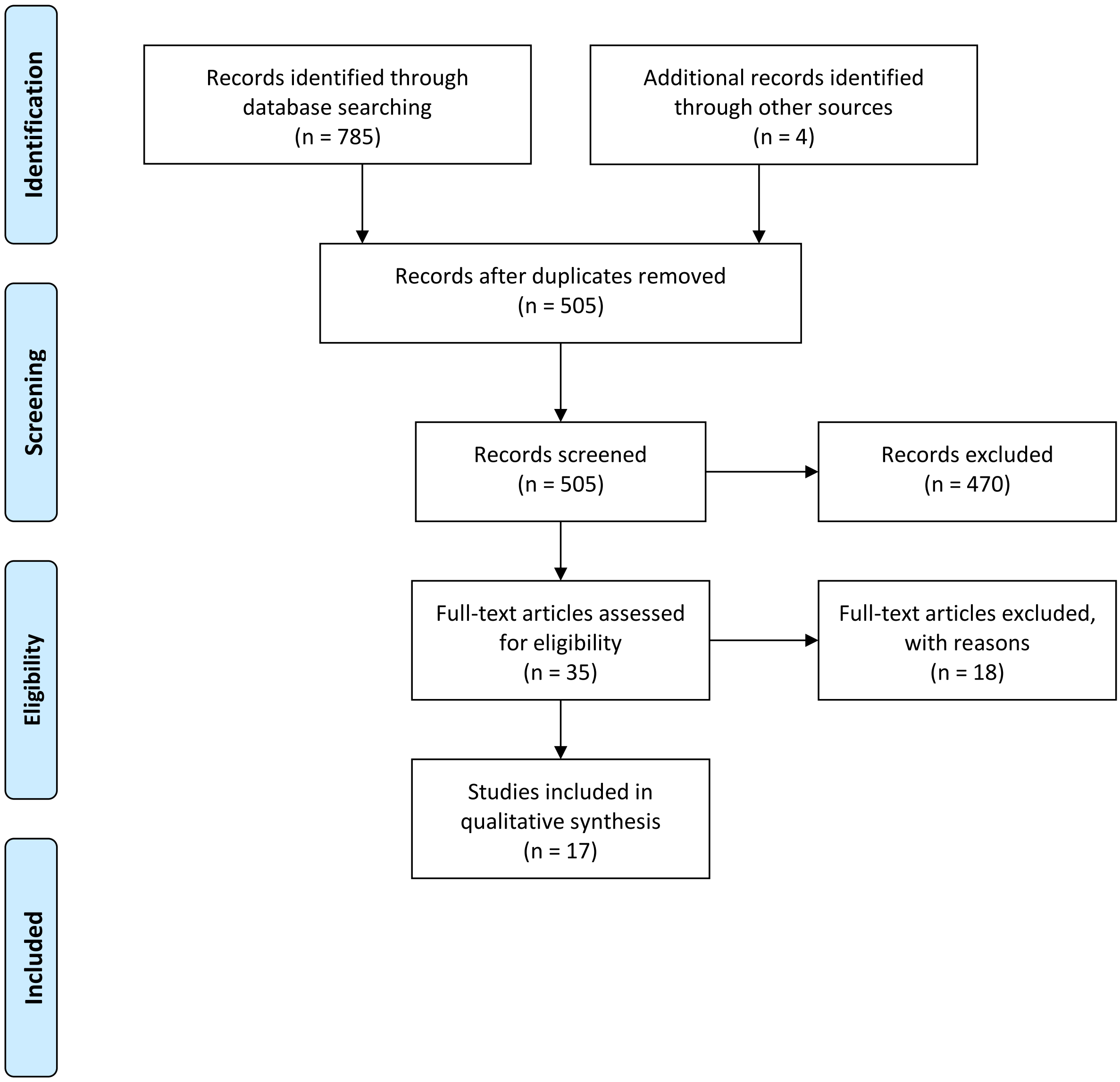
```
((child OR children) AND (education OR learning OR calculating OR primary AND school)
    AND (robot OR robotics) NOT autism NOT disability NOT disabilities NOT illness NOT
    health)
```

B.3 IEEE XPlore Digital Library

```
((child OR children) AND (education OR learning OR calculating OR primary AND school)
    AND (robot OR robotics) NOT autism NOT disability NOT disabilities NOT illness NOT
    health)
```

Appendix C PRISMA-diagram

PRISMA 2009 Flow Diagram



From: Moher D, Liberati A, Tetzlaff J, Altman DG, The PRISMA Group (2009). *P*referred *R*eporting *I*tems for *S*ystematic Reviews and *M*eta-*A*nalyses: The PRISMA Statement. PLoS Med 6(6): e1000097. doi:10.1371/journal.pmed1000097

For more information, visit www.prisma-statement.org.

Appendix D Interviews with an educational expert and teachers from primary schools

D.1 Interview with an educational expert

This section contains the reporting of the interview with an educational expert from the company Bien4Kids[1]. Her ideas, as to what could be done with a NAO robot as an educational tool for basic arithmetic, is valuable to this project. From her previous experience with the NAO, she had an idea of the possibilities. The age group in question was the age when children start learning arithmetic in the Netherlands, initially starting with addition, around the age of 6 to 8.

Quickly it became apparent that the educational expert believed that the robot's way of teaching needs to try and stay away from the 'flat plane'. She believed that children can have difficulty transitioning from physical objects to the 2D versions of the concepts on paper or on the tablet. In the Netherlands, children start learning to count in the first and second year of primary school. In these two years, the children are mainly taught to count with physical objects, making use of blocks by counting them and grouping them together. At this age, the children are already learning the very basics of arithmetic. An example that she gave was a scenario where in the classroom she would place 3 towers, each tower containing 2 blocks and asked the children to say what they saw. She would then rearrange the blocks into 2 towers of 3 blocks, to show that the same amount of blocks in different arrangements could still give the same total. This allows the children to get an initial feel of the 2D plane, as well as a feel for multiplication. She also said that it was very important to let the children do as much as possible themselves, while keeping away from 2D.

Gamification was considered a good method, but it has to keep the children active. She also suggested trying to get the NAO to depict numbers with its arms, to give a visual depiction of the numbers. Children around the age of 5 to 8, find physical object and images or pictures a very valuable way of learning. The sensors of the NAO could also be used as a way to answer questions. For example, the NAO could give a sum and have each foot and arm represent a possible answer. The child would then have to press the correct bumper to give the answer. A problem with this is that in the child's enthusiasm they could cause the NAO to fall over, considering that it is not incredibly stable on its feet. Another problem with this approach would be that the child would have to remember the choices which NAO has given to them, therefore the number of choices would have to be limited. The pupil would also have to remember which arm or foot corresponded to which answer.

The key point that the educational expert made was that the methods to teaching need to stay interactive. The children need to stay active, and should actually sit down as little as possible. Another idea that she had was that the NAO could be put in a gym hall where, for example, 4 (coloured) squares are put at different corners of the hall. NAO could then ask a sum, and say which corner represents which answer. The children could then run to the corner that represented the answer that they believe to be correct.

The word 'wrong' or 'incorrect' needs to be avoided. A positive atmosphere needs to be kept, and the children need to be encouraged. If a child gives the wrong answer, it is better to say: "That is almost correct, try again!" than "Unfortunately that is wrong.". Therefore a number of standard positive phrases need to be devised, and randomly chosen. There should be many different phrases, otherwise the robot will sound repetitive. If a child makes a mistake, the robot could either give them help or the choice could be made to let the child try again. A system could be build that allows a child to, for example, try answering 3 times and if they still give the incorrect answer then have the NAO help the child to try and reach the correct answer. Phrases also need to be made for correct answers. In addition to phrases, there could also be sounds and songs that play when a child answers correctly. The educational expert also mentioned that a build-up could be implemented, for example if a child gets 5 questions correct in a row then they get a high-five from the NAO, for 10 correct answers the NAO could do a dance. The attention span of a child is not very long, hence a 10 minute session with the NAO is already more than enough for a young child.

Another point made was that it could be important to obtain the foreknowledge of a child. The NAO could use the questions that it asked a child in previous sessions, to test if the child

[1] `http://www.bien4kids.nl/`

has remembered what they have learned and to check if the NAO is asking questions of the correct level of difficulty. The NAO needs to be able to differentiate between different levels as some children struggle more with calculations than others. The teacher could input the level they believe the child to be at, and the robot could adapt the difficulty throughout the sessions as needed. Thus if the child is struggling, then the NAO could make the questions slightly easier and if the child is getting everything correct then move them up a level. The NAO can then record the answers that the children have given, through both checking if the child answered the question correctly as well as the speed at which they answered the question. In general, the quicker the answer (given that the answer is correct) the better the child is at that level. The names of the levels should also be given in a discreet way, this should be discussed with the individual schools where the NAO will be put to work. The pupils should not feel they are worth less due to the fact that they are in a lower group. In general, primary schools have three levels within the classroom. It is of course possible to add extra levels if needed. If there are children that are ahead of the rest, they could be challenged by asking questions in a higher group or to give them less time to answer questions.

D.2 Interview with teachers from 'De Voorsprong' in The Hague

In the following section, the results of an interview that was conducted at one of the pilot primary schools of Interactive Robotics will be described. This school will start using some of Interactive Robotics' products during the course of the following year. Consequently, these teachers were very willing to help realise a valuable product. Present at the meeting were 3 teachers that teach the target audience of this project. This means that they teach children that are starting with arithmetic education, around the age of 6. Additionally, some individuals from the school management attended the meeting.

The interview began with an explanation of the project, which was met with a lot of enthusiasm. It was thought that the robot could be a great addition to the education provided by teachers. Teachers are often overwhelmed by different types of pupils, who need different instructions. The group is often divided in three; the slower pupils, the faster pupils and the middle group. When asked which group was the best target for the resulting application of this project, the answer was clearly all three of them. The teachers would like the flexibility to be able to sending any child, from any level, to the robot so that they can focus on the other groups.

An excellent point that was made by the teachers was that some children might feel left out if the robot would only be used for one specific group. On the other hand, the teachers also observed that if only the excelling pupils are allowed to use the robot, this resulted in some children trying extra hard to be able to get access to the robot. Sadly, when the pupils did not manage this, they would get frustrated. The tactic should thus be used with great caution. When instructing pupils who are excelling, the teachers differentiate by letting them skip the easy questions. Additionally, they ask the pupil to work with larger numbers, but refrain from making the pupils 'carry' numbers, for example 24 + 11 as this goes into the next 10's. A problem that sometimes occurs is that there is a limit as to how high an excelling child can go. If they are provided with the curriculum of higher age groups, the problem becomes that the material is very mathematical, resulting in the pupils needing extra instruction, which the teacher do not have the time for.

The next item of discussion were the group exercises. As discussed in previous sections, it is well established that group exercises, particularly physical exercises promote learning. An exercise done at this particular school was letting the children solve questions on a small white board and holding that board up when they have the answer. Making the exercises a competition seems to be a common theme. Additionally, creating a physical number line on the ground and having the pupils jump on a number to indicate an answer, was also used. Lastly, having a child come up with a question, then throwing a ball to a fellow pupil and the pupil who catches the ball must answer the question was also an exercise that was used. The teachers believe that having the robot be a part of these types of games is a welcome addition. Having the child see the robot as buddy is certainly desired.

When designing the exercises, both single and group exercises, the attention span was crucial information. The answer was that 1 to 2 minutes is reasonable to ask of a child. An example given was that they let the pupils race the teacher, to see who could complete more questions in one minute. The teachers mentioned that, even if they do not beat the teacher, when they try again the next time they are delighted to see their own progression. This led to the topic of letting the system keep track of which question the pupils found easy and maybe even how much time they spent on a particular question. The teachers thought that recording such information is certainly helpful, but did not have a clear opinion about what is relevant and what is not. This will become more clear in the testing phase. There was a remark that a pupil should also be able to indicate when he or she does not understand the question. In such a case they might be able to do the arithmetic, but the problem is in the language and reading aspects. The teachers would like to know when this occurs.

The teachers did not foresee any problems with the pupils working with the tablet; they would not leave the children alone with the hardware, but would put them in the corner of the classroom. The teachers noted that it would be extremely useful to use a headset as not to distract the other children, and also a microphone to answer the questions as a full classroom can be noisy. The voice of the robot did not matter very much to the teachers. They suggested to let it use the default boyish voice.

The idea to offer a multitude of different ways of explaining a question was met with enthusiasm. The teachers noted that is it important to be consistent in explanations to one child, but the option to switch methods when one does not work, was very welcome.

When using clocks in education these teachers were limited to the quarters and maybe even 5 minutes past or to halves. They made it clear that they consistently start with analogue clocks. Therefore only a limited number of question types should be asked of the pupils when it comes to clocks.

Personalisation through using the name of the pupil is fun, but names can easily be mispronounced therefore the advice was to avoid using names, or to use a system where voice sounds can be input to give the NAO the ability to pronounce the names properly.

When asked about the use of difficulty levels, and the sensitivity that sometimes is needed, they remarked that children are aware of their own progress. They know who are the smarter children in a classroom and where they stand. At this school they obfuscate levels by giving labels such as rockets, star and moons.

The option of using the physical buttons on the robot is also certainly possible. The teachers think that if they were to explain to the children that they have to use the buttons gently, then they will be careful. The idea of the robot giving a high-five is also great for the children, according to the teachers. The teachers would like the option to customise the experience. However, these customising options should never be at the expense of the clarity and easy use of the default settings. The teachers mentioned that some online platforms give jokes as a reward for a completed exercises. They mentioned that the children often react very well to this.

Overall the teachers were enthusiastic and were curious to test the solution of this project. They looked forward to the testing phase, and welcomed the idea of getting support in the classroom in the form of robots.

D.3 Interview with a teacher from 'De Gantel' in The Hague

This section reports the interview with a teacher from the primary school 'De Gantel' in The Hague. This interview was very similar to the interview conducted at the 'De Voorsprong' primary school. Roughly the same questions were asked to validate the beliefs that were formed from the first interview.

When asked which types of pupils she would like to use the robot for, the response was that it would be especially great to have something for the excelling pupils, considering the lack of means to deal with such pupils. She often had too much on her plate, so being able to set aside the more advanced pupils with the robot would help a great deal. However, having capabilities for all types of pupils would be ideal.

When asked how the lessons could be more three dimensional, the teacher suggested having the robot or tablet instruct the children to do exercises with blocks. The teacher believed that using the robot in such exercises would generate a lot of excitement. Using a tablet was considered a good way to help visualise a question, she believed that the use of tablets in conjunction with the robot will pose no problems to these children.

Providing each task at multiple different levels was met with enthusiasm. Different pupils work at different speeds, having multiple levels in the material will help giving each pupil what they need. The teacher did not see the need to obfuscate the order in levels. The pupils will simply have to learn that not everyone is at the same level.

The teacher did believe that every pupil should be able to work with the robot and have access to all material. Having the system switch the pupil between difficulty levels based on how the learning is going, is an essential feature. When presented with the ability to add a database, the teacher believed that recording how many right or wrong answers have been given, as well as how long it took the pupil, would be very helpful.

When the teacher wants to help the pupils to develop a sense for arithmetic, exercises based on stories are often used. The robot might be an excellent tool in telling such stories. The amount of time an exercise should take depends on the type of question. When attempting to 'automate' calculations, the session should be very short because the pupil has to answer a lot of questions in a short time frame. With the story questions on the other hand, the teacher said that 10 minutes could be an appropriate time. The teacher believed that not every exercise has to be fun. The children have to learn to focus and work hard as well.

When it came to having the robot in the classroom, the teacher believed that after a short adjustment period, the robot will not be a distraction to the other pupils who are not using the robot.

A great suggestion that was brought forth during the interview, was that it would be ideal if the pupils had a button on the tablet to repeat the previous step or explanation. Children can easily miss something and sometimes have problems with reading the question. Having the ability to repeat the explanation from the robot, would be very valuable. The teacher also said that having a pausing ability is extremely desirable.

D.4 Interview with a teacher from 'De Notenbalk' in Zwijndrecht

This section reports the interview with a teacher of the last grade in primary school. This interview was held to get a sense of the possibilities of using the solution proposed in this paper for older children. To look at the wider scope, and to keep in mind that this product will in the end not only be for pupils aged 8-9.

At the start of the interview it became apparent that the last grade does not handle much new information. The last year is mainly meant to repeat and consolidate knowledge. This means that it might be worthwhile to focus on exercises instead of explanations for these older children. The teacher mentioned that the robot might be a great fit for exercises based on stories, from which the children have to select the relevant information. Having the robot tell the story might improve engagement. For these older children, there is not a lot of focus on making things three dimensional or physically active.

For excelling pupils, exercises could be made harder and they could be given less time to complete the question, as it is not be in the scope of this project to start moving towards mathematics. Assisting the weaker pupils mainly consists of having the children explaining how they do the exercises and correcting mistakes they make in this process.

It might be appropriate to have the robot assist by letting it ask the children to do a small presentation about what they have learned and having the teacher guide this interaction.

One of the more difficult parts for this particular age group is geometry. The teacher believed it could be fun if the robot would instruct the children to measure something in the classroom, for example an aquarium, and then having the robot explain how to calculate the volume or surface area.

The teacher mentioned that one of the advantages of digital platforms is that pupils can only see one exercise at a time. Especially for pupils with attention problems, this can help them focus on the question at hand. Additionally, for the older children, the attention spans are not significantly longer.

The teacher noted that it might be an idea to, after a pupil has had a session with the robot, get them to go to another pupil in the classroom and notify them that it is their turn. This way the teacher can focus on other things.

These older children already start slacking off sometimes, so having the robot say something when they are not engaging might be very useful. Actions like high-fives or dancing should be used sparsely.

Appendix E Background information: Long multiplication

For long multiplication questions, using the SLO core objective 30, the following holds per age group:

- Year 5/6
  - Introduction to the multiplication of a one digital number with a multiple digital number, making use of a column method.
  - Expansion of multiplication to a multiple digital number multiplied with a multiple digital number, making use of a column method.
  - Introduction of the number procedure used for multiplying a multiple digital number with a multiple digital number, using a simplified column method.
- Year 7/8
  - Introduction of the number procedure used for multiplying a multiple digital number with a multiple digital number
  - Further practice of the column method and the number procedure for multiplication.

The ontology flow that would accompany the long multiplication question can be seen in Figure 12.1

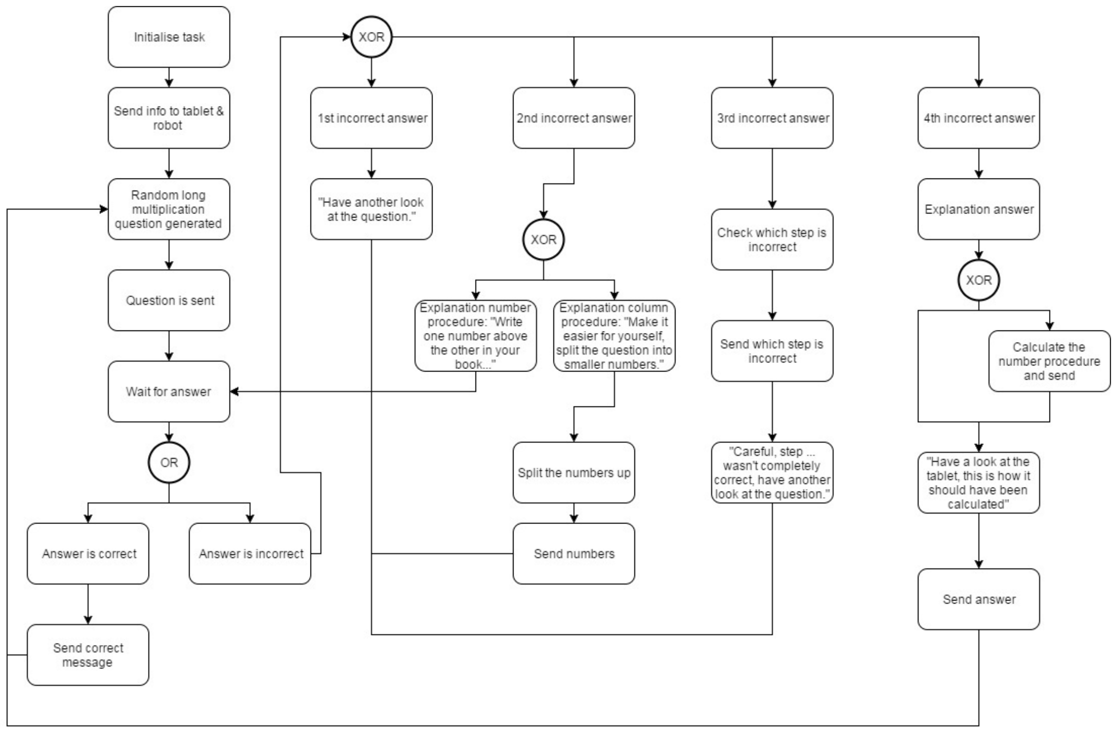


Figure 12.1: The GOAL agent flow diagram for a long multiplication question

Appendix F Product design and implementation: Long multiplication questions

F.1 Introduction

When it comes to the multiplication questions, the NAO will originally start with: ‘Today we will be practising multiplication’. The multiplication sessions are always 10 minutes long, and the child can always ask the NAO for help.

F.2 Initialisation

The boundaries for the long multiplication questions are decided before hand by the teacher. This means that:

- A random question needs to be generated, where every number is bounded; random(1,Bound1,A), random(11,Bound2,B) Answer is A $\times$ B.
- For the eventual column procedure, both values need to be split into 1’s digits, 10’s digits, 100’s digits and 1000’s digits. For example, 273 is split into 200, 70 and 3.
- For the eventual number procedure, the intermediate steps need to be calculated. For the example question ‘895 $\times$ 4’, then the answer is calculated by: $5 \times 4 = (2)0$, $9 \times 4 + (2) = (3)8$, $8 \times 4 + (3) = 35$, therefore the answer is 3580. This is done in the format as seen in Figure 12.2. This can only be done in this form if a number is multiplied by a one digit number.

$$
\begin{array}{ccccc}
 & \mathit{2} & \mathit{4} & & \\
 & 5 & 3 & 8 & \\
 & & & 6 & \times \\
\hline
3 & 2 & 2 & 8 &
\end{array}
\qquad
\begin{array}{ccccc}
 & \mathit{3} & \mathit{6} & & \\
 & 9 & 3 & 7 & \\
 & & & 9 & \times \\
\hline
8 & 4 & 3 & 3 &
\end{array}
$$

$$
\begin{array}{ccccc}
 & \mathit{4} & \mathit{5} & & \\
 & 2 & 4 & 6 & \\
 & & & 9 & \times \\
\hline
2 & 2 & 1 & 4 &
\end{array}
\qquad
\begin{array}{ccccc}
 & \mathit{1} & \mathit{1} & & \\
 & 8 & 3 & 2 & \\
 & & & 6 & \times \\
\hline
4 & 9 & 9 & 2 &
\end{array}
$$

Figure 12.2: Four examples of long multiplication using the number procedure, where a number is multiplied by a one digit number. The numbers in italics are the ‘carried’ values.

The levels that can occur for the long multiplication questions can be found in Table 12.1.

Level	Boundary	Example	Notes
1	One digit number $\times$ Double digit number	$6 \times 48 = 288$	One digit number = 0-9, Double digit number = 10-99
2	One digit number $\times$ Triple digit number	$7 \times 234 = 1638$	One digit number = 0-9, Triple digit number = 100-999
3	One digit number $\times$ Quadruple digit number	$3 \times 1235 = 3705$	One digit number = 0-9, Quadruple digit number = 1000-9999
4	Double digit number $\times$ Double digit number	$24 \times 31 = 744$	Two digit number = 10-99, Double digit number = 10-99
5	Double digit number $\times$ Triple digit number	$16 \times 156 = 2496$	Two digit number = 10-99, Triple digit number = 100-999

Table 12.1: The different levels for the long multiplication questions.

F.3 Input

Originally, the only input will be the answer. If the child has gotten the answer wrong, they are then given the ability to fill in the intermediate steps, a very basic view of this is found in Figure 12.3. Each of the boxes can then be filled in, which then becomes the new input, and the GOAL agent can check that all the values are correct, or else hint at which boxes have the wrong number in them.

$$\begin{array}{ccccc} & \square & \square & & \\ & 5 & 3 & 8 & \\ & & & 6 & \times \\ \hline \square & \square & \square & \square & \end{array}$$

Figure 12.3: In this example, the square boxes will be where the child can click to fill in that specific box, for the number procedure.

F.4 Explanation step 2

For the column procedure, the NAO will say that it has split the numbers into smaller questions, and ask the pupil to calculate each of the individual questions. When they have done that, they can add each of the individual answers, to get the overall answer.

For the number procedure, GOAL agent will show the numbers above each other on the tablet. The child then has to multiply step by step, the top value with the bottom value. If the sub answer is larger than 9, then the remainder can be written above the next number.

F.5 Explanation step 3

The intermediate steps are checked by the GOAL agent. For both the number and column procedures, the GOAL agent can let the child know which values they have gotten wrong, and explain how and why.

F.6 Explanation step 4

The NAO explains step by step how the answer should have been reached, and gives the correct answer.

Appendix G SIG 1st feedback (in Dutch)

'De code van het systeem scoort 2,5 ster op ons onderhoudbaarheidsmodel, wat betekent dat de code ondergemiddeld onderhoudbaar is. De hoogste score is niet behaald door lagere scores voor Unit Size en Duplication.

Voor Unit Size wordt er gekeken naar het percentage code dat bovengemiddeld lang is. Het opsplitsen van dit soort methodes in kleinere stukken zorgt ervoor dat elk onderdeel makkelijker te begrijpen, te testen en daardoor eenvoudiger te onderhouden wordt.

Jullie doen dit nu heel wisselend. In sommige bestanden heeft elke functie een duidelijke verantwoordelijkheid, maar in andere bestanden staat alles door elkaar. Het bestand clock.js begint bijvoorbeeld met een reusachtige functie. Met name het verschillende gedrag afhankelijk van de huidige taak zou je veel beter naar aparte methodes kunnen verplaatsen.

Voor Duplication wordt er gekeken naar het percentage van de code welke redundant is, oftewel de code die meerdere keren in het systeem voorkomt en in principe verwijderd zou kunnen worden. Vanuit het oogpunt van onderhoudbaarheid is het wenselijk om een laag percentage redundantie te hebben omdat aanpassingen aan deze stukken code doorgaans op meerdere plaatsen moet gebeuren.

In jullie project komt in het bestand numberLine.js bijvoorbeeld vier keer hetzelfde stuk code voor. Het zou beter zijn om de gedeelde code naar een nieuwe methode te verplaatsen, en deze methode vervolgens aan te roepen.

Als laatste nog de opmerking dat er geen (unit)test-code is gevonden in de code-upload. Het is sterk aan te raden om in ieder geval voor de belangrijkste delen van de functionaliteit automatische tests gedefinieerd te hebben om ervoor te zorgen dat eventuele aanpassingen niet voor ongewenst gedrag zorgen.'

Appendix H SIG 2nd feedback (in Dutch)

'In de tweede upload zien we dat zowel de omvang van het systeem als de score voor onderhoudbaarheid is gestegen. Door grote verbeteringen op het gebied van Unit Size en Duplication zijn jullie gestegen naar 3 sterren, en daarmee nu marktgemiddeld onderhoudbaar.

Ook hebben jullie unit tests toegevoegd. Het aantal is nog erg klein, maar jullie hadden natuurlijk een achterstand op dit gebied. Voor de toekomst is het wel handig om direct te beginnen met het testen van de belangrijkste functionaliteit.

Uit deze observaties kunnen we concluderen dat de aanbevelingen van de vorige evaluatie zijn meegenomen in het ontwikkeltraject.'

Appendix I Infosheet

Project title:
RekenRobot: Assisting primary school teachers in basic arithmetic education.

Date of presentation:
4th July 2017

Project Description:
The description of the project was to develop a system that uses the NAO-robot, which assists the primary school teachers in teaching basic arithmetic. The challenge however, was to develop the system in such a way that it can be deployed in the existing system of our client. This platform is not yet a finished system, therefore a significant part of the project was trying different architectures and designs that would suit the client in the future. The research that was conducted was focused on the didactic aspect of this project. How could young children be taught to do arithmetic. Additionally, what are the unique selling points of using the NAO in this process. The client asked to use an agile development strategy in line with what had been used in the past by the client. This included planning and documentation on JIRA and Confluence. The product that was created uses a tablet together with the NAO. Additionally, a web interface for the teachers was created. The teacher can assign a task to a pupil and select the difficulty and what type of additional explanation will be used. The children can then answer arithmetic questions on either the tablet or vocally. The tablet also functions as a way to offer visualisation and explanation. In the future, the RekenRobot-application will be deployed on the system of our client. The supply of different lessons and ways of teaching will be expanded. Extra features should also be included like using the vision-system of the robot and improving the speech-interface.

Client:
Dr. Ir. D. J. Broekens, CTO of Interactive Robotics

Coach:
Dr. K. V. Hindriks, Associate Professor at TU Delft

Team members:
J. de Boer, J.deBoer-2@student.tudelft.nl
Interests: Computer Science Education, Back-end development.
Description of contributions: Agent programming and Didactic research

E. de Bree, E.E.M.deBree@student.tudelft.nl
Interests: Computational Finance.
Description of contributions: Front-end development, Lead on paper.

P. Remeijsen, P.Remeijsen@student.tudelft.nl
Interests: Data science and front-end Development.
Description of contributions: Front-end development, Agent programming.

M. Verzijl, M.A.R.C.Verzijl@student.tudelft.nl
Interests: Application of CS in the medical field.
Description of contributions: Front-end development, Messaging systems.