Universidade Nova de Lisboa
Faculdade de Ciências e Tecnologia
Departamento de Informática

Dissertação de Mestrado

Learning with Tangible Interfaces

Laetitia dos Reis e Silva Mendes (aluna nº 26508)

Orientadora: Professora Doutora Teresa Romão

Trabalho apresentado no âmbito do Mestrado em Engenharia Informática, como requisito parcial para obtenção do grau de Mestre em Engenharia Informática.

2º Semestre de 2008/09
29 de Julho de 2009

Acknowledgements

First of all I would like to thank Teresa Romão for the constant support and amazing trust in my work capacity.

I would also like to thank my parents and sister for the patience during this very stressful year.

For the availability and support I thank José Carlos Danado and Emanuella Mazzone(who was a key help in the design sessions planning and analysis).

To the Interactive Multimedia Group (IMG) members, whom help was essencial in the last few months.

To Ilídio Nunes, who was a constant presence and showed a great support and availability to every phase of this work's development.

And last but not least, to all my friends who were there and helped through all the process, especially to João Pedro Marques and Pedro Lobo for their help in the design meetings and final tests.

Resumo

A tecnologia é já parte activa das nossas vidas e, mesmo sem nos apercebermos, parte das nossas actividades diárias tornaram-se já dependentes dela. Por esse motivo, os criadores de software começaram a dar uma atenção especial às necessidades dos utilizadores e à usabilidade das interfaces, com vista a melhorar a interacção dos utilizadores com o *hardware* e o *software* a utilizar.

As crianças são um grupo de utilizadores em crescimento, uma vez que são confrontadas com a tecnologia desde uma fase inicial do seu desenvolvimento. Sabendo que as crianças vêem o mundo de uma forma diferente da dos adultos e não têm ainda a destreza necessária para interagir com alguns dispositivos físicos, surgem preocupações muito específicas em termos de usabilidade infantil. Isto acontece especialmente se a aplicação tiver uma finalidade educativa, já que as crianças são mais propensas a necessitar de motivação extra do que os adultos. Por isto, surgiu um novo campo de estudo dentro do âmbito da Interacção Pessoa-Máquina, cujas preocupações se orientam especificamente para as crianças e para o modo como as mesmas se relacionam com a tecnologia, denominado Interacção Criança-Máquina.

Ao criar tecnologia direccionada para crianças, o conceito de ubiquidade surge quase naturalmente, visto que seria perfeito que as mesmas pudessem interagir com o software sem se aperceberem desse facto. Isto pode ser alcançado se as interacções se basearem em objectos reais do dia-a-dia e acções que lhes forem familiares.

O objectivo desta tese é criar uma ferramenta que permita às crianças criarem os seu próprios jogos educativos, baseando a interacção em objectos físicos do dia-a-dia. Esta ideia baseia-se num método de ensino que dá às crianças o papel de professores (método *Learning-by-Teaching*).

Investigadores descobriram que a melhor forma de criar aplicações para crianças é incluí-las no processo de construção. Tendo por base o método *Bluebells*, foram assim realizadas três reuniões de desenho com as crianças, com o objectivo de compreender o que é, para elas, uma aplicação intuitiva.

Uma vez tendo o protótipo desenvolvido foram realizados testes de usabilidade à aplicação por forma a não só estudar a sua usabilidade, mas também perceber se o facto de construirem o seu jogo motiva as crianças a aprender mais sobre o assunto à volta do qual o jogo é criado.

Palavras-chave: interacção criança-máquina, sistemas pervasivos, interfaces tangíveis aumentados, jogos educativos

Abstract

Technology is an active part of our lives and, without even noticing it, part of our daily activities became dependent on it. For that reason, software constructors began to pay special attention on people's needs and interaction with both hardware and software they must deal with.

Children are an emergent users' group, as they are confronted with technology from an early stage of their development. Knowing that children see the world in a different way adults do and haven't got yet the necessary dexterity to interact with some physical devices, special concerns arise. This happens especially if the application has an educational purpose, because they are more likely to need an extra motivation to use it than adults. Given that, a new subfield of Human-Computer Interaction appeared with special concerns related to children's applications and how they interact with them: Child-Computer Interaction.

When creating children's technology the concept of ubiquity seems to rise almost naturally. The idea of children interacting with technology without even noticing it seems perfect. This may be achieved if the interactions are based on everyday objects and actions children are used to.

The purpose of this thesis is to create a tool that enables children to build their own educational games, based on physical objects with which they usually interact. This idea follows a Learning-by-Teaching approach in which children are given the instructor's role.

Researchers have found that the best way to create children's software is to let them take an active part on the construction process. Bearing that in mind three design sessions were conducted with children, based on the Bluebells Method, so they could give us the insight needed to create an intuitive application.

Finally, usability tests were made to the created prototype in order not only to study its' usability but also to understand if children's motivation to create their own game engages them into learning more about the application's subject.

Keywords: children computer interaction, pervasive systems, tangible augmented interfaces, educational games

Contents

1 Introduction **1**

1.1 General Context 1

1.2 Problem Description 2

1.3 Solution Approach 4

1.4 Main Expected Contributions 6

1.5 Document Organization 7

2 State of the art **9**

2.1 Cognitive Development 9

2.1.1 Piaget and Vygotsky 9

2.1.2 Motivation, Goals and Learning 10

2.1.3 Learning by teaching 12

2.2 Human-Computer Interaction 12

2.2.1 Children are not adults 14

2.2.2 Gathering the requirements of children applications 17

2.2.3 Methods to include children in the design process 17

2.2.3.1 Cooperative Inquiry 18

2.2.3.2 Bluebells Method 19

2.2.3.3 Warp Speed Design 21

2.2.4 Gathering the children's opinions 22

2.2.5 Computers' role in education 23

2.3 Pervasive Systems 25

2.3.1 Tangible Augmented Reality 27

2.3.2 ARToolKit and derivates 30

2.3.3 The TUIO Protocol and ReacTIVision 31

3 Methodology - Design 35

3.1 Initial Technology Study and Concept 35

3.2 Design Meetings with Children 40

3.2.1 First Session - Tig 41

3.2.1.1 Before the session 41

3.2.1.2 During the session 43

3.2.1.3 After the session 44

3.2.2 Second Session - Blind Man's Bluff 47

3.2.2.1 Before the session 47

3.2.2.2 During the session 49

3.2.2.3 After the session 49

3.2.3 Third Session - Warp Speed Design 50

3.2.3.1 Before the session 50

3.2.3.2 During the session 52

3.2.3.3 After the session 52

3.3 Meetings' Results 53

3.4 System Architecture 55

4 Methodology - Implementation 59

4.1 Table 59

4.2 Game Creation Application 60

4.2.1 Configuration File 60

4.2.2 TUIO Protocol and reacTIVision 62

4.2.3 Flash Communication 64

4.2.4 Application Flow 67

4.3 Game 68

4.3.1 Configuration File e Application Flow 68

5 Tests, Results and Analysis 75

5.1 Behind the tests 75

5.1.1 Demonstration of a previously created game 76

5.1.2 Preparation of the story and questions/answers 76

5.1.3 Creation of the game 77

5.1.4 Play the created game 77

5.1.5 Questionnaire 78

5.2 Testing and Results 79

5.2.1 Preparation of the story and questions/answers 79

5.2.2 Creation of the game 80

5.2.2.1 Introduction and Story 80

5.2.2.2 Character 81

5.2.2.3 Scenario 81

5.2.2.4 Questions 82

5.2.3 Questionnaire 83

5.3 Results' Analysis 83

6 Final Considerations and Future Work 89

6.1 Final Considerations 89

6.2 Future Work 91

List of Figures

1.1 Application phases 5

2.1 Software Design Cycle 13

2.2 Boehm's Spiral Model, from [6] 14

2.3 Four main roles a child can play in the technology design process. 17

2.4 Contextual inquiry notes by a 7-year old child, from [14]. 19

2.5 The Bluebells method activities 21

2.6 A Smileyometer awaiting completion, from [32]. 23

2.7 A Completed Fun Sorter showing how children position the picture cards in the boxes, from [32]. 23

2.8 An excerpt from a Completed Again Again table that was being used to compare different word processing packages, from [32]. 24

2.9 Milgram's virtuality continuum, from [28] 28

2.10 HITLabNZ's explanation of how AR works, from [20]. 30

2.11 Using the MagicBook interface to move between Reality and Virtual Reality, from [5]. 31

2.12 reacTIVision diagram, from [22] 33

2.13 Amoeba fiducials, from [22] 33

2.14 Reactable 34

3.1 Cardboard 42

3.2 Game Cards 43

3.3 Photos from the first design session 44

3.4 Cardboard of one of the children, when the activity was finished 44

3.5 The different game parts, and their order 45

3.6 The different input "devices" and the children's choices 46

3.7 The different character's features and their importance according to the children that participated in the activity 46

3.8 The characters that children chose 47

3.9 Children's answers to the questionnaire 48

3.10 The concept chosen from the application. 51

3.11 Children testing the prototype at the third design meeting 52

3.12 System Flow 55

3.13 Game Creation Application Architecture 56

3.14 Game Architecture 57

4.1 The table construction 60

4.2 Game Creation Application Schema 63

4.3 Schema of the general features that must be defined for each phase (introduction phase presented as example) 64

4.4 Game Creation Application Classes 70

4.5 Screenshots from the construction phases 71

4.6 gameClasses 72

4.7 Game Configuration File Schema 73

5.1 The children preparing their game 76

5.2 Game creation flow and scenarios with tasks children had to accomplish 77

5.3 Children creating their game 78

5.4 Introduction and Story Usability Results 80

5.5 Character Usability Results 81

5.6 Scenario Usability Results 82

5.7 Questions and Answers Usability Results 83

5.8 Questionnaire results 87

List of Tables

2.1 Different roles a child can play in an application development process, based on Table 2 of [13] 16

2.2 Semantic types of set messages, from [23] 32

3.1 Cognitive Requirements, based on Tables 1 to 4 of [8] 36

3.2 Physical Requirements, based on Tables 5 and 6 of [8] 37

3.3 Social/Emotional Requirements, based on Tables 7 to 9 of [8] 37

3.4 Game phases and their respective input methods 38

1. Introduction

1.1 General Context

Learning is an intrinsic part of life and rarely a day goes by without the acknowledgment of something brand new. Our cognitive development is based on what we learn and a great part of what defines us as adults is based on the education we have received as children. Therefore, today's society should grant a greater importance to children's intellectual development since it will define the future of adulthood.

For many years a special attention has been paid to children's cognitive development in order to improve their learning process, and even though there are many different factors contributing to that process, motivation is a key ingredient. For this reason, every method created with the purpose of helping children's development, must be engaging enough to make them want to learn.

Technology is becoming more present in our everyday's lives. This is more true regarding children behavior changes. Due to early developments of computer sciences and electronic fields, children not only deal with technology at home, when watching television or playing with a game console, but also at school, since it is becoming more usual that technology takes a great part in their cognitive development.

Computer based education represents a major challenge when directed to children, given that they are not yet completely formed, both physically and intelectually. This fact obliges the constructors of children's technology to start thinking in a different way that they would if constructing for adults: not only the software has to be more intuitive and motivational, but also the hardware should be adapted to correspond to children's little dexterity and natural clumsiness.

For these reasons, pervasive technologies started being explored in order to enhance the children's learning process. Seeing that it is not supposed that children pass through a large learning period in order to be able to use a computer application, when creating software one should take into account that it must always be as intuitive and natural as possible. Ideally children wouldn't have to learn to use it at all. It is also relevant that children have fun when using computer applications.

This work's purpose is to develop an educational game authoring tool for children and to study if the perspective of doing so raises their motivation, eventually helping them in the learning process.

1.2 Problem Description

According to constructivism, children learn better if they have to construct the knowledge instead of simply learning it. One educational method based on constructivism that has been used for a few years now is the "learning by teaching" approach, further explained in section 2.1.3, which main idea is to engage students by giving them the teacher's role. It is based on the principle that while having the responsibility to teach, people strive harder into learning the concepts and put larger efforts into preparing both the class and questions - to and from the tuttees. This method has been largely studied and has proven results in many schools.

Based on this method appeared the main concept of this work: provide means for children to construct their own educational game application. On the light of the recent studies in the Child Computer Interaction field, we have decided to apply some of the studied techniques, referred in section 2.2.1, in order to construct an application that engages children, by involving them in the design process. At the age of 7 to 11 - stage concrete operational [31] - kids are already able to generate questions and answers about some subject which we hope will motivate them to learn more about the subject of the application.

The main goal of the system is to help children to learn by teaching their colleagues. The suggested authoring tool allows children to build interactive games, that involve the resolution of problems related to the subjects to be learned. While creating the applications, children need to know the answers in order to formulate the questions.

The idea of creating a pervasive application seemed appropriate since even though surrounded by technology, children feel more comfortable when interacting with everyday objects. With the intention of minimizing the application learning period, we have chosen to create a Tangible User Interface, which allows an interaction with real objects that are familiar to children, thus increasing their motivation.

The idea of children creating their own application with physical objects appeared from the notion that they are more familiar with their own toys, which they are used to use from the moment they have the necessary dexterity. If they don't have to be concerned about learning how to work with the application they have more time to spare and be creative.

A true pervasive interface is one that is completely embedded in the world, meaning that all the inputs and outputs are constructed in a way that the user doesn't realize he's interacting with a computer. Following this idea, during the interaction we have the goal to minimize the use of keyboards or other usual peripherical devices for data input, and base the interaction on physical objects and actions. This will be possible due to the advances in Tangible Augmented Reality (TAR) research.

1.3 Solution Approach

Keeping in mind the goal to create a tangible user interface that enables children to create their own educational games, one was confronted with some obvious challenges:

- *Creation a Pervasive System* - When working with the application, children shouldn't be aware they are interacting with a computer, which raised questions about data input and output;
- *Creation of a Tangible User Interface* - One had to find the tool that best suited the requirements of our system.
- *Development for (and with) children* - Creating an application for children should ideally be constructed or idealized by them.

The created application enables children to construct their own games, based on the structure of the "Game of life"[1] but with multiple choice questions in some of the houses (as the Trivial Pursuit[2]).

This construction, illustrated in Figure 1.1, is done in four different phases:

1. *Game Story Creation* - In this phase the children are able to create the story of their game. In the developed prototype, this is done by recording it with a microphone.
2. *Character Definition* - The children may choose between a person character (boy or girl) or an animal character (cat or dog). After that choice they choose their character's features.

[1] The Game Of Life in Wikipedia http://en.wikipedia.org/wiki/The_Game_of_Life
[2] Trivial Pursuit in Wikipedia: http://en.wikipedia.org/wiki/Trivial_Pursuit

3. *Scenario Choice* - In this phase children choose one from the available scenarios(the path through which the character will walk).

4. *Questions and Answers Definition* - Children input, for each question, the question itself and the four options (one correct and three wrong ones). In out prototype the questions were in audio format (input with a microphone) and the options in text (input with a keyboard).

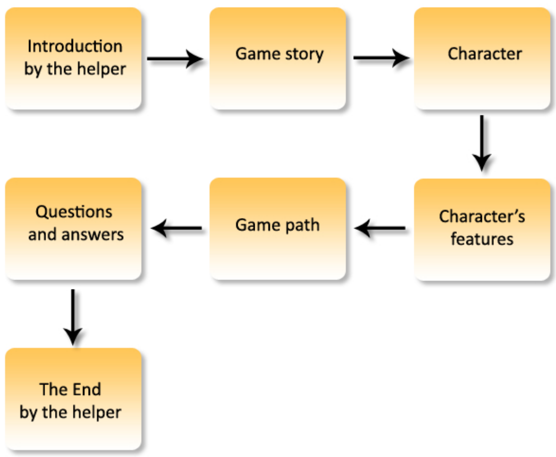


Figure 1.1 Application phases

The children are guided through each of these phases, obeying to well-defined goals at each one.

As explained further in section 2.2, the software creation process is made of three main distinct phases: design, implementation and testing, which may run in cycles until the desired result is obtained.

Bearing in mind that the target audience are children (aged between 8 and 10 years-old), part of the solution was the way they were integrated in the process. Even though it was not possible to include them as design partners through all the design process, they had an active part during the three design sessions described in Section 3.2 where they helped to answer to essential questions the team had about the application design.

The meetings were modelled according to recent research conducted mainly by the ChiCI group [3]: Bluebells and Warp Speed Design methods.

In order to test the usability of the developed application (and, after all, the efficiency of the design sessions) a final usability test was done to the created prototype with different children that were not involved in the application design.

1.4 Main Expected Contributions

Our contribution is the development of a tool that enables children to create their own educational content following the Learning-by-Teaching method. The game is constructed through the use of real objects, recurring to Augmented Reality technology to visualize the content created.

This tool was created by involving children in the design process in order to create a more intuitive application that requires less (or ideally none) learning.

The final prototype was tested with a group of children in order to study their motivation towards the game construction and the application usability. They also answered a small questionnaire with a method developed specially to test applications with children[13][32] This questionnaire had the purpose to study their general opinion and motivation towards the application concept.

[3] http://www.chici.org/ - Child Computer Interaction Group at UCLAN, UK

1.5 Document Organization

This document is organized as follows:

1. Chapter 2 - In this chapter some related work and literature are presented. Initially the cognitive development one goes through as a children is explained. Next, some literature about child-computer interaction is presented and, finally, the concept of pervasive interfaces and tangible user interfaces are studied.
2. Chapter 3 - In this chapter the main design aspects are covered. The initial study, requirements gathering, design sessions with children and application architecture are explained.
3. Chapter 4 - In this chapter the main aspects of the application implementation are explained of both the game creation application and the game itself.
4. Chapter 5 - Describes the final usability tests, their results and analysis.
5. Chapter 6 - Presents he final considerations related to this work and the work that can be done to follow this study.

2 . State of the art

2.1 Cognitive Development

Cognition "is generally accepted to mean the process of thought" [38]. Understanding cognition development is essential when working with children, since the major difference between them and adults is the fact that their cognition is still in an early development stage, which leads them to acknowledge objects, facts or people in a different way that adults do.

2.1.1 Piaget and Vygotsky

Jean Piaget (1896 - 1980) and Lev Vygotsky(1896 - 1934) were two major researchers of cognitive development. Even though they never met, they had similar interests and developed their theories in the same decades. They were both a big influence in constructivism, an educational theory that's based in the belief that "Knowledge is actively constructed by the learner, not passively received from the environment". [12]

Piaget developed the "Theory of Cognitive Development" [31], which divides children's development in 4 different stages, according to their age: Sensori-motor (up to 2 yrs), Pre-operational (2 to 7 yrs), Concrete operational (7 to 11 yrs) and Formal operational (from 11 yrs). Each one of these stages corresponds to a new scheme of world interpretation and, as he states, are triggered by biological factors.

Vygotsky, on the other hand, concluded that even though there are several development stages, they do not necessarily depend on the child's age. His research aimed at proving that the social factors inherent to the child are the main sources to her development.

Even though the two psychologists had different ideas, both theories were accepted in developmental psychology: not only the children's age is important but also the social incentives that she is bounded to.

2.1.2 Motivation, Goals and Learning

Even though there is no formal definition for motivation, one intuitively knows that it is the foundation of education. As Terrell Bell, a U.S. Secretary of Education, once said: "There are three things to remember about education. The first is motivation. The second one is motivation. The third one is motivation" [18].

There are two different types of motivation: intrinsic and extrinsic. Intrinsic motivation occurs when there are no apparent incentives in learning and, therefore, extrinsic motivation occurs due to an exterior incentive (material issues are the most used example and, most of the time, the more compelling ones).

In every decision that is made there's a goal to reach (even if the goal is not to reach some other state). This is why goals are a fundamental piece when speaking of motivation, and Achievement Goal Theory [10] is normally the most referred when studying children's cognitive and learning process.

When in a class, there is no immediate goal to be achieved, since usually a child doesn't acknowledge the future importance of learning. Hence, unless she is naturally interested in the subject, she won't be as motivated as if she was playing a game, which has well-defined goals as well as immediate feedback, and she probably won't stop playing it until she achieves them. Compelling games also have another important component associated: the flow. The flow concept was introduced by Mihaly Csikszentmihalyi: "Flow represents the feeling of complete and energized focus in an activity, with a high level of enjoyment and fulfillment" [11]. During a flow experience one loses track of time and the surrounding world. A flow experience happens when the activity challenges are balanced with the user skills to overcome them. The flow may be defined by its eight components[7]:

- A challenging activity requiring skill
- Merging of action and awareness
- Clear goals
- Direct, immediate feedback
- Concentration on the task at hand
- A sense of control
- A loss of self-consciousness
- An altered sense of time

Chen [7] relates the flow concept with computer games, referring a four-step methodology to create games with which players reach the flow - a high-level motivation state :

1. Mix and match the components of Flow
2. Keep the user in a zone where the challenges and his skills are balanced so that he doesn't reach either anxiety or boredom
3. Offer adaptive choices, allowing different users to enjoy the Flow in their own way
4. Embed choices inside the core activities to ensure the Flow is never interrupted

Thus, when creating a game-like application one should try to go through these steps keeping in mind the cognitive state of the target audience.

2.1.3 Learning by teaching

Learning by teaching is a method developed by Jean-Pol Martin and consists in giving the students the educator's role, by letting them teach a lesson or part of it. It first appeared due to the lack of teachers but, when people realized that students that were teaching learned more and started to gain capacities and methods the others wouldn't, it became a more used methodology, although more controlled.

The two major phases of this method are explaining, where students study and prepare the way they'll present the subject, and questioning, where the tutors make and answer questions to the tutees. Probably the main reasons for the students' motivation to teach (which compels them to engage in the activity) are the well-defined goal of teaching other students some subject and the responsibility that comes from it. This phases' success is further explained in [26].

Based on this theory the Teachable Agents Group at Vanderbilt University[1] developed a system where students interact with a teachable agent, an entity that is prepared to be taught, which acquires the knowledge and is able to answer questions posed by the student according to what it acknowledged. They evaluated the software by running some tests on students and concluded that the students using the system acquired a more effective knowledge that the ones that didn't [34].

2.2 Human-Computer Interaction

Through the years, computers have become an active part of people's daily activities, and Human-Computer Interaction is a computer science's field that arose from the need to optimize the usability of computer applications, in order to facilitate their use, and maximize users'

[1]http://www.teachableagents.org/ : homepage of the Teachable Agents Group at Vanderbilt University

satisfaction. Human-computer interaction can be defined as "a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them." [1]. Its main purpose is to develop techniques that will help to create the so-called user-friendly applications by focusing on the target users.

Its main concerns are the design, implementation and evaluation of interactive computer software, developing techniques that facilitate these development phases and that help the users, minimizing their effort both in testing and using the applications. These three phases commonly run in a cycle as shown in Figure 2.1.

Iterative design is an important element in usability engineering. It is the current most used process for developing user interfaces. It's a specification of the spiral model described by Boehm[6]. In iterative design the software is refined in each successive trip around the design cycle, following the spiral model (Figure 2.2), and achieving more complete, precise and realistic visions on each iteration. The radial dimension of the spiral model corresponds to the cost of each iteration step.

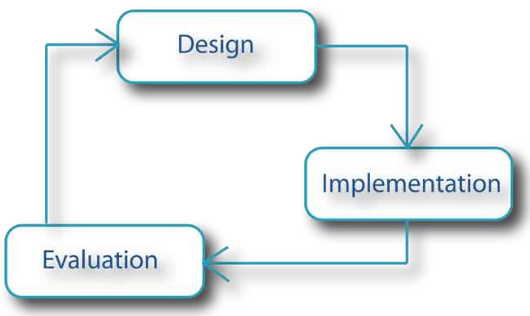


Figure 2.1 Software Design Cycle

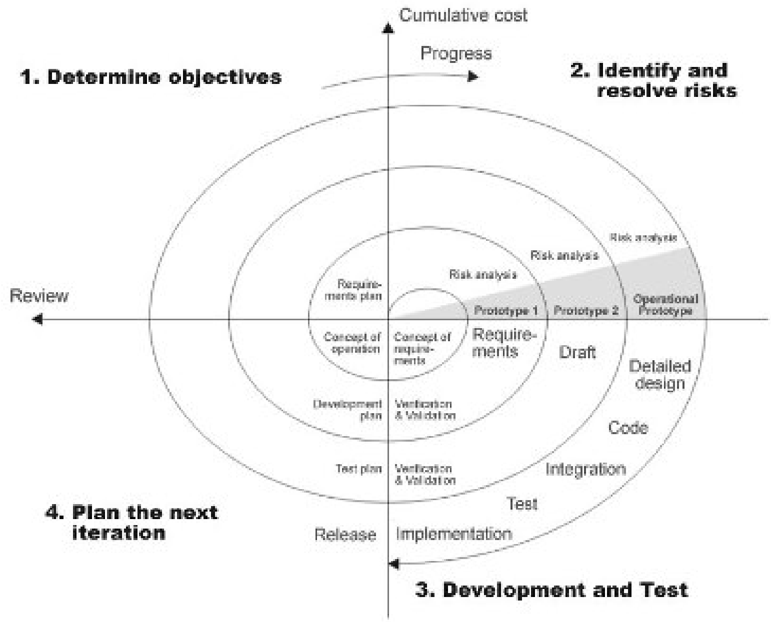


Figure 2.2 Boehm's Spiral Model, from [6]

2.2.1 Children are not adults

Today's children are likely to grow up in a technology-rich reactive environment, especially since technology started being used as an active part of education. Even though everyone knows that children are not adults, this knowledge has only practically been used in software construction in the past few years. Up until then, the only concern when making children software was, in the best cases, to make a more appealing and colorful application. As seen in Figure 2.1 there are three main phases in software construction and in both design and evaluation phases it is essential that children play an active part. This is how Child-Computer Interaction [2] appeared, as a subfield of Human-Computer Interaction that takes into account children's specific attributes and needs.

"According to Piaget, children that are less than ten years old have great difficulty with

[2] Child-Computer Interaction Group HomePage : http://www.chici.org/

abstraction and conceptual problem solving, skills which are integral to most interviewing and participatory design techniques." [2]. Based on this idea, a new set of design techniques were created (or adapted from the HCI field). Alison Druin, a pioneer researcher in the area, sees four main roles (Figure 2.3) that children can play in the technology design process [13] :

- *User* - Children playing this role use and test technology that has already been released. The researchers observe them to understand if they adapt well to the application and to determine the possible future developments.

- *Tester* - Children playing this role test the prototypes of the application before it is released. They are observed and/or asked about their opinions by professionals.

- *Informant* - Children playing this role probably take part in the many different stages of the design process. Probably they are even observed before any development in order to conclude what are their needs, during the development in order to determine how they interact, learn and adapt to the application, and after the completion in order to find bugs and usability issues.

- *Design Partner* - Children playing this role are completely involved in the design process: they are as important as the adults and they are both equally in charge, which helps them feeling empowered and, therefore, more willing to engage in the application design.

Each of these roles represents different involvement levels and can be related to each other: an informant probably will be asked to be a tester and, therefore, a user. Hence, the more complete way a child may be involved in the process is within the design partner role. The historical roots of each type of involvement, as well as their strengths and challenges are shown in Table 2.1.

Role	Historical Roots	Strengths	Challenges
User	Late ’60s / Early ’70s	Easier to use with children and keeps the researcher in charge.	Children participate less, the results will only be useful for future researches.
Tester	Primarily in the late ’80s/ early ’90s	Children can feel that adults want to listen to what they have to say. Also, it is usually little time consuming and can have a great impact in the development, helping to mitigate design errors in early development stages.	Children still don’t have a very active role since the researchers are still in charge and the suggestions they make may never lead to changes. May be hard for developers since children are extremely honest.
Informant	Mid ’90s	Children feel more empowered and challenged that in the previous roles, since they participate more actively, in tests and brainstormings at very different phases of the development process.	Although they have a more active role, children are not yet in charge. Since they have to be consulted more often, it is a challenge for both educators and parents.
Design Partner	Mid ’90s	Children feel very empowered as they see that their opinion is as important as the ones of any adult. Both change and learn with the experience.	Important decisions must be discussed by all elements of the team, and neither adults nor children are in charge, which makes it difficult. Also, takes more time than the other roles, since it’s a time consuming process, the one of having children and adults as equal stakeholders.

Table 2.1 Different roles a child can play in an application development process, based on Table 2 of [13]

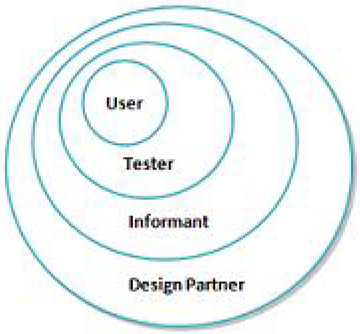


Figure 2.3 Four main roles a child can play in the technology design process.

2.2.2 Gathering the requirements of children applications

As with every other kind of application, a children's one design starts by gathering and analysing its' requirements. Sonia Chiasson's studied this question in "Design Principles for Children's Technology"[8].

In her work she finds, for each different type of children's development (cognitive, physical and social/emotional), their specific needs: "In order to properly meet children's needs and expectations, children's technology must take into account and support these development areas" [8].

Chiasson[8] gathers the general requirements of a children's application in her work.

2.2.3 Methods to include children in the design process

As stated before, the children's inclusion in the design process is extremely important. Some methods were created describing techniques In this section the most studied and documented ones are explained.

2.2.3.1 Cooperative Inquiry

When designing a children's application, if there are enough resources available, the ideal is to include the children in the process as design partners. Throughout the world this idea is being spread and probably the most used method to implement this concept is the one pioneered by the Human Computer Interaction Lab in the University Of Maryland, the *Cooperative Inquiry*, which appeared from "the belief that partnering with users is an important way to understand what is needed when developing new technologies." [14]. It is based on other techniques created for the HCI field and consists of three different methods:

1. *Contextual Inquiry* - This technique is based on the idea that children should be observed in their natural environment. It involves the "users", an interactor and the observers. The interactor is the person who's role is to ask questions and initiate the discussions concerning the activities. The interesting point is that the observers are both adults and children, even though each collect the data in their own way: adults find it easier to write text description and to work in pairs, so that one collects the activities and the other some quotes said by the users, and children find it easier to make it through drawings with little text descriptions, as in Figure 2.4 . The interactor's role has to be carefully assigned, since it's difficult both to children and adults: children are likely to get involved in the activity and forget what they are supposed to do, and most adults find it hard to forget that traditionally they are in charge and it's hard to let go of that structure and simply help children in the interaction.

2. *Participatory Design* - This technique does not have to follow *Contextual Inquiry*, but investigators found that when used next, it may focus in the main issues collected in the first. It functions mainly by low-tech prototyping with both children and adults, which is not always easy, since adults tend to think that crafting is mainly for kids and let them do

the major work, when it has to be done as a team. It is very easy to apply since it only needs some art supplies, and the design team.

3. *Technology Immersion* - This technique aroused from the idea that if children were immersed in an environment rich in technology for a long period of time (many hours a day, during a week, usually), which normally they are not used to, some new problems and behavioral patterns would appear. So this technique is based simply on observing them in a technology-rich environment and look for those patterns. In the CHIKids99 [3] this technique was employed through the whole conference duration.

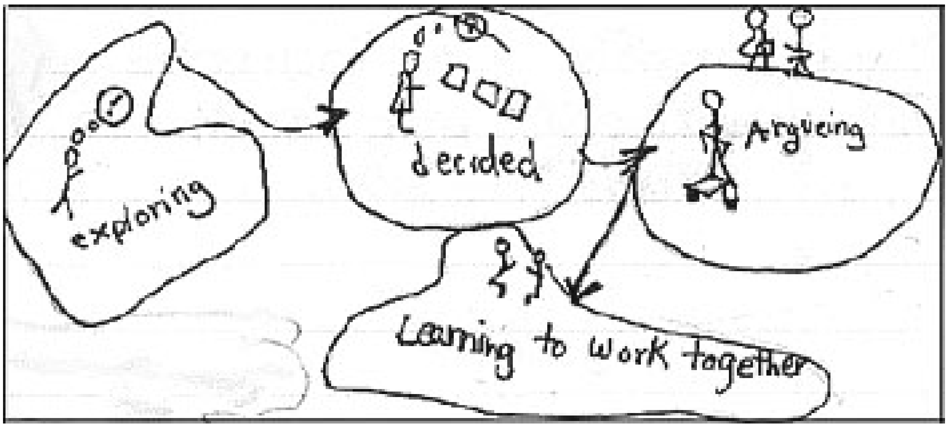


Figure 2.4 Contextual inquiry notes by a 7-year old child, from [14].

2.2.3.2 Bluebells Method

The Bluebells Method is "a design method that balances child-centered design with expert design in a progressive approach that marries the best of both disciplines"[25]. This method lays on the idea that each design session's results will lay the bases for the next one, until the team reaches a proper design. All the activities are divided into 3 phases: before, during and after.

[3]ChiKids99 conference in http://sigchi.org/chi99/chikids/

Before an activity occurs it must be prepared by the design team (requirements gathering, session goals, ..). During the activity the children are actually participating in the design of the application. Once the activity ends, the teams must gather the session results and analyse them.

This method comprises four activities, represented in Figure 2.5, that may be chosen to do with the children:

- *I-Spy* - This activity consists of observing the children as they interact in the environment of the application to be created. Children shouldn't know the nature of the project they are involved in. The design team should observe them and take notes.
- *Hide and Seek* - This activity begins by briefing the children about the application and ask them to write the words that they associate with the application and group them. Next, they are shown a prototype and are asked to put the words on some artifacts that are used in the application.
- *Tig* - This activity also starts by briefing the children with the application contents but, this time, they are also explained the concept of maps and web site navigation. The main purpose of this activity is to find out what the children think is intuitive as navigation in the application, by showing them a map with some application screens and ask them to place the navigation content on the locations.
- *Blind Man's Bluff* - In this activity a group (two or more children) works in order to define the visual aspect of the application. One of the children covers her eyes and imagines how the application looks like. The others draw, in a paper sheet, what the children is imagining. At the end they discuss whether the design has anything to do with the imagined application. The discussion phase is very important for it usually leads to some talking about the interface visual appearance and what makes or not sense to them.

This is a very flexible method and it gives the design team the freedom to choose which activities to apply and when to applied them, based on their needs.

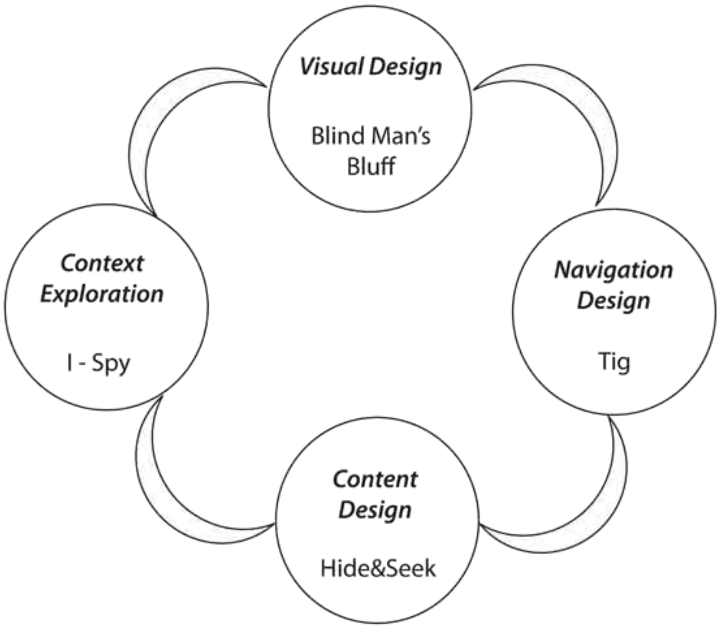


Figure 2.5 The Bluebells method activities

2.2.3.3 Warp Speed Design

One other method studied was the Warp Speed Design, created by Janet Read "for the design of workable tangible games" [33], a design method for children of ages 9 and up. This method is divided into 3 stages:

1. *Learning and Idea Creation* - In this phase the team shows the children a sketch of a tangible game and explains them how it works and what can be done with the available technologies (RFIDs,..). The team also tells the children about other games that can be done, so that they keep the idea for the next stages.
2. *Children as Interaction Designers* - The children are given action sheets, where they can draw the interactions that occur in their game. Due to the lack of time they are encouraged

to pick the first idea that comes into their mind.

3. *Children as Game Designers* - Once the action sheets are prepared, children give them detail and add the images and sounds they want to.

2.2.4 Gathering the children's opinions

Every time a prototype is finished (whether it is a paper or a full-working one) it should be tested. The objective of this testing process's objective is to measure the usability and fun, which are both essential and inseparable when dealing with children.

Janet Read, the leader of the ChiCI group [4], developed a tool that helps to analyze what children think of technology: the Fun Toolkit [32]. It is composed by three main techniques and the one to use depends on the application:

1. *Smileyometer* - It is composed of five smileys arranged in a line, with words associated to each one of them (Figure 2.6). It may be used before the test to measure the expectation of the child and after, to measure the satisfaction. The smileys were created by both researchers and children and, since they don't require a reading ability, they can be used with children as young as 4 years old.

2. *Fun Sorter* - This technique is to be used when the objective is to compare many technologies according to different constructs. It is composed of *n+1* columns (where *n* is the number of technologies) and *m+1* lines (where *m* stands for the number of constructs). An example is shown in Figure 2.7.

3. *Again Again Table* - This technique appeared from the idea that if one wants to repeat an activity, then it is likely that he has liked it. Since it's main objective is to compare

[4]http://www.chici.org/ - Child Computer Interaction Group at UCLAN, UK

activities or products, it cannot be used when evaluating a single one, but it also shouldn't be used when having an excessive quantity to compare, because the users may become bored. An example can be seen in Figure 2.8.

Figure 2.6 A Smileyometer awaiting completion, from [32].

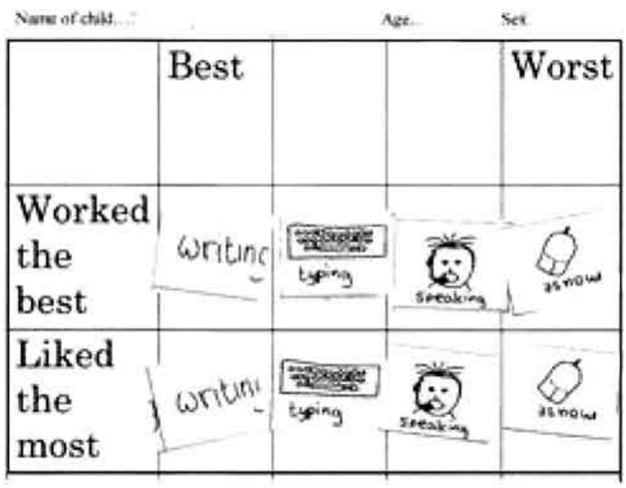


Figure 2.7 A Completed Fun Sorter showing how children position the picture cards in the boxes, from [32].

2.2.5 Computers' role in education

Seymour Papert is a mathematician, computer scientist and educator who, one may say, was also the pioneer of computer based education. He studied with Jean Piaget and not only has built a cognitive theory named constructionism, based on Piaget's theories [31], but he also created LOGO, a programming language built specially to improve the way children think and

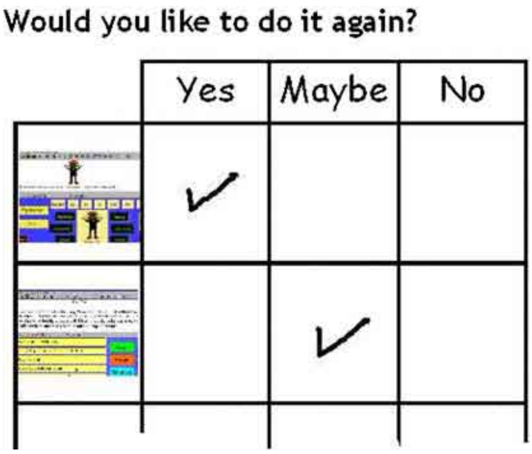


Figure 2.8 An excerpt from a Completed Again Again table that was being used to compare different word processing packages, from [32].

solve problems [39]. As the creator of constructivism, Papert believed that "Children can make their own educational software, and by making the software, they learn much more than by using it. Because when you make a piece of software, when you teach something, you have to think about what’s really going on, you have to think about the ideas." [29]. Papert shaped is research around the beliefs that "children can learn to use computers in a masterful way, and that learning to use computers can change the way they learn everything else" [30].

Based on LOGO, appeared the Scratch, a graphical block-based environment that "emphasizes media manipulation and supports programming activities that resonate with the interests of youth, such as creating animated stories, games, and interactive presentations" [27]. It was tested in urban youth (ages 8-18) at an after school center, over an 18-month period with positive results: the users, that has little or no prior knowledge of programming, learned on their own and used "commands demonstrating the concepts of user interaction, loops, conditionals, and communication and synchronization".

Another engaging educational software is Alice that appeared from the need to help university students who studied computer science to improve their programming skills in a 3D environment. They ran an evaluation procedure and concluded that "at risk students that participated in Alice, on average, received significantly higher grades than at risk students that did not participate in Alice" [9].

Many other educational applications have been created, and due to the fact that education is not always as engaging as it should, the term edutainment appeared as a form of entertainment directed to education. The most used edutainment applications are the educational games. This approach usually works better with kids, since adults' cognition is usually too complex to be engaged by educational games. However, one successful example of an educational game directed to adults is the UniGame, "a framework that provides the possibility for every interested teacher to apply game-based learning to his/her classes"[15].

Due to children's natural engagement with games, most educational games are directed to them.

2.3 Pervasive Systems

" (...) we are trying to conceive a new way of thinking about computers in the world, one that takes into account the natural human environment and allows the computers themselves to vanish into the background." [37]. This notion was introduced by Mark Weiser, the father of ubiquitous or pervasive computing. There are two main goals to reach when designing an ubiquitous system: detect people and objects presence in the surroundings and having an application completely embedded in the environment.

The concept appeared once computers started to become a part of people's lives and they had to start dealing with them in a daily basis. The idea that one may be surrounded with computers without even noticing is amazing and opens many doors in technology research.

A very important part of ubiquitous systems are the sensors and actuators: the first ones allow collecting the data about the surroundings and the second ones allow the system to act according to the data perceived. One common use of these technologies are the smart spaces, which have the technology embedded and react to data captured by sensors. An example of a smart space is a room where the light, heating or music played varies depending on people's presence or their preferences.

A factor that mostly contributes to the success of ubiquitous applications is their invisibility: the more invisible a system is, the less intruded users feel and, therefore, the more comfortably they use it. This question raises two problems: how to design the hardware so that it becomes embedded in the environment, and how to design the software so that all the hardware interacts with each other and with the users.

Of course that when dealing with children we also have to pay further attention to the application's design because children and adults have different dexterity, both psychological and physical, which influences the way the application is seen and the devices are manipulated.

For instance, it was studied that often children find museums "staid and boring places, austere environments where they passively view objects, or passively listen to historical accounts of the past" [19]. Since museums are a very important part of our education and culture many of them started searching for methods to enhance people's visits, making them more interactive. One of the solutions implemented, which was actually directed to children, was the one developed in the "Re-Tracing the Past" exhibition in the Hunt Museum of Limerick, Ireland. The project[19] consisted of a tangible collaborative replica of a study room. It was populated with objects one could find in a study room in the past and had also two interactive elements: the desk and the trunk. Both detected the objects placed recurring to RFID tags and presented related information.

Technology is becoming more pervasive in children's environments and can be found embedded in items they interact with in a daily basis, like their toys, lego blocks or musical instruments[17].

2.3.1 Tangible Augmented Reality

In order to understand what augmented reality stands for, one should first understand the concept of virtual reality(VR) and its' surroundings. Sutherland introduced the concept as we know it today when he explained his vision[35]:

- "A display connected to a digital computer gives us a chance to gain familiarity with concepts not realizable in the physical world. It is a looking glass into a mathematical wonderland."
- "If the task of the display is to serve as a looking-glass into the mathematical wonderland constructed in computer memory, it should serve as many senses as possible"
- "There is no reason why the objects displayed by a computer have to follow the ordinary rules of physical reality with which we are familiar."
- "The computer can easily sense the positions of almost any of our body muscles.(...)Our eye dexterity is very high also. Machines to sense and interpret eye motion data can and will be built.(...) An interesting experiment will be to make the display presentation depend on where we look."
- "The ultimate display would, of course, be a room within which the computer can control the existence of matter. A chair displayed in such a room would be good enough to sit in."

Sutherland created what is considered the first virtual reality device: the head-mounted display(HMD), a display device which purpose "is to present the user with a perspective image which changes as he moves"[36]. As the user turns his head, the images' perspective change, giving the illusion of seeing three-dimensional objects.

As Feiner stated [3] "Virtual reality focuses an enormous apparatus on simulating the world rather than on invisibly enhancing the world that already exists.", meaning that the concepts of virtual reality and ubiquity are somehow opposite of each other, but the first helped to define Augmented Reality (AR), for it is rarely defined without recurring to Virtual Reality definition. Probably the most comprehensible notion of AR is "AR allows the user to see the real world, with virtual objects superimposed upon or composited with the real world. Therefore, AR supplements reality, rather than completely replacing it" [3]. This definition may be complemented with the virtuality continuum created by Milgram (Figure 2.9).

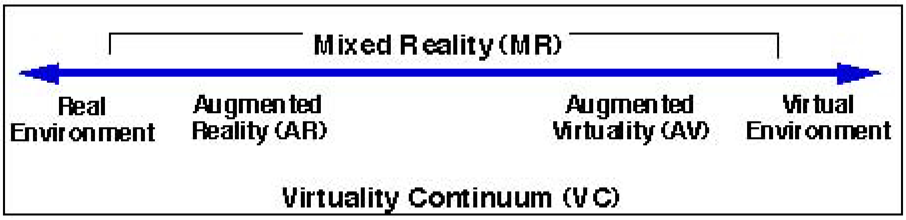


Figure 2.9 Milgram's virtuality continuum, from [28]

Even though the concept of Augmented Reality is amazing, one may easily think that it is not really ubiquitous, since usually a piece of paper with a marker or some other specific device, such as an HMD, is needed. For these reason a new type of HCI was established by Hiroshi Ishii and Brygg Ullmer research team (MIT Tangible Media Group[5]): Tangible User Interfaces (TUIs). "TUIs will augment the real physical world by coupling digital information to everyday physical objects and environments."[21] which was the way to create a bridge from the bits (the cyberspace) and the atoms (the physical environment). They worked around three concepts which are still used as different ways to physically interact with the digital world :

[5]MIT Tangible Media Group HomePage: http://tangible.media.mit.edu/

1. *Interactive Surfaces* - "Transformation of each surface within architectural space (e.g., walls, desktops, ceilings, doors, windows) into an active interface between the physical and virtual worlds";
2. *Coupling of Bits and Atoms* - "Seamless coupling of everyday graspable objects (e.g., cards, books, models) with the digital information that pertains to them";
3. *Ambient Media* - "Use of ambient media such as sound, light, airflow, and water movement for background interfaces with cyberspace at the periphery of human perception".

One important thing to keep in mind is that "when designing a tangible AR interface, the form factor and affordances of the physical objects used in the interface must be carefully considered"[4], which means that the user should intuitively understand what happens when interacting with an object without having to think about it.

One common use of Tangible Augmented Reality (TAR) is to support storytelling applications. An example is the project developed in the HCIL, the "StoryRooms", a "physical storytelling technology" that enables children to program objects' behavior in order to tell the stories they want to. The team developed a theoretical model to create storytelling physical oriented technologies (SPOT) which describe techniques and results obtained [17].

"Typically, physical educational environments are revered as a more natural, explorative environment for young children, since children generally play with and learn in physical environments"[16]. It is intuitive that AR would be ideal when dealing with children, especially when in an educational environment, as it is expected that the natural interaction will motivate them into exploring the application further.

2.3.2 ARToolKit and derivates

The ARToolkit[6] is a tool that enables the creation of Augmented Reality applications. It is a C/C++ library which is based in the recognition of markers in order to augment the perceived reality. The creation of this toolkit was a big step in AR development since it facilitates the creation of AR applications.

The project BlackMagicBook was developed by the Human Interface Technology Laboratory in New Zealand (HITLabNZ)[7] and it was one of their first projects. It consisted of an augmented reality book, and it is still available at [20]. They provided an explanation of how augmented reality worked through Figure 2.10.

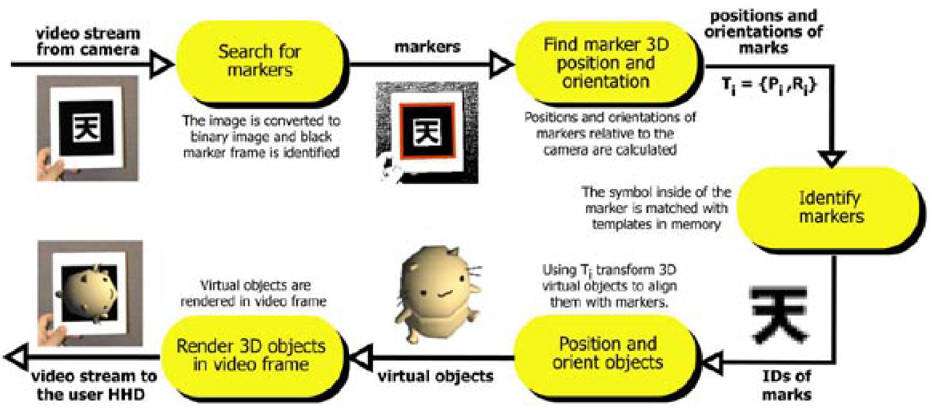
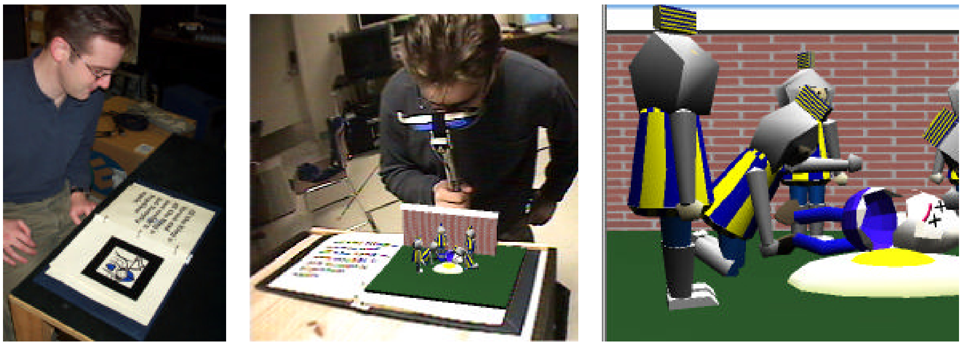


Figure 2.10 HITLabNZ's explanation of how AR works, from [20].

One example of an application created with ARToolKit is the MagicBook, "a Mixed Reality interface that uses a real book to seamlessly transport users between Reality and Virtuality"[5]. It is a book that allows the users to choose the kind of experience and immersion they want when reading it. It may be read like a normal book as in Figure 2.11(a), without recurring to the use of any device, but it also may provide both augmented and virtual views. Recurring

[6]ARToolKit Home Page : http://www.hitl.washington.edu/artoolkit/
[7]HITLabNZ HomePage : http://www.hitlabnz.org/

to an HMD the user may choose to simply view the scene he is reading in a 3D way (static or animated), as in Figure 2.11(b) or to immerse and interact with it as in Figure 2.11(c).



(a) Reality (b) Augmented Reality (c) Immersive Virtual Reality

Figure 2.11 Using the MagicBook interface to move between Reality and Virtual Reality, from [5].

Recently new toolkits based on the ARToolKit appeared: NyArtToolkit[8] and the FLAR-ToolKit[9] implemented respectively for the Java/C-Sharp/Android/C++ and ActionScript3 programming languages.

The magicBook refered previously is a perfect example of a way in which AR could be used in education, if the book's content was related to a subject to be learned. If a child can grab an object that can instantly be presented with a video, animation, story or related images, she would probably pay more attention becoming a motivated learner. Of course that the content presented is of extreme importance and must be carefully chosen.

2.3.3 The TUIO Protocol and ReacTIVision

One other approach is the table-top tangible user interface where one has a transparent table and the camera is below it, capturing the markers on the objects. This way the users never have to look at the markers and won't feel as intruded as if they did.

[8]NyArtToolKit HomePage:
[9]FLARToolKit HomePage: http://www.libspark.org/wiki/saqoosha/FLARToolKit/en

s	sessionID, temporary object ID, int32
i	classID, fiducial ID number, int32
x, y, z	position, float32, range 0...1
a, b, c	angle, float32, range 0..2PI
X, Y ,Z	movement vector (motion speed and direction), float32
A, B, C	rotation vector (rotation speed and direction), float32
m	motion acceleration, float32
r	rotation acceleration, float32
P	free parameter, type defined by OSC packet header

Table 2.2 Semantic types of set messages, from [23]

The TUIO Protocol is "a simple yet versatile protocol designed specifically to meet the requirements of table-top tangible user interfaces"[23]. It mainly defines two types of messages: *set* and *alive*. The first ones have information about the state of an object and the second ones have the information about all the objects that are currently on the table. In order to uniquelly identify each message, one other message was defined - *fseq* - which is sent before any update with the unique frame id.

The messages are formatted as follows:

```
/tuio/[profileName] set sessionID [parameterList]
/tuio/[profileName] alive [list of active sessionIDs]
/tuio/[profileName] fseq (int32)
```

The TIUIO protocol also defines some parameters, which are used in the sent messages. These parameters and their meaning are explained in Table 2.2.

ReacTIVision is a framework that implements the TUIO protocol. It is "an open-source cross-platform computer-vision framework primarily designed for the construction of table-based tangible user interfaces"[22]. Its architecture is illustrated in Figure 2.12.

ReacTIVision is a multitouch framework and implements both fingertracking and markers recognition. In order to optimize the recognition algorithm the amoeba markers were created

(Figure 2.13).

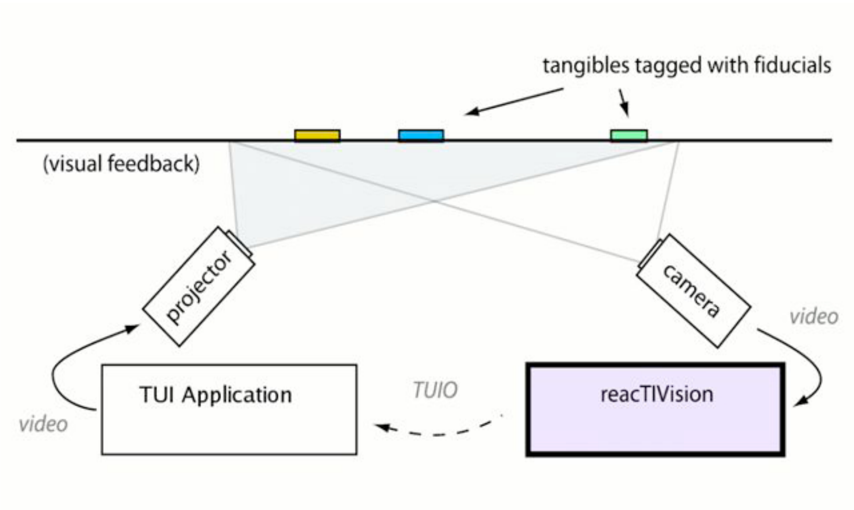


Figure 2.12 reacTIVision diagram, from [22]

Figure 2.13 Amoeba fiducials, from [22]

One application created with reacTIVision is the reacTable[10]: "is a novel multi-user electro-acoustic musical instrument with a tabletop tangible user interface"[24]. Its' architecture is as shown in Figure 2.12 and is composed by mainly two parts: markers and finger tracking by the camera and the images and sound projection on the table (Figure 2.14).

It also allows collaboration between clients which is further explained in [24].

[10]reactable homepage: http://www.reactable.com/reactable/

Figure 2.14 Reactable

3 . Methodology - Design

In this chapter the application design is described. Initially a technology study was developed in order to arrive to the initial concept. Next, the design sessions with the children were conducted in order to get a better idea of how the application should be. During these phases, the system architecture of the application was defined. Each of these parts are further explained in this chapter.

3.1 Initial Technology Study and Concept

Considering the goal to create a tangible application that allows children to create their own educational games, one had to structure what the application would look like and choose the technologies that could make it possible.

The game to be created is structured as the well-known "Game of life"[1] but with multiple choice questions in some of the houses (as the Trivial Pursuit[2]).

The initial idea was to create an application which divided the game creation in four different phases:

1. Introduction/Story - In this phase the children will be able to define an introduction to their game.
2. Character - In this phase the children may choose what their game character will look like
3. Scenario - In this phase children will choose how the game board will look like.
4. Questions/Answers - At the end children will define the game questions, correct answers and other options.

[1]The Game Of Life in Wikipedia http://en.wikipedia.org/wiki/The_Game_of_Life
[2]Trivial Pursuit in Wikipedia: http://en.wikipedia.org/wiki/Trivial_Pursuit

Bearing in mind these phases and the ubiquity principles, one started gathering the application's requirements. Tables 3.1, 3.2 and 3.3 summarize the requirements that guided the development of the work presented in this document. This tables were obtained by gathering only the requirements that specifically concerned the application previously described.

Phase	Requirements
Literacy	Interfaces should be strongly visual, avoiding text as much as possible and reducing cognitive load Content-specific metaphors are useful in helping children navigate interfaces Instructions should be presented in an age-appropriate format Instructions should be easy to comprehend and remember
Feedback and Guidance	Children are impatient and need immediate feedback showing that their action have had some effect, otherwise they will repeat the action until some outcome is perceived Interfaces should provide scaffolding and guidance to help children remember how to accomplish tasks Icons should be visually meaningful to children The interface should provide indication of the current state of the system, whether it is busy processing or waiting for input from the user
Development	Children's interfaces need to take into account the fact that children may not yet understand abstract concepts Children's interfaces should not make use of extensive menus and submenus as children may not yet have the ability to categorize or have the content knowledge required to navigate efficiently Children are accustomed to direct manipulation interfaces, their actions should map directly to the actions on the screen. If other styles are used, expect that most users will require training and that some will be unable to grasp how the interaction works

Table 3.1 Cognitive Requirements, based on Tables 1 to 4 of [8]

After defining these phases, one considered these requirements and gathered a list of questions to be answered in order to create the application:

- What type of instructions are easily understandable by the children?
- Which technology provides for presenting the necessary immediate feedback?

Phase	Requirements
Tangibility	Children like tangible interfaces because they enjoy being able to physically touch and manipulate the devices Direct manipulatives allow children to explore and actively participate in the discovery process Physical props and having large input devices encourages collaboration Superficial changes to the design can produce very different physical interactions. Different interfaces emphasize different actions

Table 3.2 Physical Requirements, based on Tables 5 and 6 of [8]

Phase	Requirements
Motivation and Engagement	Technologies should give children the ability to define their experiences and be in control of the interaction Animated pedagogical agents are useful for learning environments; even those who do not provide any advice or interaction are perceived positively Expressive, domain-specific agents are useful due to pedagogical benefits and positive affective impact On-screen character interventions should be supportive rather than distracting Activities should be inherently interesting and challenging so children will want to do them for their own sake Supportive reward structures that take into account children's developmental level and context of use help keep children engaged
Social Interaction	Children's technology should facilitate social interactions between children Children's technology should account for children's beliefs about computers and interact in a socially consistent manner

Table 3.3 Social/Emotional Requirements, based on Tables 7 to 9 of [8]

- Which icons (to the markers) are meaningful to children?
- How to present a guidance through the interface?
- How to map children's actions?
- How to create a physical interface that enhances children interaction?
- Which activities are interesting and which are boring?
- How to create an application where children interact and collaborate with each other?

Phase	Data to be gathered	Methods to Gather Data			Technology Requirements
		Markers	Audio	Other	
Intro/ Story	The game initial story / presentation	–	√	–	Audio Recording
Character	The game character and its characteristics	√	–	–	Changing the character's features and immediate feedback - hair, eyes, cloth colors
Scenario	One from the available game scenarios (previously created)	√	–	–	Immediate feedback of the chosen scenario.
Questions/ Answers	The game questions, as well as the respective right a wrong answers	?	√	?	Audio Recording

Table 3.4 Game phases and their respective input methods

Some of these questions had to be answered before the design meetings with the children while others could only be answered by the children.

Considering the guidelines of the requirements tables and the questions created, one decided to draw the initial concept for each phase and input methods to be used. The two main input methods were the objects (with markers) and the audio (captured with a microphone). In Table 3.4 the relation between these methods and each construction phase are presented. There were also drawn the requirements the technology had to meet in order to develop each phase. These input methods, as well as the technology requirements, were only ideas and could still be changed if the design sessions' results pointed to another direction.

As can be noticed, the questions/answers phase hasn't got the input methods well defined so this became another question to be answered by the design sessions.

Regarding the technology to be used, one chose the ReacTIVision framework (Section 2.3.3) since it was specially created to develop table-top tangible interfaces and is, for this, more efficient than the others (as the ARToolKit). After making this decision, one started to study the possible programming languages to implement the TUIO client in the application.

In order to be able to provide the necessary immediate feedback to the children's actions, in some phases, like the character setup one, the Flash technology's use was essential, since it is the one that better allows the immediate changing of cloth, eyes, or any other character part, without changing the rest of the character's appearance.

Even though the Flash language was necessary to provide immediate visual feedback to the children, it had one very important limitation: it doesn't allow audio recording.

After analysing several possible languages such as C++, Java and C# one the decided to use the C# language since it is integrated in the .NET framework, it allows the creation of Windows applications (easier GUI objects control), has a simple integration with the ActiveX controls and, comparing to the C++ (a language that also presents these advantages) it has a smaller learning period.

Finally, one decided to use the C# language to create the base application which communicates with a flash module, when the necessary feedback to provide to the children couldn't be done otherwise.

Regarding the physical interface, and bearing in mind that children shouldn't perceive the existence of the camera or any device), one decided to create a closed table/box that would be of a regular size (as tall as a normal table) and markers with a size that was easy to manipulate by the children.

As for the guidance in the interface the team decided to create an agent that would be present through all the game creation process, which provided the instructions and comments to the children's actions.

Concerning the collaboration requirement, one decided that the game creation activity was better made in groups of children, giving them the option to create at least as many characters as the number of group elements.

Relating the rest of the presented questions, they were all specific to children and had to be answered by them. These were the questions one wanted to see answered by the design meetings' results.

3.2 Design Meetings with Children

As explained before, part of this thesis' study refers to children's inclusion in the design process. In order to do that one had to study several design methods, choose the one that best suited what we wanted to find out and the available resources.

From the methods referred in Section 2.2.3 both the Bluebells Method, as a guideline for the sessions, and the Warp Speed Design, as inspiration and base for the first prototype testing, were considered when planning the sessions. Also, both the methods are documented in an understandable and reproducible way.

Due to the fact that only three meetings of 45 minutes each were arranged, they had to be carefully planned, so that the team could study as much as possible.

Based on both the Bluebells method [25] and the Warp Speed Design [33], the 3 sessions were initially structured as follows:

1. *First session*- Tig Activities - In this session the children were briefed about the main purpose of the meetings and the importance of their opinions and involvement's in the design process. The focus of this session was to study both the content and the navigation design.
2. *Second session*- Blind Man's Bluff - Using the results obtained on the previous session, this one intended to study the visual design of the application.
3. *Third session*- First Prototype - In this session one wanted to show the children the first

prototype of the application which was created based on their contributions during the previous sessions. One also wanted to evaluate the prototype, gathering children's opinions and comments. A working prototype was presented to the children so that they could simulate the creation of a game of their choice.

3.2.1 First Session - Tig

3.2.1.1 Before the session

As the Bluebells method explains, before the session occurs some planning must be made. The more important is to understand what are the main questions the design session should answer: the outcomes of the session. These outcomes are supposed to serve as the base for the next session. After a cautious study the team gathered some questions that should be answered in the session:

1. What should be the logical game construction path?
2. What is the preferred input "device"?
3. What should be customizable or not about the game character?
4. Is the game's idea interesting and worth studying?

In order to answer the first three questions, an activity was prepared for the first session. To each children it was given a cardboard (Figure 3.1) previously prepared and four cards. The cardboard represented the game construction path and each card represented one part of the game construction process.

Each of the four cards corresponds to a different activity, related to what we wanted to know:

1. Introduction/Game Story ((Figure 3.2(a))) - The children had to write or draw the story of their game.

2. Character (Figure 3.2(b)) - The children had to draw or describe their character and then rank (from 1 to 10) what was the character's most important feature(1) to the least important one (10).
3. Scenario (Figure 3.2(c)) - In this phase the children only had to draw or describe their game's scenario.
4. Answers/Questions (Figure 3.2(d) - This phase's activity was to glue the given input methods (Figure 3.2(e)) by their preference order, in the Questions card.

At the end, each children would glue the cards in an order that had meaning to them which would show their logical construction path.

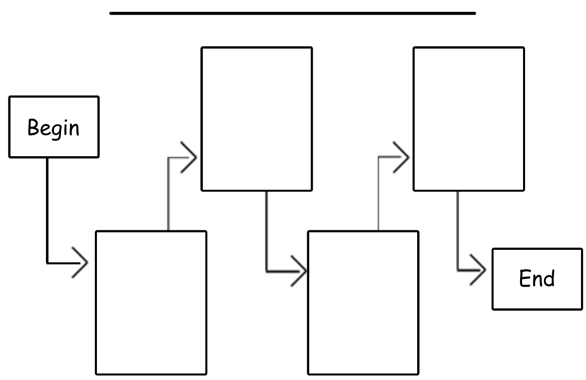
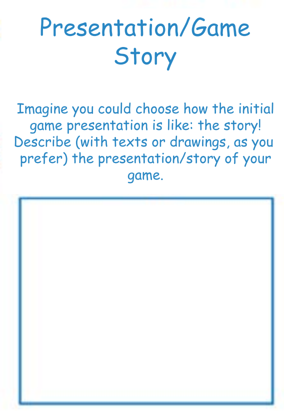


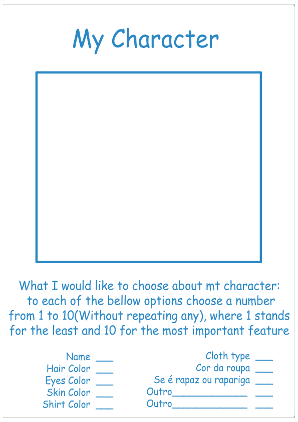
Figure 3.1 Cardboard

Once they finished the activity the children were asked to answer a questionnaire with four questions:

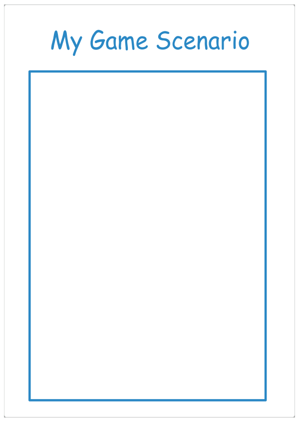
1. What did you think of today's activity?
2. Do you think that what you did today will help in the application's construction?
3. How many days a week do you use your computer (both to work or play)?
4. Would you use this program to construct games to your friends and family if you had it?



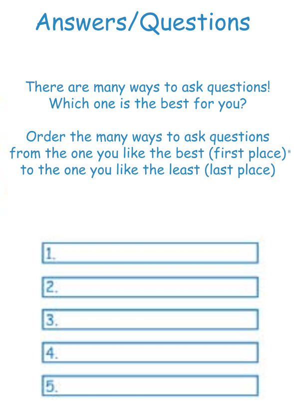
(a) Presentation



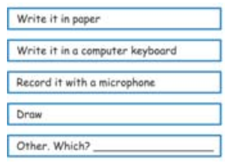
(b) Character



(c) Scenario



(d) Questions and Answers



(e) Questions' Options

Figure 3.2 Game Cards

All answers to these questions are ranked from 1 to 5, where 1 corresponds to the most negative answer and 5 corresponds to the most positive answer. In this questionnaire one applied the smileyometer technique [32].

3.2.1.2 During the session

First there was an explanation part, where children were told how important their participation was, what was the objective of the session and, more specifically, what they were going to do.

Each children worked on a table with a cardboard and the respective cards. Once they finished the activity they were asked to answer the questionnaire, reinforcing the idea of the importance of their involvement in the project.

(a) (b)

Figure 3.3 Photos from the first design session

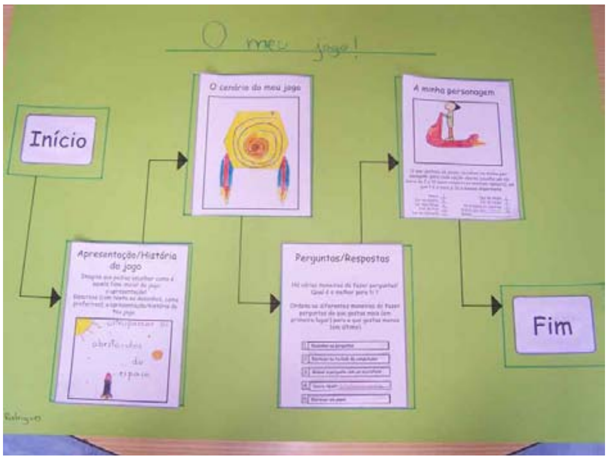


Figure 3.4 Cardboard of one of the children, when the activity was finished

3.2.1.3 After the session

Once the children finished the activity the data was gathered and its analysis provided some insight on the children's preferences. Regarding each of the previous described questions that were supposed to be answered by this activity, we came to the following conclusions:

1. *What should be the logical game construction path?* - Without noticing it the children gave some insights about the construction path that should be followed. As an example, none of the children thought of starting the construction of the game by defining the Questions/Answers. Mostly they've chosen that as the last thing to define. According to the results in Figure 3.5, the path chosen for the application construction was: Presentation, Character, Scenario and Questions/Answers. Although Character and Scenario have the same weight in both second and third construction phases, one decided to define the Character before the Scenario, since the character was present in every story told by the children (which means that it is more important to them than the Scenario).

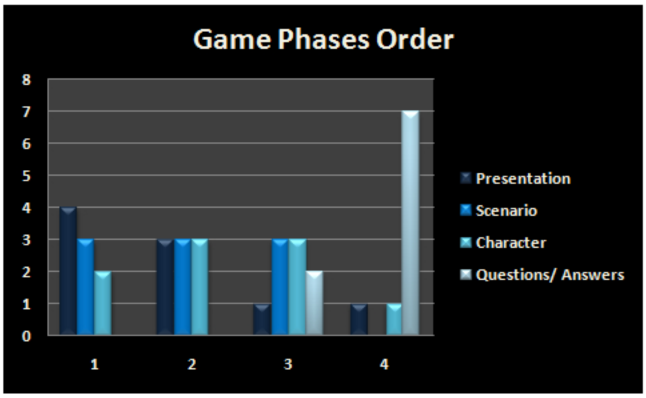


Figure 3.5 The different game parts, and their order

2. *What is the preferred input "device"?* - In this part children have noticed a little more that they were directly contributing to some decision making of the team. Although one initially thought that writing on a paper would be preferred, the computer keyboard was the most chosen input method, as it can be seen in Figure 3.6.
3. *What should be customizable or not about the game character?* - In order to be able to answer this question, the children were asked to draw whatever character they liked and,

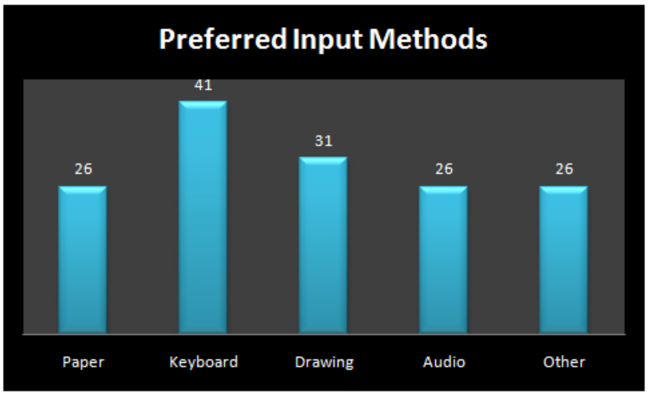


Figure 3.6 The different input "devices" and the children's choices

afterwards, rank from 1 to 10 (where one stand for more and 10 for less important) the main features of a possible game character that could be customized. They also had two blank spaces so that they could suggest and rank some other features that weren't present in the cards.

First of all the ranks were studied and, as can be seen in Figure 3.7, the two more important character features were considered to be the name and the gender of the character. Right next came the eyes, skin and hair color. In the white spaces, the recurrent general features were the character's personality and the shoes' color.

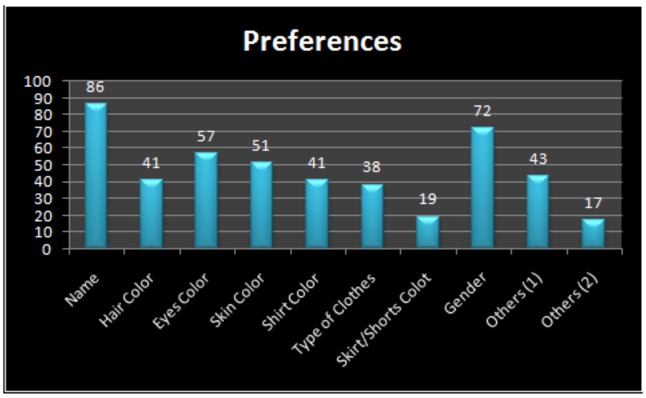


Figure 3.7 The different character's features and their importance according to the children that participated in the activity

Even though most children chose a person as their character (Figure 3.8) one noticed that 3 of the 10 children ended up choosing animals. The team didn't expect that and, although most children still chose people as the main character, one decided to give them both options in the construction: human (girl or boy) and animal (cat or dog).

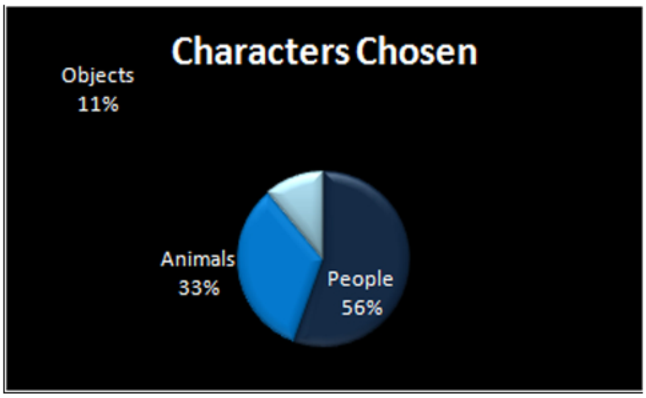


Figure 3.8 The characters that children chose

From the questionnaires, we obtained the results shown in Figure 3.9: which describes the average answer for each question (with the respective standard deviations:0, 0.65, 0.94, 0.45). From these results and from the children's attitude, one may conclude that the children were very enthusiastic about the activity and were curious about the next one. One may also conclude that the children were used to working (or playing) with computers and that they liked the application concept.

3.2.2 Second Session - Blind Man's Bluff

3.2.2.1 Before the session

The second session started being prepared right after analysing the first one results. The purpose of this session was to have a better idea of the children's conceptualization of the application

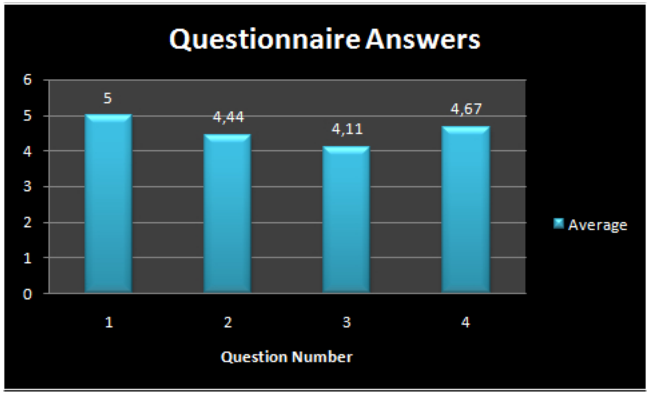


Figure 3.9 Children's answers to the questionnaire

design and interaction.

In order to do that a team of four people was gathered so that each member of the team would work with a group of children separately (two groups of two and two groups of three).

The main idea was to introduce the children to two new concepts: the helper and the "magic table". The helper would be "something" (an animal, a person, an object,..) that would help them through the game construction process. The "magic table" would be a table which could "tell" the helper which objects the children put on it. The team also introduced the idea that at the end of each construction phase there would be an object that had to be put on the table to tell the helper that they were ready to go to the next phase: the passage marker.

The main questions that should be answered in this session are:

1. How should the helper look like?
2. What object should the passage marker represent?
3. Should the instructions be given in an audio or text way ?
4. Did the children understand the concept of the game creation introduced in the previous session?

3.2.2.2 During the session

The children were divided into four groups. In each group one children was assigned to imagine how the interface would look like, and the rest (one or two, depending on the group), were in charge of drawing what the first one is imagining (and describing). Initially the activity was explained and the children were asked to choose which one would be imagining the construction. In each phase the children had to imagine both the visual style of the application and the objects that should be put on the table. The children that were drawing had some paper sheets previously prepared to each phase.

Once finished, each group would discuss if the imagined interface had anything to do with the drawn one. The entire process should be supported by the team member in charge.

The team member would act as facilitator in order to guide the children through the process of construction (defined based on the first session's result).

One of the groups wasn't able to finish the activity, but the others had spare time to discuss the results (the difference between what they've imagined and the drawings).

It wasn't easy to explain the children the concepts we wanted to introduce. The ones to which it was harder to explain were mainly the younger ones.

3.2.2.3 After the session

The results of this session were not as easy to draw as the ones from the first session. The answers to the questions were:

1. *How should the helper look like?* - The answer to this question could not be concluded from the session' results.
2. *What object should the passage marker represent?* - Even though no absolute answers

can be drawn, one can conclude that the passage marker should be related to the helper since in all groups they had a strong connection (mainly were objects that belonged to the helper.

3. *Should the instructions be given in an audio or text way?* - Since both the options were chosen by all groups, one can conclude that both are important and, when possible, both should be present).
4. *Did the children understand the concept of the game creation introduced in the previous session?* - Three of the four groups perceived the concept from the first session and no further explaining was needed. The only group that didn't, understood it after some further explanation.

One also noticed that although no recurrent sentences were used, the language used by the children to define what the helper said was informal, direct and simple.

3.2.3 Third Session - Warp Speed Design

3.2.3.1 Before the session

The main idea of this session was to simultaneously show the children the results of their work from the previous sessions and test the results the team had gathered at this point. In order to do that a prototype of an application that helps the children to construct a game was shown to the children. The prototype consisted of an application with the following phases:

1. The helper introduces himself and the application.
2. The helper asks the children to tell the game story to the microphone.
3. The helper asks the children which kind of character she prefers (girl,boy,cat or dog).
4. The helper asks the children to define the features of the previously chosen character.

5. The helper asks the children to choose one of the four possible scenarios.

Since the second session didn't allow to draw conclusions relating the helper character, one decided to use the concept of a magician (Figure 3.10(a)), which has a magic table (where the children put the objects) and a magic wand (the passage marker the children use to tell the magician they are ready to go to the next phase - Figure 3.10(b)).

(a) The Wizard (b) The Magic Wand

Figure 3.10 The concept chosen from the application.

The main questions to be answered in this session were:

1. Are the concept of the magic table and helper easy to understand?
2. Do children find the interaction method intuitive and fun or is it hard or boring to use?
3. To which instruction output (audio or text) do children pay more attention to?
4. Do children feel like they've helped and can they find their suggestions incorporated in the prototype?

Although all their choices were saved, the children weren't able to see the resulting game (the one they created) since, at this point, a full working prototype wasn't considered necessary.

3.2.3.2 During the session

Keeping in mind the requirements drawn at the beginning of the chapter, this application was tested with pairs of children so they would be able to collaborate with each other. Hence, the children had to discuss the choices they had to make (path, character,..) and often struggled to be the ones to put the pieces with the markers on the table. A facilitator was present to help the groups through the construction process.

All the children were disappointed by not being able to see the resulting game. Some children asked to repeat the construction in order to create different games with different histories, characters and paths.

Figure 3.11 Children testing the prototype at the third design meeting

3.2.3.3 After the session

Even though no formal questionnaire was made, one could conclude, by the observation of the children's behaviour when using the application, that:

1. The children remembered both the concepts of the helper and the magic table, and identified them as soon as they appeared.
2. None of the children presented problems with the interaction and, as stated before, they often struggled with each other to decide which one got to put the markers on the table.
3. Children payed more attention to the audio comments or tips and only recurred to the texts when they forgot what the "helper" had told them to do.
4. Some children found that the options they had, mostly regarding the character, were there because of their choices in the first session. Some also commented: "Look, this is my cat!".

The main difficulties the children encountered during the construction process appeared due to the fact that the children had no prior preparation and didn't know exactly what to do. They appeared mostly in the presentation part, where the children were asked to tell the story of the game to microphone.

There were four available paths and mostly the children chose the most complex one.

3.3 Meetings' Results

Some important conclusions were taken from these three session meetings and laid the bases of the application design. The concept of the magician, magic table and magic wand have shown to be easily understandable for the children.

Regarding the construction path, it was defined by the children that the most logical way to create the application was following the path: Story, Character, Scenario and Questions/Answers.

Although initially one thought about letting the children choose only between a boy or a girl character, after the sessions one decided to give them the option to also choose a cat or a dog (since one noticed, on the first session, that some children chose them).

Regarding the type of instructions given, although no conclusion was reached some examples of instructions given by the children in the Blind Man's Bluff activity were taken into account, and one used the same type of language they did (informal, simple, direct).

One very important finding was that in every phase, every instruction given to the children should always exist in both audio and text forms, as well as any important information (per example, the passage between phases), since mostly children listen to the audio instructions but, if they don't, they recur to the written texts to understand what they are supposed to do.

The creation of a game requires some preparation for both the story and questions/answers part (basically the ones that require audio recording). Also it would be important if the questions would be carefully prepared in order to create a challenging game(if it is too easy children will loose interest).

As for the chosen markers, no difficulties were found with the children's interaction, so one is lead to believe that as long as the image shows clearly what they represent, children understand what they mean.

Although the answers to the questionnaires can't be taken into account as absolute indicators, they showed that the children were interested and motivated. The fact that, in the third session, they struggled to put the object of the table and that they asked to repeat the construction (even though no final result could be shown) led the team to believe that the construction process wasn't boring.

Creating a multi-player game or an interface where children can work in groups, has happened with this one, allow them to socialize which was a very important requirement of a children's application. Hence, the final prototype will be tested with groups of children as well (and they will produce a multi-player game).

3.4 System Architecture

Even though the game creation is the most important part of this research one also had to think about the best way to implement the game itself.

Since its' interaction is keyboard based no special concerns or requirements had to be thought of. Also, as a game, it should be as visually attractive and have as many animations as possible. In order to do this, one decided to implement it in ActionScript.

The final solution reached was as shows in Figure 3.12. Each of the two applications has a different architecture, although the game architecture is very simple.

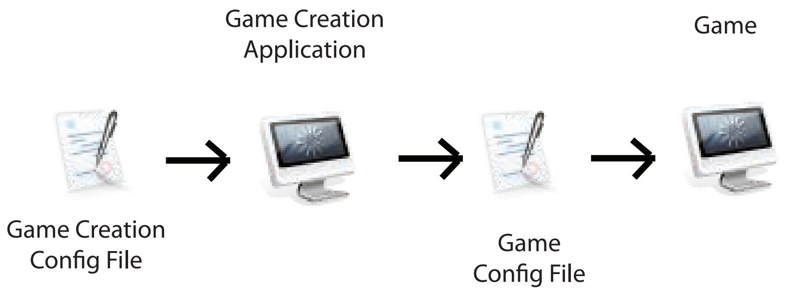


Figure 3.12 System Flow

Hence, two different architectures were created, one for each logical part. The architecture of the game creation application is the one shown in Figure 3.13 and is divided into:

- Input Devices - The devices that provide the data input by the children and communicates it to the layer above it: the camera will send it to reacTIVision, the microphone to the audio module and the keyboard directly to the application.
- Audio Module - The module that receives the data from the microphone and records it in the database.
- ReaTIVision - The module from the chosen platform that receives the data from the camera, analyses it and send the present markers to the application.

- Database - Where all the necessary data is stored in order to be used in the game.
- Application - Receives the data from the TUIO module and acts accordingly (sending the respective data to the GUI and sending commands to the other modules, when needed).
- Flash Module - The module that, along with to the GUI, allows the children to get the immediate feedback in the Character selection phase, when choosing the character's features.
- GUI - The visual part of the system, where the helper, as well as the visual instructions and the necessary immediate feedback is shown to the children.

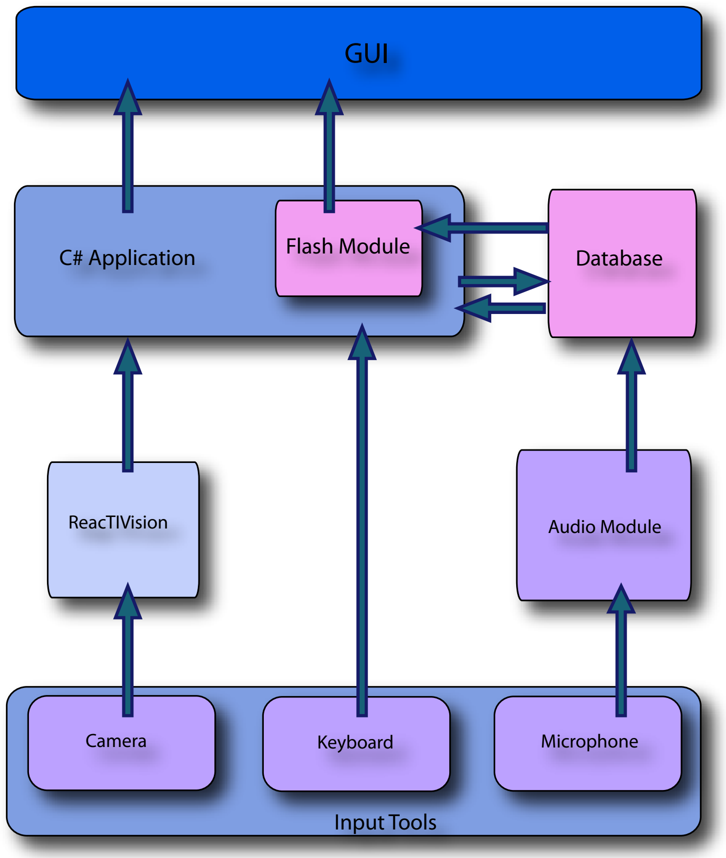


Figure 3.13 Game Creation Application Architecture

The game's architecture, in Figure 3.14 is only composed by:

- Keyboard - The input device used by the children to roll the dice and choose the answer from the possible options.
- GUI - The visual part of the system, where the game is presented to the children as well as the necessary immediate feedback.
- Application - Receives the data from the keyboard and acts accordingly until all players reached the end.

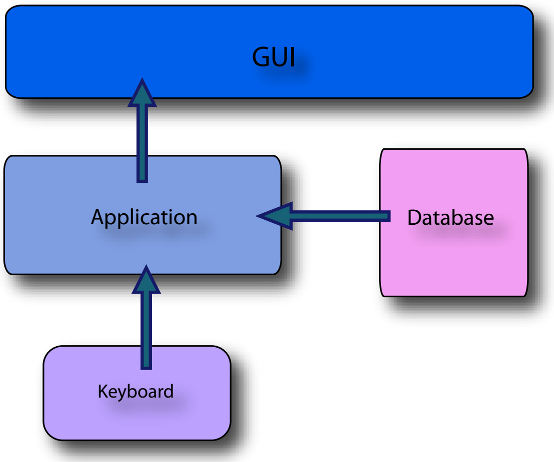


Figure 3.14 Game Architecture

4 . Methodology - Implementation

In order to develop the system regarding all the requirements, one decided to divide the system into two different applications (since their requirements are also completely different): the application that allows the game creation, and the game itself. Their organization is shown in Figure 3.12.

4.1 Table

In the spirit of pervasive computing, one had to create the interface in a way that the children saw the minimum necessary. In order to do that the table/box idea appeared.

The main components in our system (1 to 4 refer to the numbers in the Figure 4.1) are:

- (1) - Display - The visual aspect of the application, what is shown to the children.
- (2) - Camera - Sends the images to the reacTIVision, so that they can be processed.
- (3) - Light - Illuminates the box, so that the image captured by the camera is nitid enough to recognize the markers
- (4) - Table/Box - Where the children put the objects and interact with the system.
- (Hidden) - Microphone - In order to record the audio needed(in the prototype the story and questions).
- PC - Which assembles and computes all the data.

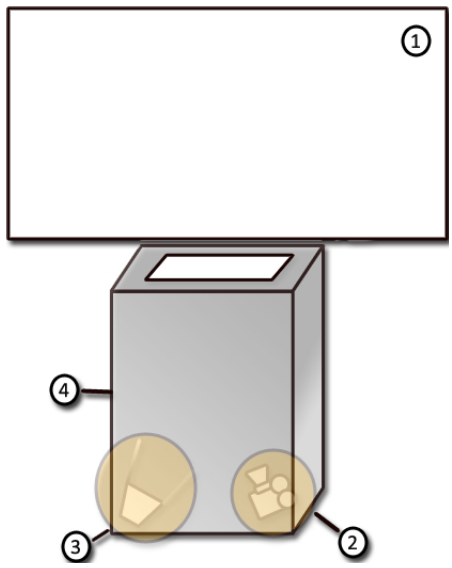


Figure 4.1 The table construction

4.2 Game Creation Application

4.2.1 Configuration File

In order to organize the application configuration data, one decided to store it in a xml file, that follows the schema shown in Figure 4.2. The xml allows a user to change the visual aspect as well as some general specifications of the application.

The xml has the following children:

- name - The application name, which will appear as the window title.
- passageMarker - The marker that will be used to go from one phase to the next one.
- bgImg - The background image common to the whole the application that will be present during all the construction process.
- intro, story, character, characteristics, scenario, question and end- the application's construction phases.

- removeMarkersWaiting - The definition of what should happen when, between phases, children have to remove all objects from the table top.
- audioRecordOptions - The general audio options.

Regarding the available options of the audio parts, the user may define the markers that will be used to record or stop recording, the images that appear on the screen when each of these markers is detected and the location of these images on the screen.

Regarding the construction phases there are some common parameters to define in each phase, as well as others that are specific to one phase. The general parameters that can be defined are illustrated in Figure 4.3, which refers to the introduction phase, since it is a phase that only has the general parameters, which are:

- (attribute) id - The phase's id.
- nextPhase - The phase that is next to the current one (defined by the id).
- waiting - Boolean that defines if the application should wait that the children remove all the visible objects before going to the next phase.
- disableGeneralMarker - Boolean that defines if, when showing the passage marker, the application should go to the next phase.
- hasAudioPart - Boolean that defines if that phase has an audio part.
- hasSpecialMarkersPart - Boolean that defines if that phase includes any part that uses special markers or any special behaviour. There were defined two different special markers parts: flash and image. The flash part will be explained later on, but it is essentially a part where the application communicates with a flash module. The image part is a simple part where, for each marker there is an associated image and, as the camera finds one of the markers, that image is immediately shown on the GUI.

- hasOtherMarkers - Boolean that defines if there are other markers that should be considered for the current phase only.
- elements - List of elements (image or audio) that are displayed (or played) when the application enters in that specific phase.

The only phases that have specific parameters are the "character" and the "questions" ones. In the first one the user may specify the number of characters the children will define as well as if the character has any characteristics to be created. On the questions phase, the user may define the number of questions to be defined, the questions' input format, the number of options and the options' input format.

4.2.2 TUIO Protocol and reacTIVision

ReacTIVision is a framework that implements the TUIO protocol and is prepared to send messages from both found markers and finger tracking. Seeing that in our application all we need is the markers' recognition, and not even their position is needed, there were only two functions we had to worry about in order to implement a TUIO client that receives messages from ReacTIVision and keeps track of the objects on the table:

- addTuioObject - function that is triggered every time the camera detects a new marker. Regarding the application, this happened every time an object is put on the table. This function is the motor of the application for almost every events that occur when a children puts a new object on the table.
- removeTuioObject - function that is triggered every time the camera detects the removal of a marker from the table. This event was essential to the phases in which the application had to wait that the children removed all the objects in order to proceed to the next one.

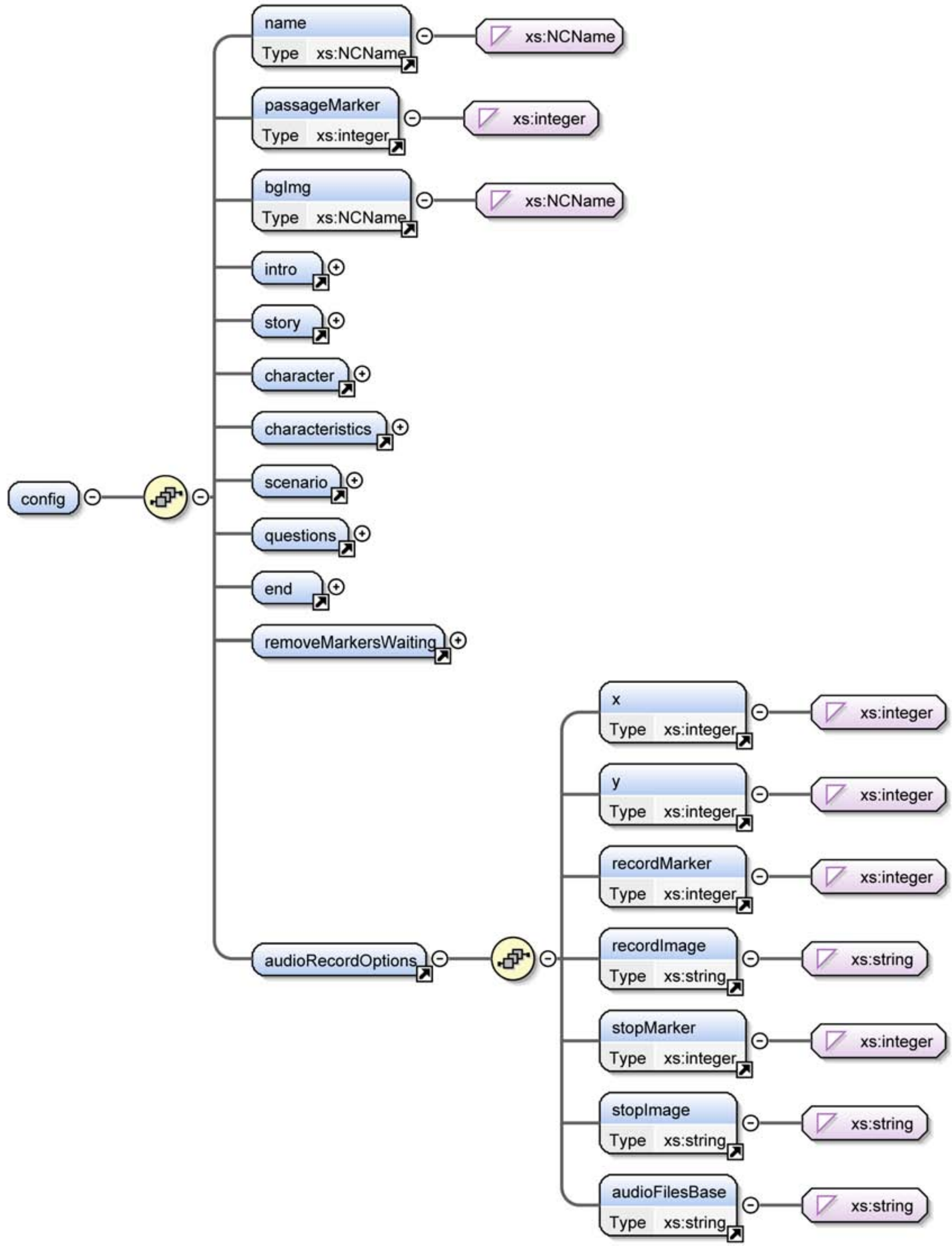


Figure 4.2 Game Creation Application Schema

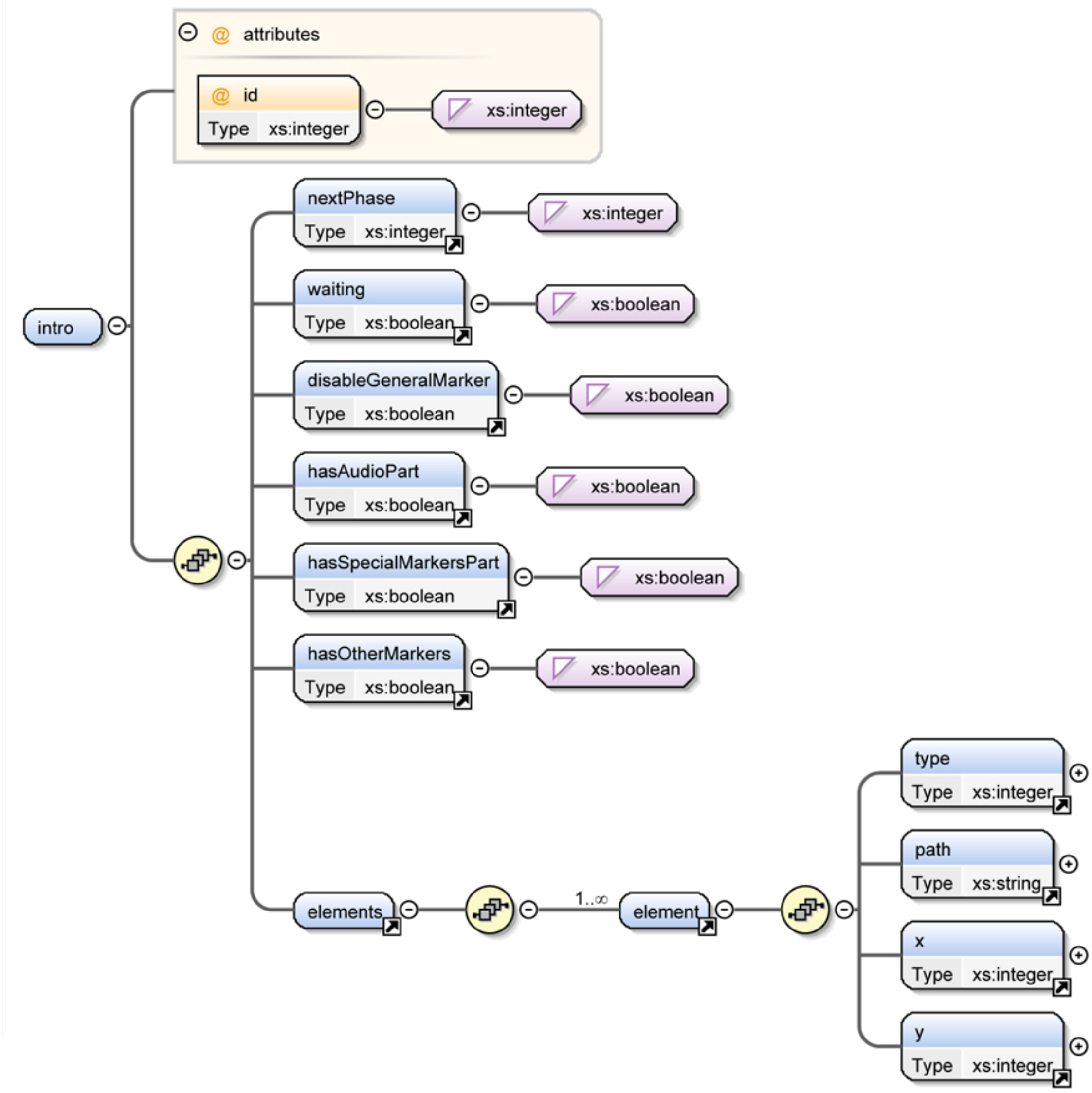


Figure 4.3 Schema of the general features that must be defined for each phase (introduction phase presented as example)

4.2.3 Flash Communication

In order to provide the necessary feedback, some phases required the use of the flash technology. In this application the character's specification phase required that each time a child chose a new feature for his/her character it would be instantly displayed. This could have been done recurring simply to images, but it would mean having one image for each of the possible combinations of features.

The communication of the application with the actionScript module is possible when recurring to a Shockwave ActiveX control - "axShoxkwaveFlash". Sending data to the flash component is done like follows:

```
private void sendDataToFlash()
{
 string outText = "<invoke name=\"sendDataToFlash\"
   returntype=\"xml\"><arguments><string>Hello From CSharp</string></
       arguments></invoke>";
  axShockwaveFlash.CallFunction(outText);
}
```

In ActionScript, receiving the data from the external C# is also simple:

```
ExternalInterface.addCallback("sendDataToFlash", getDataFromCSharp);
public function getDataFromCSharp(data:String):void {
  trace("received data from csharp... "+data);
}
```

Sending data back to C# is done by:

```
public function sendToCSharp(text:String):void {
  var result:Object = ExternalInterface.call("sendtoCSharp", "Hello From
      CSharp");
}
```

Back to C#, receiving the sent message from Flash is as easy as:

```
private void player_FlashCall(object sender,
    _IShockwaveFlashEvents_FlashCallEvent e)
{
```

```
    XmlDocument document = new XmlDocument();
    document.LoadXml(e.request);
    XmlAttributeCollection attributes = document.FirstChild.Attributes;
    String command = attributes.Item(0).InnerText;
    XmlNodeList list = document.GetElementsByTagName("arguments");
    switch (command)
    {
      case "sendtoCSharp": System.Console.WriteLine("Received From Flash: "+
          list[0].InnerText);
        break;
      default: System.Console.WriteLine("The data received is in an unknown
          command"); break;
    }
}
```

For the created game construction application, this communication is only used during the character selection phase, when children choose their characters' features.

Once they've chosen which character will be their character, the application sends this information to the actionScript module which displays it. For each marker put on the table (recognized by the camera, and sent to the application via reacTIVision) the application checks if the marker represents a character's feature and, if it does, sends its code to the actionscript module, so that the character is shown with the recently chosen characteristic.

Once the children is finished (the magic wand marker is put on the table) the application sends a simple finish message to the actionscript that sends all the chosen features back to our application (to be stored in the game configuration file).

4.2.4 Application Flow

The application has seven major types of functions: read data, write data, display, keyboard, audio, flash and tuio. The application classes are structured as shown in Figure 4.4 .

It starts by reading the configuration file and storing the necessary data. When all the data is read, the TUIO client starts and the intro is shown. With this, the game construction starts and goes as follows:

1. Introduction: the introductory screen appears (Figure 4.5(a)) and the application waits until the passage marker is put on the table. When it happens, the current phase is hidden and the next one is shown.
2. Story: the story screen (Figure 4.5(b)) is displayed and the application waits until the children put the record marker on the table in order to start recording. When this happens, the application waits for the stop marker. The children can repeat the record process as many time as he/she wants to. Once the passage marker is put on the table the application goes to the next phase (on the condition that at least one recording has been done).
3. Character: in this phase (Figure 4.5(c)) the children only have to choose the character they want to, by putting the correspondent marker on the table. As soon as they do the next screen appears.
4. Characteristics: In this screen (Figure 4.5(c)) the children choose how their character looks like. Through the previously mentioned communication with the Flash module, children have an immediate feedback concerning their character's features. Once the passage marker is put on the table the application goes to the next phase (unless there are more characters to be defined and, in that case, the application goes back to the character screen).

5. Scenario: In this phase (Figure 4.5(e)) the children only have to choose one of the possible game scenarios. With the passage marker the next phase is shown.
6. Questions: This is probably the most complex phase (Figure 4.5(f)). First the application waits for the record and stop markers in order to record the question. Once the children puts the passage marker the first option (the right one) is highlighted in green, so that the children know that the application is waiting for the correct option. Then the four options can be typed and, once they're done, the application goes to the next phase (unless there's some question left to record, in which case it starts this phase all over to allow the insertion of a new question).
7. End: This screen (Figure 4.5(g)) appears to let children know the construction is over. As soon as the screen is displayed all the data chosen by the children is written in the game configuration file, which will be explained in the next section.

4.3 Game

4.3.1 Configuration File e Application Flow

According to the settings established by the children during the game creation process, a xml game file configuration is produced, which has the following children:

- story - stores the path to the audio file.
- players - stores, for each player, its name and the necessary data related to its' parts and respective colors chosen by the children.
- scenario - stores the className, in Flash, of the path chosen, the number of houses in the path, the background image of the scenario and the begin house(by default it is 0).

- questions - stores the input format of both questions and options (text, image, audio) and, for each question, the question itself(or its' path, if the format is audio or image) and its' options.

Since no external modules are needed to create the game, its' classes structure is rather simple(Figure 4.6), as well as its flow: initially the data is read from the configuration file and, when it's finished, it displays the game board, the miniatures of the players, and waits that the "ENTER" key is pressed. When it happens, the dice rolls and a random number from 1 to 6 appears on the screen. The current player goes forward to the respective house. If this house is a questions' one, a question is displayed (or played) as well as its' corresponding answers. The application then waits that one of the keys 'a','b','c' or 'd' is pressed and checks if it is the correct option. If it is, it updates the player score and if it isn't that player goes back to where he was before the dice was rolled. After this, it is the turn of the next player and the application is, again, waiting for the "ENTER" key to be pressed. The game ends when all the players reached the end of the path. The classes needed to implement this application are illustrated in the classes structure in Figure 4.7.

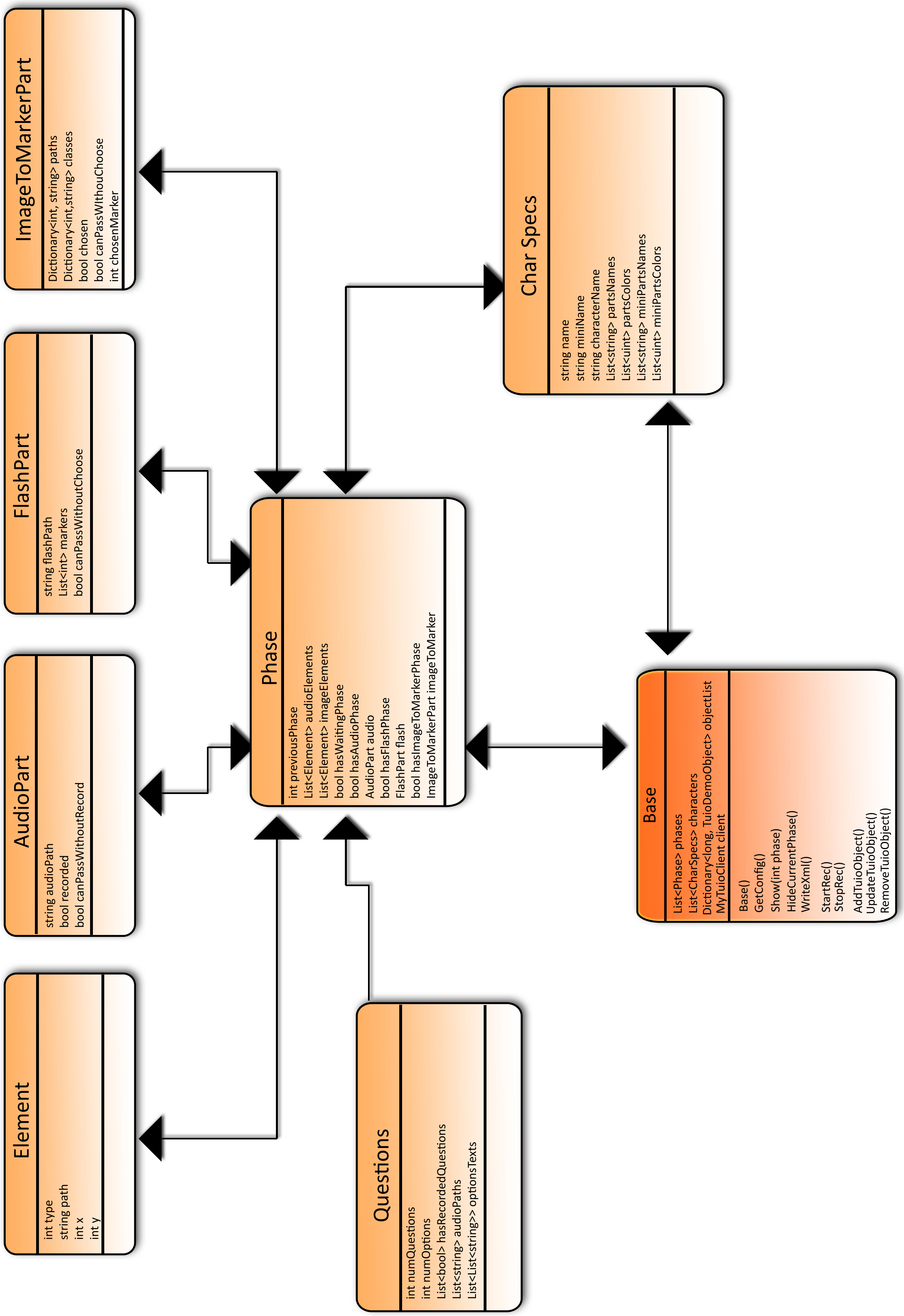


Figure 4.4 Game Creation Application Classes

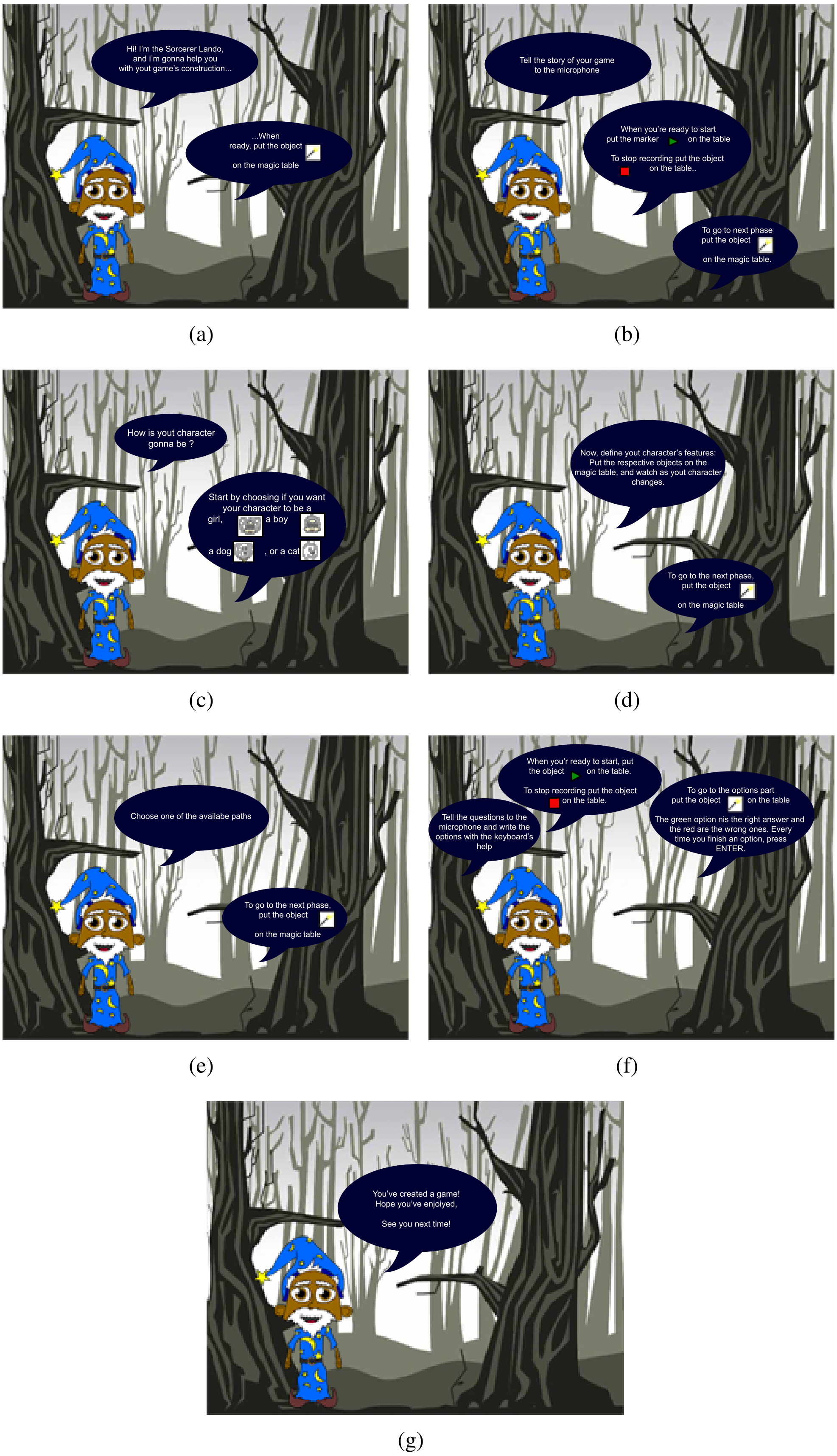


Figure 4.5 Screenshots from the construction phases

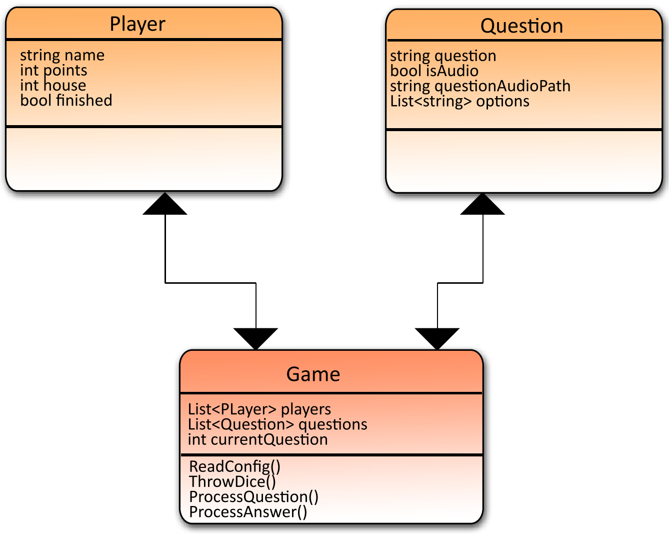


Figure 4.6 gameClasses

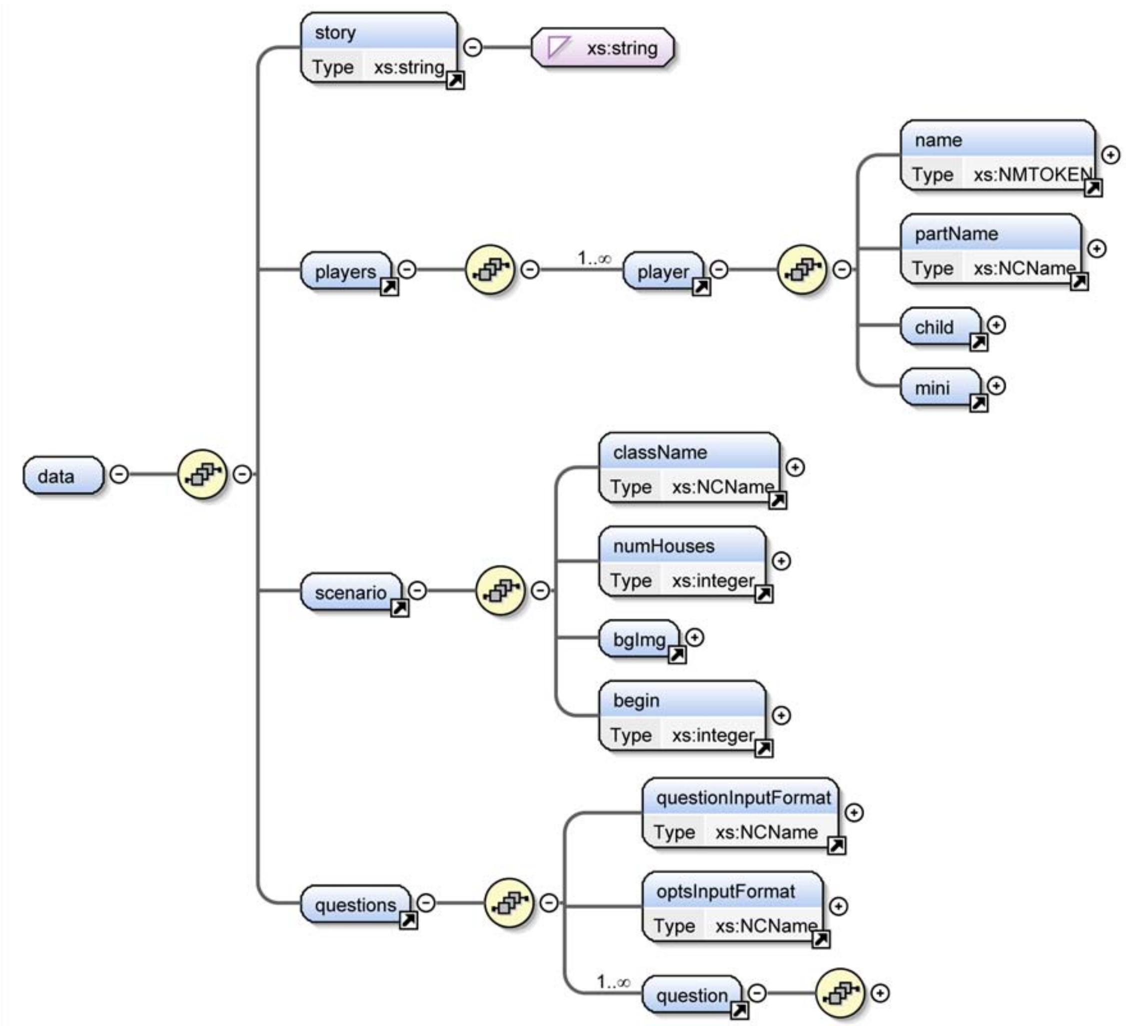


Figure 4.7 Game Configuration File Schema

5 . Tests, Results and Analysis

In usability engineering, as explained in Section 2.2, for each design interaction the logical step that follows the implementation is testing the created application. There are many ways to do that, and some are specifically created to test children' applications. The team decided to both observe the children interacting with the application and ask them to answer a brief questionnaire.

5.1 Behind the tests

As the focus of this thesis is to create both an application that motivates children to learn and an application design that is intuitive to them (hence their heavy participation in the design phase), one had to structure the tests in order to be able to draw some conclusions relating these goals.

By observing the children interacting with the application one could evaluate the usability and intuitiveness of the application. The questionnaire's purpose was to evaluate their motivation once they have created and played their (and others') games.

From the requirements mentioned in Section 3.1, one concluded that the application should promote children's collaboration and social interaction. Hence, one decided to create a multiplayer game. Since there were ten children, they were divided in 5 groups of 2. The children's ages were comprised between 8 and 10 years old and weren't the same ones that participated in the design process (they had no prior contact or knowledge of the application).

In order to be able to observe and study both the system's usability and the children's motivation, the test session consisted of five different parts: Demo, Preparation, Game Creation, Play and Questionnaire. Each one will be further explained in the following sections.

5.1.1 Demonstration of a previously created game

As the children arrived to the test, the team explained them what was the purpose of the test and, in order to give them an idea of what could be done with the application, showed them a previously created game. Since it was a two-player game the children were divided in two groups so that they could play the example game with each other.

5.1.2 Preparation of the story and questions/answers

After the demo the children were given some scholar books they have used during the current scholar year and were asked to prepare a game about it. Each group of two children was asked to create an introduction/story of their game as well as 6 questions and the respective right and wrong answers.

(a) (b)

Figure 5.1 The children preparing their game

In this phase the effort that children put into preparing the questions, looking for the answers and finding three more options that fit the question's context could be an indicator of their motivation, and even though it is not quantifiable, it can be observed by the team.

5.1.3 Creation of the game

Only after the group had the game fully prepared the creation started. They were asked to create a full game with intro, scenario, character and questions/answers. An observer was present, who took notes of the children's evolution through the game creation. The game creation was divided into four scenarios (as shown in Figure 5.2) and each one had from 2 to 4 tasks the children had to accomplish to go to the next one.

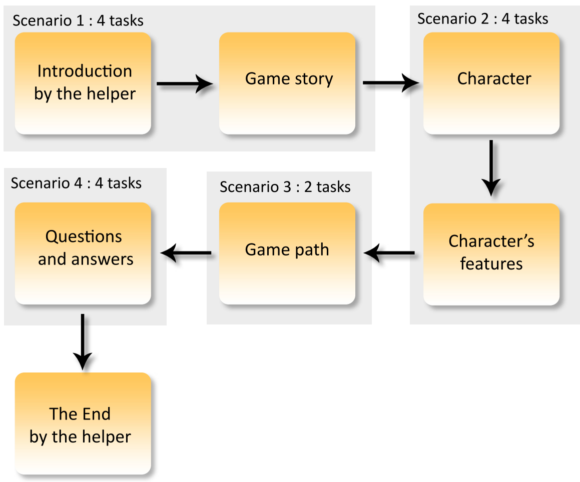


Figure 5.2 Game creation flow and scenarios with tasks children had to accomplish

5.1.4 Play the created game

Once the game creation was finished, the children were anxious to see the final results. The present groups played the game, sometimes against the one that created it(which always also wanted to play). The children enjoyed playing a game they created as well as seeing other children playing and enjoying their game.

(a) (b) (c)

Figure 5.3 Children creating their game

5.1.5 Questionnaire

At the end of the session the children had to, individually, answer a small multiple-choice questionnaire. Two of the questions used the Smileyometer[32]. The questionnaire comprised the following five questions:

1. *What did you think of today's activity?* - There were five possible answers to this question, that went from "Very Uninteresting"(1) to "Very Interesting"(5), using the smileyometer as a visual aid.
2. *Which part did you like more?* - This question's purpose was to find out whether the children preferred to create the game or to play it. The possible answers were: "Create the game", "Play the game", "None" and "Both equally".
3. *Imagine you could create another game, but you may choose between creating it as you just did (with the objects and the magic table) or in a computer (with a mouse). Which would you chose?* - The purpose of this question was to find out if the children liked and preferred to interact through a tangible interface. The possible answers were, obviously, "Create the game as I just did" and "Create it in a computer, with a mouse". In this question the facilitator also asked the children why they have chosen that option.

4. *If you had this program, do you think you'd use it to create games to your family/friends/others?* - This question's purpose was to find out if the children would use the program if they had access to it. The possible answers were "Never" , "Few times","Maybe","Sometimes" and "Many times". In this question the smileyometer was also used.
5. *You just created a game with a friend of yours. Would you prefer to create it by yourself?* - This question's purpose was to find out if the option to test the application with pairs of children was, from their point of view, the correct one. There were only two answers to this question:"Yes" and "No".

At the end of the questionnaire, the team member helping the children asked them if they had any suggestions to make the application better.

5.2 Testing and Results

5.2.1 Preparation of the story and questions/answers

In this part the children were given the classes manual of a previously chosen subject. They were given the books of their grade, so they could relate to them and easily look for the subjects they preferred. ´

Even though the team expected the children to find this phase boring and expected them to hurry they were really interested in exploring the subject. They looked very interested and carefully prepared the game story, questions and answers. This motivation to prepare the questions and find the answers is the key to learning the concepts.

5.2.2 Creation of the game

This part started by introducing the children the concepts of the magician(helper) and the magic table. The purpose of the magic wand marker, as an object related to the magician was also explained (the only way the magician had to know how they were ready to go to the next phase was if they put the magic wand on the magic table).

5.2.2.1 Introduction and Story

In the introductory screen the children were only asked to put the magic wand marker on the table. Next, the magician (helper) asked them to record their story with the microphone. They were, for the first time in the game, confronted with the sound markers. There were only two: one to start recording, and one to stop. If they wanted to, they could repeat the recording. Once they were finished they had to put the magic wand on the table. There were four tasks they had to accomplish to go to the next phase, which the children(80%) easily completed. As can be seen in Fig 5.4 the results were good. The main problem in this scenario was the fact that most times children didn't remember to put the stop marker on the table, to stop recording.

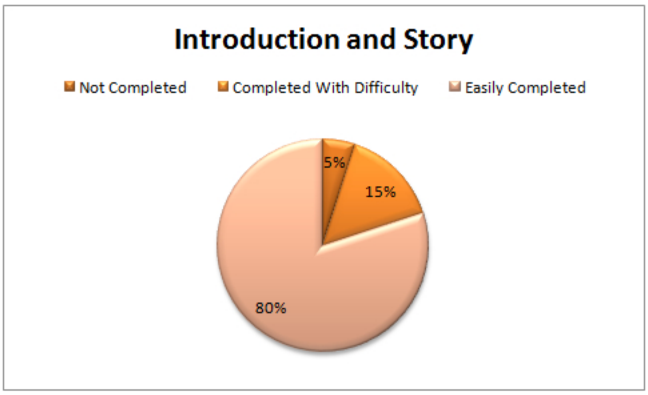


Figure 5.4 Introduction and Story Usability Results

5.2.2.2 Character

After the creation of the story the children were asked to create two characters (one by each member of the group). They could choose between a girl, a boy, a cat or a dog. Once they made their choice they were asked to define the chosen character's features.

They enjoyed the fact of being able to try different options before making a decision and, most of the times, tried all the available ones before deciding.

The results in Fig 5.5 are much alike the previous ones. This time the problems appeared due to the fact that children wanted to choose the character's features before having chosen the character.

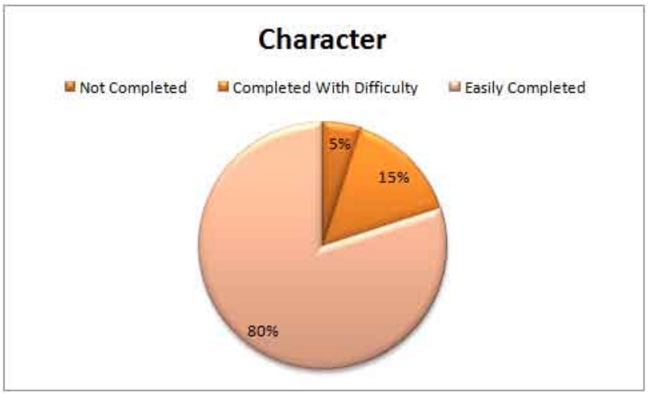


Figure 5.5 Character Usability Results

Because of the character

5.2.2.3 Scenario

To compose the scenario of the game the children had to accomplish two tasks: choosing the preferred path and putting the magic wand on the table. As can be seen in Fig 5.6 it was rather easy, no problems were found.

The children also tried all the possible scenarios to see how they looked like, before choosing the one they wanted for their game.

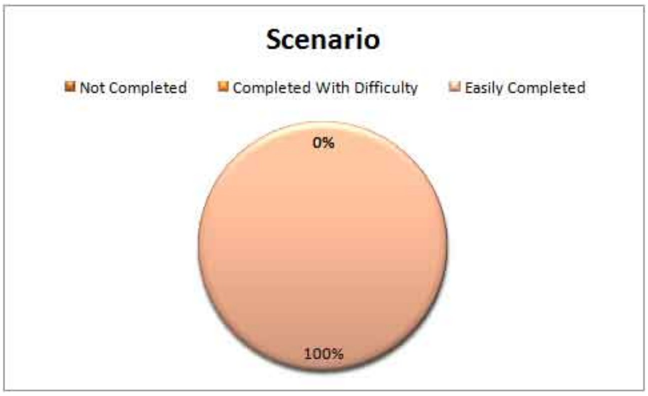


Figure 5.6 Scenario Usability Results

5.2.2.4 Questions

This was the more complex phase. For each question the children were asked to record the question and put the magic wand on the table to start typing the answers with the keyboard (first the correct and next the wrong ones). To go from one option to another they could either press 'Enter' or put the magic wand on the table (all the children chose to press "ENTER" since they were already using the keyboard, seemed more natural).

In order to explain it to the children, there were three balloons with text on the screen and the explaining audio was rather long. However, most children didn't pay much attention to the audio instructions and only after reading they knew what to do.

Recording the questions was rather easy once they understood the purpose of each marker.

The results are shown in Fig 5.7 and they aren't as good as the previous ones mainly due to the fact that, as expected, the children didn't have the dexterity necessary to use the keyboard. They had to look for the keys and often asked someone for help with the writing. It took them long to accomplish this phase, even though they didn't seem to matter.

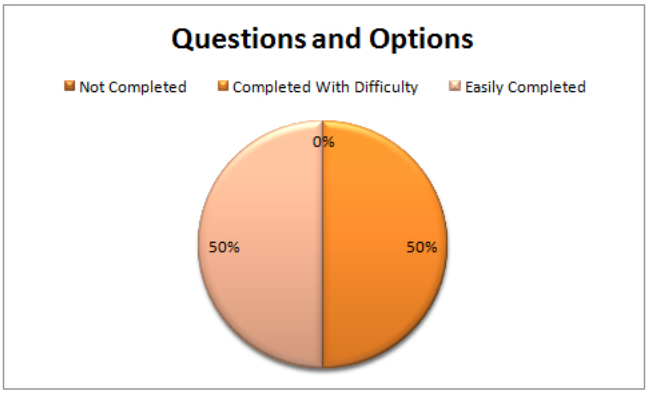


Figure 5.7 Questions and Answers Usability Results

The idea of giving the children the option to also choose a cat or a dog as their character appeared due to the first design meeting's analysis. In order to find out if the idea was correct, the character's chosen in the test were registered by the observer: 5 of the 10 children chose an animal and the other 5 a person as their character.

5.2.3 Questionnaire

The questionnaire part was the last one of the test session. The team member who posed the questions to the children was one of their teachers, so they would feel more comfortable when answering and giving opinions and suggestions.

5.3 Results' Analysis

During the game construction children had to input data in three different ways: using the markers recognized by the camera, audio through a microphone and text through a keyboard.

Once the children understood the concepts of the objects, the magic table and the magic wand, no further problems were found with their interactions, regarding the markers.

Besides the natural discomfort children have when talking to a microphone and the audio markers' concepts (play and stop), children intuitively understand and know how to work with a microphone, so this kind of interaction presented no problems.

The keyboard data input, however, presented some problems since, as expected, children don't have the needed dexterity to use the keyboard as adults do. It took them long time to create the part that required this type of input (answers) and they often asked the present team member for help with the text input. However long it took, they didn't seem to care, which led to the believe that they were having fun.

One may also conclude that the audio and text instructions are important and should always both be present.

Regarding the questionnaire, even though the results are very good, they can only be taken into account if keeping in mind that children often answer the option they think would please most the person in charge. In order to draw better conclusions, more children should test this application.

Regarding the first question, "What did you think of today's activity?", the ten children answered "Very Interesting" - 5, on a scale from 1 to 5 (Figure 5.8(a)). Even though the conclusion that the created application is very interesting, the children's motivation and general opinion of the work can somehow be analysed.

As to the second question, "Which part did you like more?", most children (8 of 10 - Figure 5.8(b)) answered that they liked both parts(creating and playing the game). The remaining 2 children answered that they preferred to create the game than to actually play it.

The third question's purpose was to study which type of interaction was preferred by the children - the one they had just experienced or a mouse-based one, which they are used to when playing in their computers. 8 out of 10 children chose the tangible interaction over the mouse-based one (Figure 5.8(c)).

As for the fourth question, "If you had this program do you think you'd use it to create games to your family/friends/others?", the results (Figure 5.8(d)) weren't as optimistic as the other ones, but 6 in 10 children answered they would many other times, if they could, and 4 answered that sometimes they would, even though not frequently.

The last question was created merely to understand if children preferred to work alone or in a group (in this case, a pair). The results, which can be seen in Fig 5.8(e) show that children absolutely prefer to work in group. In this case it was actually expected since the research/preparation part would have been kind of boring if done alone and, since the final game can be multiplayer, it's easier if each one creates their own character.

Finally, with the character's chosen by the children one could verify the efficacy of the design meetings, since before them the idea that children would choose animal of persons as game characters never crossed the mind of the design team.

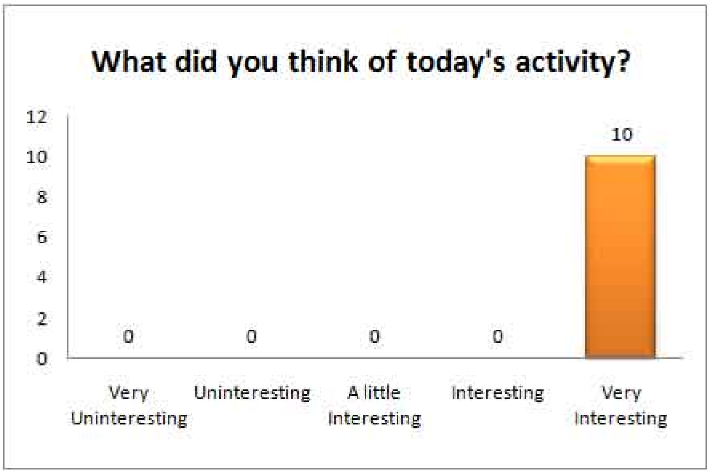
As for the reasons the children gave for preferring a tangible interface over the tradicional mouse-based interaction, some of the given ones were:

- Because its' fun
- Because its' easy
- Because its' a new interface

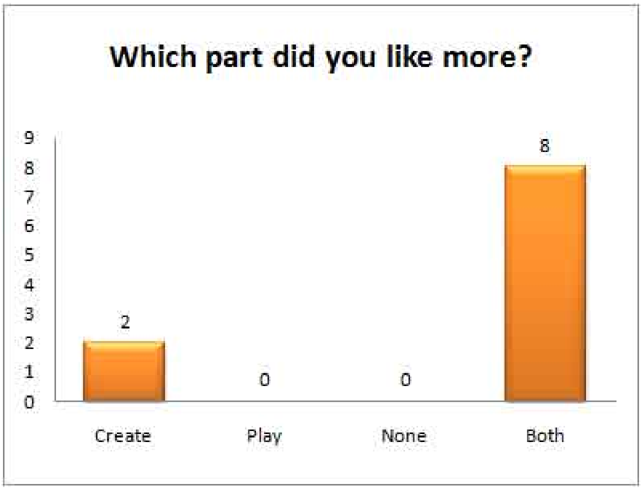
The suggestions they gave for the application's improvement were:

- More characters to choose
- More scenarios options
- More houses in the scenarios (to make the game longer)
- More questions

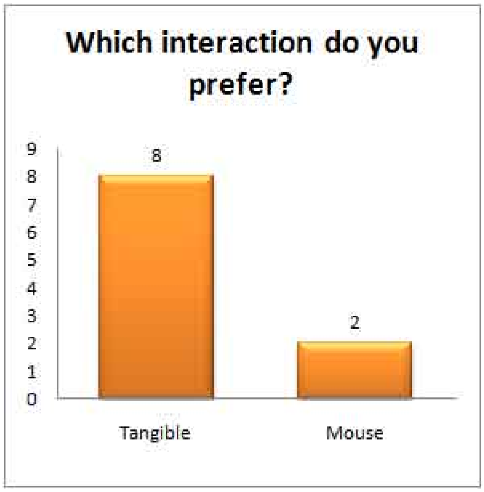
- To have objects with letters to write the words



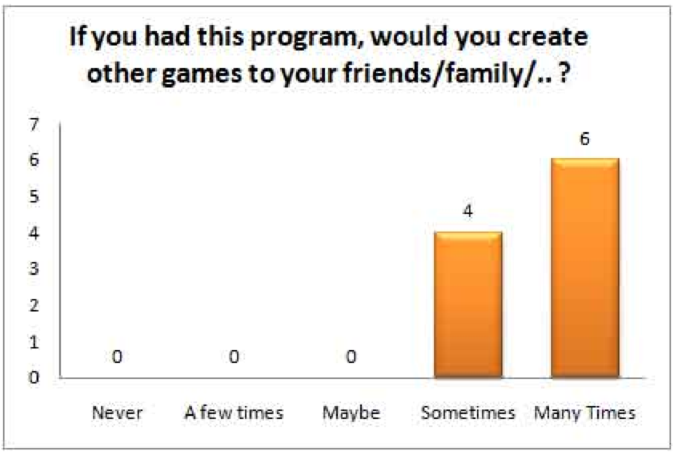
(a) First Question's Results



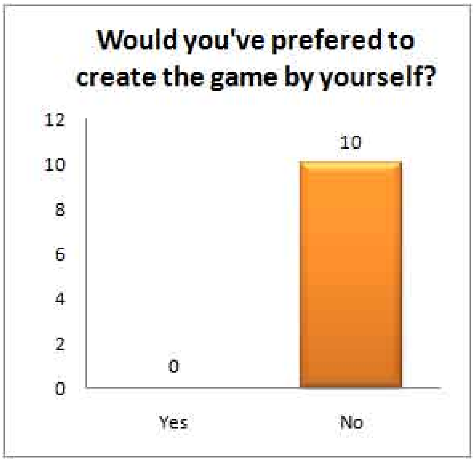
(b) Second Question's Results



(c) Third Question's Results



(d) Fourth Question's Results



(e) Fifth Question's Results

Figure 5.8 Questionnaire results

6. Final Considerations and Future Work

People have become dependents on technology and children are an emergent users group to whom special attention should be paid.

Since what mainly defines the cognitive development of a person is based on the education he/she has received as children, engaging children into learning is (or should be) a major goal to achieve when creating educational software.

Children see the world in a different way adults do and haven't got yet the necessary dexterity to interact with some physical devices, special concerns arise.

This work's goal was to create a tangible application that enabled children to create their own educational games motivating them to learn about the topics covered by the game.

Design meetings with children were conducted in order to better understand the children's ideas and conceptions of an engaging and intuitive interface.

The created application allows children to create their own games using a tangible interface system. The games created by the application are like "The Game Of Life" but with multiple choice questions in some of the houses (as the "Trivial Pursuit").

The prototype was tested with other children that weren't included in the application's design process.

6.1 Final Considerations

Regarding the design meetings with the children, many questions were answered and many useful data was concluded from them:

- On the first session, the children were asked to draw their game's main character, and one noticed that 3 of the 10 children chose an animal (cat and dog were the ones chosen). Due to these results the final prototype also gave the children that option. When confronted with the choice between a person (girl or boy) or an animal (dog or cat), 5 of the 10 children chose a person and the other 5 chose an animal.

- The application should always contain both audio and text instructions.
- The interaction with the objects and the table seemed natural and required no learning.
- The construction of a game requires some preparation from the children (this preparation is the key for the learning process).
- If the objects represented by the markers have a clear purpose, no interaction problems appear.
- The instruction to the children should be informal, clear and direct.
- The fact that the children struggled to put the marker on the table showed that they were motivated to use the tangible interface.
- Enabling the game creation in groups of children allows them to collaborate and socialize with each other.

These conclusions confirmed the need and usefulness of the children's inclusion in the design process, as well as the effectiveness of the chosen method: *Bluebells method*. Most of the questions that initially appeared were answered by these meetings.

The usability tests' results were positive. Regarding the usability one concluded that the children presented no problems with the user interface created and the only difficulties appeared due to their lack of dexterity to input the text using the keyboard. Analysing the opinions and suggestions they gave when answering the final questionnaire one may conclude they were motivated by the use of a new interaction method and the general opinion was that it was fun and easy. Most children suggestions were related to improving the game creation giving them more options (more characters, scenarios, questions,..) which may also be considered a motivation indicator. The concepts of the markers and the table were easily understood and the interaction with the table seemed natural.

Although no conclusions about learning can effectively be drawn, the fact that children were engaged into preparing their game, reading the text books given, elaborating the questions and looking for the answers is an indicator that they were in fact motivated, while learning the contents necessary to create their game.

6.2 Future Work

In order to fully study the children's engagement and effective learning while using this kind of application that allows them to build question/answer games longer studies should be done. One good project was to test this application in school field trips: dividing them into two groups, one that was previously told of the game creation and one that wasn't. The final game construction by both groups would shed lights to the children's motivation, studying the difference between the attention paid by the children of the first and the second groups. A longer study about the effective learning would also be useful.

It would also be valuable to study methods for children's text input that do not imply the use of a keyboard, which they have some difficulty to use.

Finally, conducting more(and even longer) design meetings would allow to deepen the knowledge of what the children think to be an intuitive application, of what makes sense to them.

Future work can also include a deeper study of the design methods in order to contribute for the improvement and adaptation to specific situations.

Exploring and comparing other types of educational games (adventure, action, puzzle, ..)

may also be useful in order to find out which type better engages them into learning.

Bibliography

[1] *ACM SIGCHI curricula for human-computer interaction*. New York, NY, USA, 1992.

[2] Alissa Nicole Antle. Child-personas: fact or fiction? In *DIS '06: Proceedings of the 6th conference on Designing Interactive systems*, pages 22–30, New York, NY, USA, 2006. ACM.

[3] Ronald T. Azuma. A survey of augmented reality. *Presence*, 6:355–385, 1997.

[4] Mark Billinghurst, Raphael Grasset, and Julian Looser. Designing augmented reality interfaces. *SIGGRAPH Comput. Graph.*, 39(1):17–22, 2005.

[5] Mark Billinghurst, Hirokazu Kato, and Ivan Poupyrev. The magicbook: a transitional ar interface. *Computers and Graphics*, 25:745–753, 2001.

[6] B Boehm. A spiral model of software development and enhancement. *SIGSOFT Softw. Eng. Notes*, 11(4):14–24, 1986.

[7] Jenova Chen. Flow in games (and everything else). *Commun. ACM*, 50(4):31–34, 2007.

[8] Sonia Chiasson and Carl Gutwin. Design principles for children's technology. Technical Report HCI-TR-2005-02, Computer Science Department, University of Saskatchewan.

[9] Barbara Moskal & Deborah Lurie & Stephen Cooper. Evaluating the effectiveness of a new instructional approach. In *SIGCSE '04: Proceedings of the 35th SIGCSE technical symposium on Computer science education*, pages 75–79, New York, NY, USA, 2004. ACM.

[10] M. Covington. Goal theory, motivation, and school achievement: An integrative review. *Annual Review of Psychology*, v51:171–200, 2000.

[11] Elizabeth Debold. Flow with soul. *What is Enlightenment?*, pages 108–118, 2002.

[12] M. Dougiamas. A journey into constructivism. 1998.

[13] Alison Druin. The role of children in the design of new technology. *Behaviour and Information Technology*, 21(1):1–25, 2002.

[14] Allison Druin. Cooperative inquiry: developing new technologies for children with children. In *CHI '99: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 592–599. ACM Press, 1999.

[15] Olga Dzianbenko, Maja Pivec, Christos Bouras, Vaggelis Igglesis, Vaggelis Kapoulas, and Ioannis Misedakis. A web-based game for supporting game-based learning. In Quasim H. Mehdi, Norman E. Gough, and Stéphan Natkine, editors, *GAME-ON*, pages 111–. EUROSIS, 2003.

[16] Jerry Alan Fails, Allison Druin, Mona Leigh Guha, Gene Chipman, Sante Simms, and Wayne Churaman. Child's play: a comparison of desktop and physical interactive environments. In *IDC '05: Proceedings of the 2005 conference on Interaction design and children*, pages 48–55, New York, NY, USA, 2005. ACM.

[17] Mona Leigh Guha, Allison Druin, Jaime Montemayor, Gene Chipman, and Allison Farber. A theoretical model of children's storytelling using physically-oriented technologies (spot). *Journal of Educational Multimedia and Hypermedia*, 16(4):389–410, October 2007.

[18] Maehr ML & Meyer HA. Understanding motivation and schooling: Where we've been, where we are, and where we need to go. *Educ Psychology Rev*, 9:371–409, 1997.

[19] Tony Hall and Liam Bannon. Designing ubiquitous computing to enhance children's interaction in museums. In *IDC '05: Proceedings of the 2005 conference on Interaction design and children*, pages 62–69, New York, NY, USA, 2005. ACM.

[20] HITLabNZ. Black magic book. `http://www.hitlabnz.org/wiki/Black_Magic_Book`, 2002.

[21] Hiroshi Ishii and Brygg Ullmer. Tangible bits: towards seamless interfaces between people, bits and atoms. In *CHI '97: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 234–241, New York, NY, USA, 1997. ACM.

[22] Martin Kaltenbrunner and Ross Bencina. reactivision: a computer-vision framework for table-based tangible interaction. In *TEI '07: Proceedings of the 1st international conference on Tangible and embedded interaction*, pages 69–74, New York, NY, USA, 2007. ACM.

[23] Martin Kaltenbrunner, Till Bovermann, Ross Bencina, and Enrico Costanza. Tuio - a protocol for table based tangible user interfaces. In *Proceedings of the 6th International Workshop on Gesture in Human-Computer Interaction and Simulation (GW 2005)*, Vannes, France, 2005.

[24] Martin Kaltenbrunner, Sergi Jorda, Gunter Geiger, and Marcos Alonso. The reactable*: A collaborative musical instrument. In *WETICE '06: Proceedings of the 15th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises*, pages 406–411, Washington, DC, USA, 2006. IEEE Computer Society.

[25] S Rebecca Kelly, Emanuela Mazzone, Matthew Horton, and Janet C Read. Bluebells: a design method for child-centred product development. In *NordiCHI '06: Proceedings of*

the 4th Nordic conference on Human-computer interaction, pages 361–368, New York, NY, USA, 2006. ACM.

[26] Roscoe R. & Chi M. Understanding tutor learning: Knowledge-building and knowledge-telling in peer tutors' explanations and questions. *Review of Educational Research*, 77(4):534–574, 2007.

[27] John H. Maloney, Kylie Peppler, Yasmin Kafai, Mitchel Resnick, and Natalie Rusk. Programming by choice: urban youth learning programming with scratch. *SIGCSE Bull.*, 40(1):367–371, 2008.

[28] Paul Milgram and Fumio Kishino. A taxonomy of mixed reality visual displays. *IEICE Transactions on Information Systems*, E77-D(12), December 1994.

[29] S. Papert. Constructionism vs. instructionism. 1980.

[30] Seymour Papert. *Mindstorms : Children, computers and powerful ideas.* Basic Books, New York, second edition, 1993.

[31] Jean Piaget. *The construction of reality in the child; translated by Margaret Cook.* Ballantine New York,, 1954.

[32] Janet C. Read. Validating the fun toolkit: an instrument for measuring children's opinions of technology. *Cogn. Technol. Work*, 10(2):119–128, 2008.

[33] Janet C. Read. Warp speed design: a rapid design method for use with children. In *CHI EA '09: Proceedings of the 27th international conference extended abstracts on Human factors in computing systems*, pages 4681–4686, New York, NY, USA, 2009. ACM.

[34] R. Roscoe, J. Wagster, and G. Biswas. Using teachable agent feedback to support effective learning by teaching. *The Thirtieth Annual Meeting of the Cognitive Science Society*, pages 2381–2386, July 2008.

[35] I. Sutherland. The ultimate display. In *Proceedings of the International Federation of Information Processing (IFIP)*, pages 506–508, 1965.

[36] Ivan E. Sutherland. A head-mounted three dimensional display. pages 295–302, 1998.

[37] Mark Weiser. The computer for the 21st century. pages 933–940, 1995.

[38] wikipedia. Cognition. `http://en.wikipedia.org/wiki/Cognitive`, November 2008.

[39] wikipedia. Seymourpapert. `http://en.wikipedia.org/wiki/Seymour_Papert`, December 2008.