Project: Extending VITHEA in order to improve children’s linguistic skills

Vânia Patrícia Padrão Mendonça

Thesis to obtain the Master of Science Degree in

Information Systems and Computer Engineering

Supervisors: Prof. Maria Luísa Torres Ribeiro Marques da Silva Coheur
Prof. José Alberto Rodrigues Pereira Sardinha

Examination Committee

Chairperson: Prof. Mário Rui Fonseca dos Santos Gomes
Supervisor: Prof. Maria Luísa Torres Ribeiro Marques da Silva Coheur
Member of the Committee: Prof. Daniel Jorge Viegas Gonçalves

November 2015
Abstract

An Autism Spectrum Disorder (ASD) is a developmental disability characterized by impairments in social communication and interaction that often comprises difficulties in the acquisition of verbal language skills. In order to improve the quality of life for individuals with ASD and given their documented interest in computers, several researchers and companies have developed applications that aim to provide an alternative to verbal communication or a way for children with ASD to develop linguistic skills, necessary for verbal communication. However, there is a lack of applications in Portuguese that are tailored to the individual needs of each child.

In this context, we present VITHEA-Kids, a platform that accounts for the needs of different children, as well as their caregivers, by providing an interface where caregivers can create exercises and customize various aspects of the interaction with the platform. We also developed a module that allows the automatic generation of multiple choice exercises, meant to be integrated in VITHEA-Kids.

Our platform is under evaluation on multiple fronts: a preliminary evaluation with caregivers (which provided promising indicators), an evaluation with a child (ongoing) and an evaluation of the generation of incorrect answers (distractors) in multiple choice exercises (which resulted in acceptance rates between 61.11% and 92.22%).

Keywords: Autism Spectrum Disorder, children learning, language skills development, automatic generation of content
Uma Perturbação no Espectro do Autismo (PEA) caracteriza-se por dificuldades na comunicação e interacção em situações sociais, as quais surgem frequentemente associadas a dificuldades na aquisição de linguagem verbal. De forma a melhorar a qualidade de vida dos indivíduos com [PEA] e dado o interesse que estes manifestam por computadores, vários investigadores e empresas desenvolveram aplicações que procuram servir como meio de comunicação alternativa ou possibilitar que crianças com [PEA] desenvolvam competências linguísticas, necessárias para a comunicação verbal. Contudo, há um défice de aplicações deste tipo em Português que permitam ter em conta as necessidades individuais de cada criança.

Neste contexto, apresentamos o VITHEA-Kids, uma plataforma que procura ir de encontro às necessidades de cada criança e dos seus cuidadores, disponibilizando uma interface em que os cuidadores podem criar exercícios e personalizar diversos aspectos da aplicação. Desenvolvemos ainda um módulo que permite a geração automática de exercícios de escolha múltipla, pensado com o intuito de ser integrado no VITHEA-Kids.

A plataforma desenvolvida encontra-se a ser avaliada em várias frentes: uma avaliação preliminar com cuidadores (que deu indicadores promissores), uma avaliação com uma criança (a decorrer) e uma avaliação relativa à geração de respostas incorrectas (distractores) em exercícios de escolha múltipla (que resultou em rácios de aceitação entre os 61.11% e os 92.22%).

**Palavras-chave:** Perturbação no Espectro do Autismo, aprendizagem por parte de crianças, aquisição de competências linguísticas, geração automática de conteúdo
While one page lacks the space needed to thank to everyone that contributed in some way for this thesis, there is a set of people to whom I owe my thanks for all the support during this journey.

First of all, my greatest thanks to my parents, Joaquim Mendonça and Maria do Rosário Padrão; if it wasn’t for their support, love and sacrifices, this thesis would have never happened (nor the journey that led to it).

A huge thank you to my advisors, Prof. Luísa Coheur and Prof. Alberto Sardinha, for all the support, advices and for always believing in me (even when I doubted myself). Additionally, I’d like to thank Prof. Luísa for all the inspiration and for introducing me to the field of natural language processing, back in the BSc days, and to Prof. Alberto for all the valuable input regarding the topic of this thesis.

A word of thanks to all the therapists and caregivers who tested VITHEA-Kids and provided valuable feedback (specially to Célia Cunha, who also ran the evaluation with the child application) and to everyone who tested the exercise generation module. I would also like to thank Anna Maria Pompili for all the help regarding VITHEA and to Pedro Fialho for bearing with my newbie questions at L2F.

I am also very, very thankful to all the amazing friends and colleagues I had the privilege of meet at both campi of IST (or through people from IST), either for their support, fun times, help, advice, patience, dedication to NEIIST (specially during my mandate, which overlapped this thesis), proof reading my reports or helping with bug solving. The list is too long, so I will focus in the ones that have bear with me the most during this year or overall: Soraia Meneses Alarcão, Ricardo Laranjeiro, Bernardo Santos, Marta Cardoso, Joana Condeço, Fábio Alves, David Silva, Diva Lima, Nuno Barrocas, Maria João Aires, João Murtinheiro, Fernando Alves, André Pires, Miguel Pires, Ana Salta, Miguel Coelho, Margarida Alberto, Ricardo Carvalho, Catarina Moreira, Inês Fernandes, Denise Pedro, Rita Gomes, Patricia Santos, Francisco Freire, Diogo Barradas, Diogo Lachica, Pedro Pires, Tomás Alves, Manuel Alves, David Duarte, Jorge Heleno, Nuno Xu, Renato Vieira, Rui Fabiano, Andrea Ferrão, Élvio Abreu, Rúben Rebelo, Tiago Santos, Jorge Martins, Gonçalo Sousa, Mauro Brito, Daniel Dias, André Pedroso and many, many others who might not be listed, but are definitely not forgotten! Also, thanks to Anisa Shahidian for the help and enthusiasm regarding VITHEA-Kids, and to Cláudia Patrícia Filipe for all the motivation to continue this work – it will be an honour to work together with both of you and make this project grow even more!

Working on this thesis has complicated the task of meeting my friends outside IST, who supported me nonetheless, so here go my thanks to all of them, specially to Telma Brás, Inês Santos, Tânia Ribeiro, Nelson Mendes, Iris Santos, Duarte Coelho, André Bento, David Quadrado, Gonçalo Guerra, and also to my second family: Manuela Rato, Helena Robalo, Miguel Margarido and my goddaughter Constança.

Last but not least, I’d like to thank to all the teachers that have inspired me through all these school years, specially to my first programming teacher, Prof. Natacha Pereira, whose words of encouragement and challenge followed me through all these years at IST. Having in mind those words, I can say this thesis has been quite a “jump” (and I do not intend stop “jumping”).
To my parents: Rosário and Joaquim;
In memory of my grandparents: Teresinha, Augusta, Manuel and Abílio
## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td>iii</td>
</tr>
<tr>
<td>Resumo</td>
<td>iv</td>
</tr>
<tr>
<td>Acknowledgments</td>
<td>v</td>
</tr>
<tr>
<td>List of Figures</td>
<td>xiv</td>
</tr>
<tr>
<td>List of Tables</td>
<td>xv</td>
</tr>
<tr>
<td>List of Acronyms</td>
<td>xvii</td>
</tr>
</tbody>
</table>

## 1 Introduction

1.1 Motivation ............................................. 1  
1.2 Goals .................................................... 2  
1.3 Solution .................................................. 3  
1.4 Contributions and results ............................. 3  
1.5 Document outline ...................................... 4  

## 2 Background

2.1 Autism Spectrum Disorder ............................. 5  
2.2 Approaches for individuals with ASD ................. 7  
  2.2.1 Applied Behaviour Analysis (ABA) .................. 7  
  2.2.2 Therapies: Floortime and Son-Rise ................ 9  
2.3 Interventions .......................................... 9  
  2.3.1 Picture Exchange Communication System (PECS) ...... 9  
  2.3.2 Treatment and Education of Autistic and Communication-handicapped Children (TEACCH) .......... 10  
2.4 Summary ............................................... 10  

## 3 Related Work

3.1 Technology for children with ASD .................... 13  
  3.1.1 Characterization of user needs .................... 14  
  3.1.2 Software for children with ASD .................. 17  
3.2 Automatic generation and processing of content ...... 27  
  3.2.1 Automatic generation of multiple choice exercises 28  
  3.2.2 Resources and tools ............................... 28  
3.3 Summary ................................................. 30
4 VITHEA-Kids: A platform for children with ASD and their caregivers

4.1 VITHEA - Virtual Therapist for Aphasia Treatment

4.1.1 Word naming exercises
4.1.2 Therapist’s module
4.1.3 Patient’s module
4.1.4 Architecture and underlying technology

4.2 From VITHEA to VITHEA-Kids

4.2.1 Multiple choice exercises
4.2.2 VITHEA-Kids’ conceptual model
4.2.3 Caregiver’s module
4.2.4 Child’s module

4.3 Summary

5 Automatic generation of multiple choice exercises

5.1 System overview

5.1.1 ExerciseGeneration module
5.1.2 WordImageLookup module

5.2 Generation process

5.2.1 Correct answer generation
5.2.2 Question generation
5.2.3 Distractor generation

5.3 Summary

6 Evaluation

6.1 VITHEA-Kids

6.1.1 Evaluation with caregivers
6.1.2 Evaluation with a child

6.2 Automatic generation of exercises

6.3 Summary

7 Conclusions and Future Work

7.1 Summary of the Dissertation
7.2 Final Conclusions and Contributions
7.3 Future Work

7.3.1 VITHEA-Kids
7.3.2 Automatic Generation of Exercises

Bibliography

Appendices

Appendix A Software for children with ASD

A.1 Communication software
A.2 Educational software

A.2.1 Studies regarding the use of software to improve language skills
A.2.2 Educational applications for the development of language skills

Appendix B ExerciseGenerator: Usage instructions
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Proloquo2go’s interface</td>
<td>18</td>
</tr>
<tr>
<td>3.2</td>
<td>Simone Says’ interaction loop</td>
<td>20</td>
</tr>
<tr>
<td>3.3</td>
<td>Language Player’s interface, with the character Baldi on the left</td>
<td>21</td>
</tr>
<tr>
<td>3.4</td>
<td>Example of an interaction in Camp Discovery</td>
<td>23</td>
</tr>
<tr>
<td>3.5</td>
<td>Example of an interaction in Vizzle Player</td>
<td>23</td>
</tr>
<tr>
<td>3.6</td>
<td>Troc@s’ categories screens</td>
<td>24</td>
</tr>
<tr>
<td>3.7</td>
<td>Exercise types</td>
<td>26</td>
</tr>
<tr>
<td>3.8</td>
<td>Customization possibilities</td>
<td>26</td>
</tr>
<tr>
<td>4.1</td>
<td>Virtual Therapist for Aphasia Treatment (VITHEA)’s therapist’s module</td>
<td>33</td>
</tr>
<tr>
<td>4.2</td>
<td>VITHEA’s patient’s module</td>
<td>34</td>
</tr>
<tr>
<td>4.3</td>
<td>VITHEA’s information flow</td>
<td>35</td>
</tr>
<tr>
<td>4.4</td>
<td>A representation of the VITHEA’s layered architecture</td>
<td>36</td>
</tr>
<tr>
<td>4.5</td>
<td>Architectural view of VITHEA’s mobile version</td>
<td>36</td>
</tr>
<tr>
<td>4.6</td>
<td>Caregiver’s module main screen</td>
<td>39</td>
</tr>
<tr>
<td>4.7</td>
<td>Creation of a new exercise</td>
<td>40</td>
</tr>
<tr>
<td>4.8</td>
<td>Exercise management</td>
<td>41</td>
</tr>
<tr>
<td>4.9</td>
<td>Resource creation and management</td>
<td>41</td>
</tr>
<tr>
<td>4.10</td>
<td>Creation and management of child users</td>
<td>42</td>
</tr>
<tr>
<td>4.11</td>
<td>Animated character’s utterances customization</td>
<td>43</td>
</tr>
<tr>
<td>4.12</td>
<td>Reinforcement images customization</td>
<td>44</td>
</tr>
<tr>
<td>4.13</td>
<td>Child’s module screens</td>
<td>45</td>
</tr>
<tr>
<td>4.14</td>
<td>Communication between the child application and the server</td>
<td>46</td>
</tr>
<tr>
<td>5.1</td>
<td>Overview of the automatic exercise generation architecture</td>
<td>48</td>
</tr>
<tr>
<td>5.2</td>
<td>Architecture of the ExerciseGeneration module</td>
<td>49</td>
</tr>
<tr>
<td>5.3</td>
<td>Architecture of the WordImageLookup module</td>
<td>49</td>
</tr>
<tr>
<td>5.4</td>
<td>Overview of the automatic exercise generation process</td>
<td>51</td>
</tr>
<tr>
<td>5.5</td>
<td>Question generation</td>
<td>53</td>
</tr>
<tr>
<td>6.1</td>
<td>Results on user satisfaction regarding each task</td>
<td>59</td>
</tr>
<tr>
<td>6.2</td>
<td>Results regarding overall experience and interest on new features</td>
<td>60</td>
</tr>
<tr>
<td>6.3</td>
<td>Measurements during baseline (A), intervention (B) and follow-up (A) phases</td>
<td>62</td>
</tr>
<tr>
<td>6.4</td>
<td>Exercise to be solved in the baseline phase</td>
<td>62</td>
</tr>
<tr>
<td>6.5</td>
<td>Distractor evaluation tasks</td>
<td>64</td>
</tr>
<tr>
<td>6.6</td>
<td>Global results for adequacy and quality of the word distractors</td>
<td>65</td>
</tr>
<tr>
<td>6.7</td>
<td>Global results for adequacy and quality of the image distractors</td>
<td>65</td>
</tr>
</tbody>
</table>
List of Tables

3.1 Features considered in Lehman’s survey [39] .................................................. 15
3.2 Example of a sequence of Simone Says’ exercises ............................................. 20

4.1 Example of an exercise in VITHEA .................................................................... 33
4.2 Examples of multiple choice exercises ................................................................. 37

6.1 Results for word distractor’s quality ................................................................. 65
6.2 Results for image distractor’s quality ................................................................. 65
6.3 Results for quality of the word distractors related to a given topic .................. 66
6.4 Results for quality of the word distractors without a topic constraint ............. 66
6.5 Results for quality of the image distractors related to a given topic ................. 67
6.6 Results for quality of the image distractors without a topic constraint .......... 67

A.1 Comparison of applications for communication purposes ............................... 78
A.2 Studies regarding the use of software for language skills development purposes ... 79
A.3 Comparison of applications for the development of linguistic skills ............... 80
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Full Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABA</td>
<td>Applied Behaviour Analysis</td>
</tr>
<tr>
<td>ACM</td>
<td>Association for Computing Machinery</td>
</tr>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
</tr>
<tr>
<td>ASD</td>
<td>Autism Spectrum Disorder</td>
</tr>
<tr>
<td>ASR</td>
<td>Automatic Speech Recognition</td>
</tr>
<tr>
<td>CSS</td>
<td>Cascade Style Sheets</td>
</tr>
<tr>
<td>CSV</td>
<td>Comma-separated Values</td>
</tr>
<tr>
<td>INESC-ID</td>
<td>Instituto de Engenharia de Sistemas e Computadores - Investigação e Desenvolvimento</td>
</tr>
<tr>
<td>JSON</td>
<td>JavaScript Object Notation</td>
</tr>
<tr>
<td>L2F</td>
<td>Spoken Language Systems Laboratory</td>
</tr>
<tr>
<td>MVC</td>
<td>Model View Controller</td>
</tr>
<tr>
<td>PEA</td>
<td>Perturbação no Espectro do Autismo</td>
</tr>
<tr>
<td>PECS</td>
<td>Picture Exchange Communication System</td>
</tr>
<tr>
<td>PoS</td>
<td>Part of Speech</td>
</tr>
<tr>
<td>REST</td>
<td>Representational State Transfer</td>
</tr>
<tr>
<td>SIGACCESS</td>
<td>Special Interest Group on Accessible Computing</td>
</tr>
<tr>
<td>TEACCH</td>
<td>Treatment and Education of Autistic and Communication-handicapped Children</td>
</tr>
<tr>
<td>URL</td>
<td>Uniform Resource Locator</td>
</tr>
<tr>
<td>VITHEA</td>
<td>Virtual Therapist for Aphasia Treatment</td>
</tr>
<tr>
<td>XML</td>
<td>Extensible Mark-up Language</td>
</tr>
</tbody>
</table>
Introduction

The present document describes a software system aimed at children diagnosed with Autism Spectrum Disorder and their caregivers. In this chapter, we present the motivation behind the development of this system, our goals, the main contributions of our work and the outline for the remainder of the document.

1.1 Motivation

Autism Spectrum Disorder (ASD) comprises a set of developmental disabilities characterized by persistent deficits in social communication and interaction, as well as restricted, repetitive patterns of behaviour or interests since an early developmental period [5]. Autism is not necessarily related to intellectual disability or global developmental delay [5], but its causes are still unknown, although some risk factors, such as certain genetic or chromosomal conditions, have been identified [12]. The most recent worldwide estimations (2012) point to a proportion of 17 in 10000 children with autism and 62 in 10000 children with other pervasive developmental disorders in the autism spectrum [22]. As for Portugal, a study performed in 2005 estimates that the prevalence of children between 7 and 9 years old diagnosed with an ASD is of approximately 9 in 1000 children for Continental Portugal and 16 in 1000 for Azores [2], again according to DSM-IV’s definition [49]. Furthermore, the number of reported cases has been increasing [13], and are four to five times more prevalent in male children [12,49].

Since individuals with ASD often face challenges regarding social and communication tasks, and some of them do not develop the ability to verbally communicate at all, or develop it at a much slower pace than their typically developing peers [5,35], there are therapies that aim to minimize these difficulties. These therapies are mainly based in Applied Behaviour Analysis (ABA), which relies on techniques such as providing help (prompting) or rewards (reinforcement) in order to influence a child’s behaviour [47,60]. Typically, interventions in the context of such therapies take place in a clinical or school setting; however, since therapy might not be affordable, investment in education for children with

1 considering the definitions from the fourth edition of the Diagnostic and Statistical Manual of Mental Disorders (DSM-IV)
2 Madeira was not covered by this study
special needs might be insufficient \cite{4,15,16,24} or therapy centres might not benefit from enough conditions \cite{25}, the possibility of taking therapeutic interventions to other settings or having affordable means to perform such interventions could be desirable.

Given the interest that individuals with ASD display towards computers \cite{39,57}, some authors placed their efforts on researching about the effect of using software for the purpose of teaching academic skills to children with ASD \cite{10,14,28,29,34,44,46,70}. Additionally, in the past few years, with the increasing popularity of mobile devices such as smartphones and tablets \cite{26,30–32}, both companies and researchers have invested in the development of educational mobile applications targeting children with impairments, thus easing the practice of verbal skills in diversified settings, with or without assistance. Despite the current large offer in terms of these applications, there are still important issues left to be addressed, namely:

- Most applications are paid or include paid features;
- There is a lack of applications available in Portuguese, specially for language development purposes;
- Very few applications take into account each user's progress, characteristics and/or needs, specially the ones for educational purposes.

Considering this situation, we aimed at developing a platform to tackle these issues in order to contribute for the improvement of the language skills of children with ASD.

1.2 Goals

The main goal of this work is to develop a software platform where children diagnosed with ASD can solve exercises in order to develop or improve a set of linguistic skills regarding the Portuguese language. It should also be possible for the children's caregivers (parents, teachers, therapists, etc.) to create the exercises to be solved. Our platform should:

- Be available for free, in Portuguese, with the possibility of being used in mobile devices, so that both children and caregivers can use the application either within and outside a therapy setting;
- Allow the child to solve multiple choice exercises featuring vocabulary and images;
- Have the possibility of using prompting to help the child while solving an exercise;
- Display a visual reinforcement when the child manages to correctly solve an exercise;
- Provide feedback through a talking animated character;
- Combine a set of features that aim to meet the needs of both children and caregivers by allowing an individualized intervention, namely:
  - Exercises with custom content;
  - Customization of interaction aspects such as the animated character’s utterances and the visual reinforcement.

We also aim to develop a module to automatically generate multiple choice exercises based on a given topic, and provide them either to the platform described above or to any other project.
1.3 Solution

In order to achieve these goals, we extended an existing platform with the functionality described above. The final result, called VITHEA-Kids, comprises two modules: one for the children and another for their caregivers. The child's module includes as features:

- A set of user-created multiple choice exercises for the child to solve;
- A talking animated character to utter the exercises and feedback regarding the child's answers;
- The possibility of using prompting when the child pick an incorrect answer;
- The possibility of displaying user-created reinforcement images when the child hits a correct answer.

As for the caregiver's module, it allows to:

- Create and manage multiple choice exercises of two types;
- Upload and manage multimedia resources to be featured in the exercises;
- Create and manage the users for the child's module;
- Customize the reinforcement images and the animated character's utterances.

In addition to this platform, we also developed a module that generates multiple choice exercises (composed by a question, a correct answer, and a set of incorrect answers, also known as distractors) given a certain topic, a question template and a set of constraints regarding the distractors. This module makes use of a second module that extracts words and images based on hierarchies of synonym sets.

1.4 Contributions and results

The main contributions of our work include:

- A platform for children with [ASD] and their caregivers, that comprises:
  - An application for children to solve multiple choice exercises aiming the development of linguistic skills;
  - An application for caregivers where they can create the exercises for the children to solve and customize certain aspects of the child's application.

- A module that automatically generates multiple choice exercises, to be integrated in the platform mentioned above or in any other project;

- A module to extract the words and images that are used by the exercise generation module in order to fill the exercises' content;

In addition to this, we had a poster about VITHEA-Kids and the automatic generation of exercises accepted to ASSETS 2015—an international conference on Computers and Accessibility sponsored by Association for Computing Machinery [ACM] and Special Interest Group on Accessible Computing [SIGACCESS].

[^1]: http://assets15.sigaccess.org/ (last access on 12/10/2015)
1.5 Document outline

The remainder of this document is structured as follows: in Chapter 2 we provide some background regarding Autism Spectrum Disorder (ASD) and related concepts; in Chapter 3 we review previous works on the use of technology for children with ASD and on automatic generation of exercises, as well as the tools and resources for that purpose; in Chapter 4 we present our solution: VITHEA-Kids, a platform for children with ASD to solve exercises in order to develop language skills, and for their caregivers to create those exercises; in Chapter 5 we present another part of our solution: an independent module for the automatic generation of exercises, to be integrated into VITHEA-Kids; in Chapter 6 we describe the procedures and present the results of evaluating each part that composes our solution; finally, in Chapter 7 we summarize the highlights of our work and present some future work possibilities.
In this chapter, we will introduce several concepts related to autism that will be mentioned in subsequent chapters. In Section 2.1 we present how Autism Spectrum Disorder (ASD) has been defined through the ages, from its discovery to current diagnosis criteria; Sections 2.2 and 2.3 provide details on the approaches followed in order for the individuals diagnosed with an ASD to improve their skills and overcome or minimize their difficulties.

2.1 Autism Spectrum Disorder

In terms of research, the first reference to autism belongs to psychiatrist Leo Kanner, in 1943. In his studies, Kanner observed 11 children (8 boys and 3 girls, all of them under 11 years old) who revealed symptoms of a condition that had never been reported until then [35]. Despite the peculiarities of each individual case, Kanner was able to identify a set of common characteristics, namely:

- Inability to relate themselves to people and situations and to develop social awareness, often acting as if people were not present and failing to assume an anticipatory posture (e.g., lack of reaction when their parents prepared to pick them up as babies). It was common for them to show more interest in interacting with objects than with people;

- Delayed development of speech and language skills, in contrast to an exceptional rote memory; children were often able to memorize rhymes, lists and foreign terms, but they were unable to adapt their discourse, usually limiting themselves to echoing (immediately or with a certain delay) previously heard words and sentences (a phenomenon that goes by the name of echolalia). For instance, they would not adjust the pronouns, thus referring to themselves as “you”, nor could they provide yes/no replies to different questions;

- High sensitivity to loud noises, moving objects, etc., which they considered intrusive unless these constituted objects of their interest;

- Display of an obsessive desire for the maintenance of sameness and completeness, which would lead them to feel desperate towards changes in their routine and environment or broken/incomplete
items. They also displayed a limited range of spontaneous activity, often engaging in repetitive, stereotyped behaviours;

- Good cognitive skills, despite an potential delay in the development of language skills.

In 1944, paediatrician Hans Asperger described a group of children who also exhibited a similar pattern of behaviour, namely difficulties in social interactions, a restrictive range of interests and repetitive behaviours. However, his observations differed from those made by Kanner in the sense that the children he described displayed a typical (and sometimes exceptionally good) development in what concerns to cognitive and language skills [8,71].

The definition of autism became part of the Diagnostic and Statistical Manual of Mental Disorders in its third edition (DSM-III), published in 1980 [69], while Asperger’s Syndrome was only featured in the fourth edition (DSM-IV) [33]. According to this edition, people displaying autistic behaviours would receive the separate diagnosis of Autism, Asperger’s Syndrome or Pervasive Developmental Disorder-Not Otherwise Specified (PDD-NOS). However, the most recent edition (DSM-V), released in 2013, places all of these concepts under the diagnostic of “Autism Spectrum Disorder” (ASD). DSM-V defines the following criteria for an [ASD] to be diagnosed [5]:

A “Persistent deficits in social communication and social interaction across multiple contexts”, such as social-emotional reciprocity (e.g., difficulties in social interaction and reduced sharing of interests or emotions), non-verbal communication (e.g., difficulties regarding eye contact, body language, gestures and facial expressions, and poor integration between verbal and non-verbal communication) and establishment of relationships (e.g., difficulties in adjusting behaviour to suit various social contexts, sharing imaginative play or making friends, and lack of interest in peers);

B “Restricted, repetitive patterns of behaviour, interests, or activities” (e.g., stereotyped or repetitive motor movements, use of objects, or speech; insistence in maintaining a status of sameness by keeping rigid routines; highly restricted interests, abnormal in intensity or focus; unusual responses to sensory input, such as an apparent indifference to pain/temperature, adverse response to specific sounds or textures, excessive smelling or touching of objects, visual fascination with lights or movement, etc.);

C “Symptoms must be present in the early developmental period” (although they may not be explicit enough until social demands exceed the individual’s capacities, or they may be masked by learned strategies later in life);

D “Symptoms cause clinically significant impairment in social, occupational or other important areas of current functioning”;

E “These disturbances are not better explained by intellectual disability (intellectual developmental disorder) or global developmental delay”. i.e., intellectual disability and Autism Spectrum Disorder often co-occur, but a comorbid diagnosis of Autism Spectrum Disorder and intellectual disability requires that social communication is below than expected for general developmental level.

DSM-V also states that individuals who have marked deficits in social communication, but whose symptoms do not otherwise meet criteria for autism spectrum disorder, should be evaluated for social (pragmatic) communication disorder.
2.2 Approaches for individuals with [ASD]

Often individuals diagnosed with [ASD] go through therapy in order to improve their social and communication skills. Most therapies are based in a psychology approach called Applied Behaviour Analysis (ABA), where the focus is to modify a certain behaviour in an individual. Below we describe such approach and we also briefly present two therapies where, unlike [ABA] the focus is the strengths and interests of the individual: Floortime and Son-Rise.

2.2.1 Applied Behaviour Analysis (ABA)

Most of the therapies used in the context of [ASD] are based on Applied Behaviour Analysis (ABA). This approach follows the operant conditioning principle, which consists of analysing an individual's behaviour considering not only the behaviour itself, but also the events that precede such behaviour (antecedents) and the events that follow the behaviour (consequences) [47,66].

The general procedure for an [ABA] based therapy consists of the following steps [47]:

1. Functional analysis in order to identify the individual's behaviours and the respective antecedents and consequences;
2. Selection of a single behaviour as the focus of the treatment (target behaviour);
3. Measurement of the current level of the individual's target behaviour; this level is then defined as the treatment's baseline;
4. Implementation of an intervention in order to improve the target behaviour and measurement of this behaviour during the intervention process in order to assess its effectiveness;
5. Assessment of whether the acquired skills were generalized across different settings, people and materials, or not.

Interventions implemented in step 4 might be antecedent based (when the stimuli and events that precede the target behaviour are manipulated in order to increase the likelihood of improving it), consequence based (when the consequent events or stimuli are manipulated instead) or a mix of both approaches [60]. Consequence based approaches can be classified as positive or negative whether they add or remove a certain stimulus in order to improve a certain behaviour. These approaches can also be grouped in punishment based approaches (if they consist of adding or removing an aversive stimulus) or reinforcement based (if the manipulated stimulus is something that pleases the individual, i.e., the individual's behaviour is rewarded). The latter is the most commonly used kind of intervention and includes the following techniques [60]:

- **Token Economy**: Delivery of a reinforcing item that the individual can later exchange for a more salient reinforcer (e.g., when the child successfully completes a task, they are given a token. As soon as the child has collected a certain number of tokens, these can be exchanged for the child's favourite snack);
- **Extinction**: Stopping the reinforcement of the individual's responses (e.g, quitting the habit of giving the child a reward after they completed a certain task);
- **Differential Reinforcement**: Attempting to extinguish an inappropriate behaviour through the conjunction of the following two components:
  - Reinforcement of a certain class of responses;
– Extinction or withholding of the reinforcement for a different response class (one that includes inappropriate responses).

A possible use case for this technique is when the child displays some sort of aggressive behaviour but there is another behaviour (e.g., asking for help) that is intended to be encouraged. The child will be rewarded whenever they ask for help, while aggressive behaviours will not be rewarded.

• **Shaping**: Attempting to establish a response or repertoire of responses by altering the target response as the individual gets closer to the desired final response (e.g., during the process of learning how to write, the child is first reinforced for picking up the pencil; when the child is able to pick up the pencil, they are then reinforced for scribbling; finally, when the child is able to scribble, they are reinforced for drawing a line) [47].

• **Chaining**: Systematic reinforcement of a sequence of discrete responses (behavioural chains). For instance, the process of putting on a pair of trousers can be split in multiple subtasks (e.g., holding trousers, opening the waistband, putting the left leg into the left trouser, etc.) and the child can be taught one subtask at a time and rewarded once the subtask is completed [47].

On the other hand, antecedent based approaches include:

• **Establishing Operations**: Antecedent events that can increase the value of the reinforcer (motivation operations) or decrease it (abolishing operations). For instance, a child can be restricted from having their preferred snack unless they initiate a conversation with a peer. This restriction is used in order to increase the value of the snack and encourage the child to initiate conversations more often;

• **Stimulus Control**: Outcome that emerges once a certain pair of stimulus and consequence often repeats itself (e.g., if, whenever a child asks a certain teacher for permission to go to the toilet, that teacher always allows them to do so, but another teacher never does, the child will quit asking permission to the second teacher);

• **Prompting**: Antecedent auxiliary stimulus that aims to elicit the desired response. Once such response becomes more frequent, this stimuli can be removed. Prompting can follow a least-to-most hierarchy (when the initial prompts are not enough to elicit the desired response, they are strengthened until the response is obtained) or a most-to-least hierarchy (when the prompts are gradually removed until no prompting is needed to elicit the target behaviour; this process is also known as fading). An example of prompting is when the child is told to pick a certain object and the adult points at that object in order to help the child performing such task. In a least-to-most approach, the adult would just tell the child to pick the object and, if they did not pick it after some time, the adult would then point at the object; in a most-to-least approach, the adult would start by always pointing at the object immediately after asking the child to pick it, and would gradually stop doing so in the following iterations;

• **Choice**: Providing the individual the chance to choose among different antecedents that may or may not lead to a reinforcing item (e.g., the child is given two switches and whether they press one switch or another, they are rewarded or not).

The main interventions used in therapy for individuals diagnosed with [ASD] in the context of an [ABA] therapy are often a combination of the techniques described above. There are several combinations used in diversified contexts [47][60], but the ones that relate the most with the development of language and communication skills are usually based in B. F. Skinner’s analysis of **Verbal Behaviour** [67]. Such
kind of interventions aims to develop functional language skills, namely the following core functional units identified by Skinner [47]:

- **Mand**: A form of response that is not controlled by an antecedent stimulus. An example of a mand is when the child takes the initiative to request for their favourite snack;

- **Tact**: A response that is determined (or, at least, strengthened) by an antecedent stimulus (e.g., when the child says “Apple” after looking at an apple);

- **Echoic**: A response which is controlled by someone else’s behaviour with point-to-point correspondence (e.g., when the child says “Apple” after hearing another person saying “Apple”);

- **Intraverbal**: A form of response similar to the echoic one, but without a point-to-point correspondence. An example of this form is when an individual performs an utterance (e.g., “Apple”) in response to a question (e.g., “What do you want?”).

Verbal Behaviour based interventions usually prioritize the development of mand behaviours, since they provide the child a greater control over the environment [47] and are maintained by specified, concrete, and effective reinforcers. Additionally, mand skills are likely to be learned rapidly because the child can immediately receive what they want in return [9].

### 2.2.2 Therapies: Floortime and Son-Rise

In opposition to ABA’s principles, there are therapies that are not concerned about altering the child’s behaviour nor about their symptoms, rather focusing on the child’s strengths and interests. This is the case of the **Greenspan Floortime Approach** (also known as Floortime) and of the **Son-Rise Program**.

Floortime, proposed by Stanley Greenspan, is a system based on the idea that the caregivers should join the child in their favourite activities, and take the chance to promote their creativity and spontaneity, as well as to include interactions that develop the child’s motor skills, senses and emotions, all of this while keeping them focused and engaged.

As for the the Son-Rise Program, it was created by Barry Neil Kaufman and Samahria Lyte Kaufman, the founders of the Autism Treatment Center and of The Option Institute. This therapy has several features in common with Floortime, as it also focuses on joining the child in the activities that engage them and taking advantage of the child’s motivation to teach them in an interactive way, but, unlike Floortime, it is mostly directed towards the parents.

### 2.3 Interventions

Below we present a couple of interventions that are commonly used with children with ASD either in ABA-based approaches or in other contexts.

#### 2.3.1 Picture Exchange Communication System (PECS)

Interventions using PECS focus on teaching children several communication skills, starting with the ability to perform requests (mand behaviours) [9]. This is achieved through a combination of several concepts related to ABA-based therapies, such as token economy, prompting and reinforcement. An intervention of this kind is divided in six phases [9]:

[1](http://www.stanleygreenspan.com/) (last visited on 29/11/2014)

[2](http://www.autismtreatmentcenter.org/) (last visited on 29/11/2014)

[3](http://www.option.org/) (last visited on 29/11/2014)
1. The trainer tries to identify an item that the child considers interesting, such as the child's favourite toy or snack. Once such item is identified, the trainer gives the child a card with a picture that represents that item, places it nearby and teaches the child how to physically exchange that card for the desired item;

2. Similar to Phase 1, but the trainer and the card start farther from the child in order for them to initiate the exchange process. The child no longer gets assistance on how to perform the exchange;

3. Cards representing non-preferred or unknown items are added next to the card representing the desired item. If the child picks the wrong card, the trainer says something like “No, we do not have that” and might prompt with a gesture towards the appropriate card. Over time, this phase's difficulty is increased by adding distracting cards that represent other items of the child’s interest;

4. The child is taught to create sentences using a cardboard strip with the words “I want” and a blank space after them, where they should place the card referring to the desired item;

5. The trainer teaches the child how to respond to “What do you want?” questions using “I want” sentences. In the first training iterations, the trainer provides prompting by pointing to the cardboard after posing the question. Over time, the delay between the question and the prompt increases, until no prompting is provided at all;

6. The child is taught to perform requests in different contexts, comment on situations and label objects using sentences starting with “I see” or “I have”. Additionally, the training is extended to minimally preferred items, other preferred items and new items.

Once these six phases are completed, the training can proceed with the inclusion of new terms in the child’s vocabulary and the increase of communicative functions (e.g., answering to other types of questions).

### 2.3.2 Treatment and Education of Autistic and Communication-handicapped Children (TEACCH)

TEACCH is an antecedent-based intervention which consists of acknowledging the needs, strengths and weaknesses of each individual with ASD by creating an individualized, family-centred learning plan. The main principles of this intervention include the physical separation of areas with different purposes, such as work and play, in order to minimize distractions, and the use of visual supports in order to make daily schedules and tasks predictable and understandable. This intervention has a broad range of applications including, but not restricted to, the development of language skills.

### 2.4 Summary

In this chapter, we provided some background on Autism Spectrum Disorder (ASD) and related concepts, as well as regarding the approaches followed with individuals with ASD so that the reader is familiar with such concepts as some of them are mentioned ahead in the document.

As of the most recent Diagnostic and Statistical Manual of Mental Disorders, DSM-V, an individual can be diagnosed with ASD if they exhibit persistent difficulties in social communication and interaction, as well as a restricted or repetitive patterns of behaviour and interests, if these symptoms are present in the early developmental period, causing clinically significant impairment in social, occupational or other

[\text{http://teacch.com/} \text{(last visited on 30/11/2014)}]
important areas of current functioning.

In order to overcome or minimize the difficulties felt by individuals with ASD, one commonly followed approach is Applied Behaviour Analysis (ABA). This approach focuses on targeting a certain behaviour that should be modified (either incited or avoided), and analysing it considering not only the behaviour itself, but also the events that precede such behaviour and the events that follow the behaviour. However, there are also approaches that focus on the individuals’ strengths and preferences, which is the case of the Floortime and the Son-rise therapies.

Typically in the context of ABA based therapies, there are several kinds of interventions that can be used to modify a certain behaviour. The most commonly used with children are the ones based on the use of reinforcement techniques such as rewarding the child with an item of their preference or providing help to encourage a certain response (prompting).
In this chapter, we present a review of existing work regarding two distinct topics that are related to our goals: the use of technology for children with Autism Spectrum Disorder (ASD) and the use of Natural Language Generation in the context of education.

Related work about the use of technology for children with ASD is presented in Section 3.1 and includes a characterization of the needs of the children and their caregivers, as well as a review of studies and software developed that aims to meet such needs.

As for the use of Natural Language Generation in academic/educational contexts, presented in Section 3.2, we review a set of works on that topic, focusing on the ones which deal with the generation of exercises, and we also present the tools and resources we used in our module of automatic generation of exercises.

### 3.1 Technology for children with ASD

As we have seen in Chapter 2, Autism Spectrum Disorder often comprise impairments in communication and development of verbal language. Some children are non-verbal or only acquire verbal skills later than their typically developing peers [5,35], thus needing an alternative mean of communication in order to be able to express themselves. Additionally, verbal communication could possibly be improved if children with ASD were given tools for the development of language skills that took their needs into consideration.

Having this in mind, several authors have searched for ways in which software solutions could be used to address the difficulties felt by individuals with ASD. Some authors have run surveys in order to assess the use of computer technology by individuals with ASD, as well as their needs and expectations towards the use of such technology as way to develop certain skills [20, 21, 39, 43, 57]. Others have designed research experiments where children with ASD would interact with software applications ranging from augmentative communication software [34] to educational tools [10, 14, 28, 29, 44, 46, 70].
among others. More recently, following the increasing popularity of mobile devices, many companies and researchers have developed mobile applications that take advantage of mobile devices’ characteristics such as portability and features such as text-to-speech, and made those applications available in markets such as Google Play Store\footnote{https://play.google.com/store} and AppStore\footnote{http://store.apple.com/pt} in the next sections, we present a review of some of those studies and software applications.

\subsection*{3.1.1 Characterization of user needs}

In order to develop software that better meets the needs of users with cognitive impairments such as ASD\footnote{Assistive technology is defined as “any item, piece of equipment, software or product system that is used to increase, maintain, or improve the functional capabilities of individuals with disabilities.” http://www.atia.org/i4a/pages/index.cfm?pageid=3859#What_is_AT} several authors performed surveys regarding aspects such as software features preferred by children with ASD\footnote{https://play.google.com/store} and their caregivers, as well as feedback from users’ relationship with assistive technology\footnote{http://store.apple.com/pt} namely: previous experiences, main difficulties that should be addressed, and reasons to abandon previously adopted technology. In this section, we will present a set of surveys in terms of their methodology, participants and main findings.

\textbf{Methodology and user background}

Most authors gathered information through anonymous online questionnaires\cite{20,39,57}, while some of them chose to perform semi-structured personal interviews as an alternative or a complement to questionnaires\cite{21,43}. These approaches included both closed and open answer questions.

The number of valid participations per survey ranged from 20\cite{21} to 144\cite{43}. Most of the surveys were responded by caregivers on behalf of children of all ages or by adults diagnosed with ASD\cite{39,43,57}, while others focused on the needs of the caregivers towards the task of teaching and communicating with children\cite{20,21}. One study also included a group of typically developing participants for comparison purposes\cite{39}. In terms of the severity of the diagnosis, some studies specified that participants included individuals diagnosed with some form of high-functioning autism or with Asperger’s Syndrome and individuals diagnosed with a more severe form of autism or Pervasive Developmental Disorder-Not Otherwise Specified\cite{39,57}. Finally, regarding literacy and verbal skills, some studies included both verbal and non-verbal participants, as well as some participants who were already able to read\cite{39}. Most of the participants were either attending school or some special education program\cite{39,43}.

Finally, the following countries were covered by the considered studies: United States\cite{21,39,57}, Portugal\cite{20}, United Kingdom\cite{43}, Spain\cite{43} and Bulgaria\cite{43}. This information can make the difference in the process of adopting assistive technology since each country might follow a different approach concerning investment in special education.

\textbf{Skills to develop and difficulties to address}

Some surveys posed questions that aimed to identify the skills that, according to caregivers, needed to be developed by individuals with ASD\cite{20,43,57}. For instance, the skills that the Portuguese tutors inquired by Lucas da Silva\cite{20} found more important for the children to develop were, by relevance order, communication (86%), social (77%), imagination (68%) and emotional skills (50%). Similar results were obtained by Putnam and Chong\cite{57}: social (32%) and communication (21%) skills were the most frequent answers, followed by academic skills, such as writing, math, reading, etc. (20%).
As for the study performed by Martos et al [43], it focused on comprehension of written text skills and presents with great detail the aspects that cause individuals with ASD to face difficulties in reading/comprehension tasks. The aspects that were pointed out more often by the caregivers as the cause of difficulties include the use of phraseological units (approximately 60%), long paragraphs (approximately 60%), irony/jokes (approximately 50%) and less common words, such as slang or technical jargon (approximately 40%).

Desirable software features

A common trace among the reviewed surveys is the quest to identify what features would be desirable in software aimed at individuals with an ASD. While some studies presented in advance a list of possible features to the respondent [39], others left to the respondent the job of naming such features in open answer questions [20,21,43,57].

For instance, Lehman [39] defined a set of features based on a feature list used in a previous survey by Malone [41], which only focused on typically developing children. The features proposed by Lehman to the respondents are shown in Table 3.1.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
<th>Similarity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clear goal</td>
<td>Multiple difficulty levels</td>
<td>Particular subject matter</td>
</tr>
<tr>
<td>Visible reward</td>
<td>Involves fantasy or role-playing</td>
<td>Child controlled activity</td>
</tr>
<tr>
<td>Sound effects</td>
<td>Use of animated character guides</td>
<td>Program-guided activity</td>
</tr>
<tr>
<td>Music</td>
<td>Online help</td>
<td>Fact-based activity</td>
</tr>
<tr>
<td>Randomness</td>
<td>No reading required</td>
<td>Variety of activities</td>
</tr>
<tr>
<td>Speed of response counts</td>
<td>Text in addition to auditory info</td>
<td>Ability to use alone</td>
</tr>
<tr>
<td>Visual effects</td>
<td>Rich visual detail</td>
<td>Ability to use with another person</td>
</tr>
<tr>
<td>Competition</td>
<td>Relevant theme</td>
<td></td>
</tr>
</tbody>
</table>

Table 3.1: Features considered in Lehman’s survey [39]

Lehman also included a subset of these features, but in the context of non-computer activities, in order to compare participants’ preferences in each context. The participants classified each feature in a scale of 1 (Dislikes) to 5 (Very Important). In the context of computer activities, the features that achieved higher average classifications were Visual effects and Speed of response counts (4.74), followed by Rich visual detail (4.72) and Ability to use alone (4.55). Furthermore, most computer features achieved considerably higher averages than their equivalents in non-computer activities (e.g., Speed of response counts (4.74, 3.19), Program-guided activities (3.40, 2.65) and Randomness (2.88, 2.27)); the one exception to this is Child controlled activities (4.04, 4.43). This suggests that children with ASD might see certain activities as more engaging when practised on a computer.

Putnam and Chong [57] presented open-ended questions that allowed the participants to come up with several suggestions, such as: taking into account sensory integration issues; making products portable and easier to use (e.g., using voice activation); incorporating fun or gaming elements.

A similar approach, followed by Dawe [21], retrieved three fundamental features: device portability, ease of use (and ability to evolve along with the user) and ease to upgrade or replace. As we will see

---

4A phraseological unit is a word group with a fixed lexical composition and grammatical structure (http://encyclopedia2.thefreedictionary.com/Phraseological+Unit – last visited on 06/01/2015)
later, these features are considered critical for a durable experience of technology usage.

In the survey performed by Martos et al [43], the participants’ suggestions focused on visual features that could improve reading experience (e.g., use of thesaurus, explanations or other disambiguation tools; short and clear messages; avoiding the use of figurative language).

Finally, tutors who responded to Lucas da Silva’s survey [20] were interested in the possibility of showing pictures (95%) as well as playing videos and reading books (both with 68%), and the vast majority of them would like to work with a tool that could be fully customizable (when asked so, 86% responded “Yes” and the remaining 14% responded “Maybe”).

**User’s previous experience with assistive technology**

Another topic that was frequent across these surveys was related to previous experience with technology: what types of technology were experimented, which features enhanced user experience and what were the reasons to abandon certain technologies.

In Dawe’s survey [21], this topic was strongly addressed. Starting with the technology experimented, the devices that were mentioned more often included augmentative communication devices, word prediction/spelling software, text processors, scheduling devices (e.g., picture schedules, timers and watches), screen reading software, educational games, accessible input devices (e.g., special keyboards and large-button calculators), cell phones and memo recorders. However, there was a variety of situations where these devices were soon dropped. Some of the reasons pointed out by participants were related to the following difficulties in the technology adoption process:

- Conflicting perspectives among involved individuals: as an example, a parent can be looking for the most adequate technology for his child, while the teacher is looking for the solution which is more familiar to the child and/or the cheapest technology, having in mind the school’s limited resource pool;

- Length of the adoption process: families might be depending on some sort of financial support in order to afford a certain device and, when they finally are able to, such device has become outdated or inadequate to the child’s needs.

Some of the experimented technologies were also dropped because of their initial complexity and/or great learning curve, incomplete/non-existent documentation, lack of means to back up and restore configurations and customizations made, lack of portability, etc. In short, assistive technology that could be surpassed in terms of efficiency and effectiveness by other (non-assistive) solutions was easily dropped.

In Putnam and Chong [57], only 25% of the respondents mentioned having experience with technology designed for people with cognitive disabilities. Most of the technology experimented was designed for personal computers and had an educational purpose, and only 18% was designed for people with ASD. Nevertheless, respondents reported a “moderately high level of satisfaction” towards their previous experiences.

As for respondents to Lucas da Silva’s survey [20], they had tried generic software such as PowerPoint [5] Word [6] or MagicKeyboard [7]. In 23% of the cases, tutors felt the need to drop the technology in

5 http://products.office.com/pt-PT/powerpoint (last visited on 22/12/2014)
6 http://products.office.com/pt-PT/word (last visited on 08/01/2015)
7 http://www.fundacao.telecom.pt/ (last visited on 20/12/2014)
use because they found it too complex, old-fashioned or not practical, and, in some cases, because they found other tools they wanted to try.

Discussion

Summing up the main findings provided by the surveys reviewed above:

- Communication and social skills are the most relevant skills caregivers want to see their children develop. Particular difficulties in language skills were also reported;

- As for features that would be desirable in technology for children with ASD, many suggestions were made, ranging from hardware characteristics (such as device portability) to software functionalities (such as content customization);

- The ease of use and maintenance is critical in order to avoid technology abandonment.

Some of these findings suggest that caregivers are mainly interested in technology that is easy to use and carry around (such as mobile devices) and they are also looking for technology that provides a customized and individualized user experience in order to take advantage of each child's characteristics.

3.1.2 Software for children with ASD

In order to meet the needs of individuals with ASD and considering their documented interest towards computers [39], many researchers and companies have developed software for different purposes, spanning from entertainment and planning to the development of the communication, academic, social or emotional skills.

In the recent years, perhaps as a consequence of the evolution of mobile devices, such as tablets and smartphones, and their rising popularity [26], the amount of software available has increased. In fact, searching Google Play Store for the keyword “autism” retrieves an extensive list of results. Additionally, there are websites devoted to autism which ease the task of finding appropriate software by providing more or less exhaustive lists of applications and, in some cases, the possibility to refine the search according to the preferred platform, purpose, or other search criteria.

Given the large number of applications and studies available, below we will narrow our focus on the software developed for communication and educational purposes, and we will also present a couple of in-house applications developed at Instituto de Engenharia de Sistemas e Computadores - Investigação e Desenvolvimento (INESC-ID).

Software for communication

In order to provide a way for individuals with ASD to express themselves, specially those who cannot communicate verbally, and given the recent dissemination of mobile devices such as smartphones and tablets [26], many mobile applications have been designed to serve as communication boards that can be used as an alternative and augmentative mean of communication. Most of these applications follow a similar structure: each concept is represented by a picture and the corresponding term as a caption; concepts are usually grouped by category (e.g., animals, food, etc.) or morphological clas
(noun, verb, etc.); once a picture is clicked or tapped, an audio description corresponding to the picture’s caption is played. Some of these applications also allow to select multiple pictures in order to build sentences and then play the audio corresponding to the whole sentence. This structure resembles the mechanism described in PECS (see Chapter 2), but, instead of using physical picture cards to trade for the respective items, the child can click or tap the pictures representing desired items in order to request for them or build more complex sentences. Using software communication boards comprises important advantages, namely:

- The possibility of hearing the words or sentences selected, which is not feasible with the physical cardboard (unless an adult standing nearby vocalizes the word or the sentence built);
- The ease to update the application with more sets of pictures, terms and audio (which can be added by the user or synthesized by text-to-speech).

One popular application that follows the structure above described is Proloquo2go, a paid application that allows to customize several parameters, such as the audio’s voice and the application layout, and it also provides the possibility of creating new buttons for terms, categories and sentences. An example screen of this application is shown in Figure 3.1.

![Proloquo2go's interface](image)

**Figure 3.1: Proloquo2go’s interface**

Proloquo2go has been used in studies regarding vocabulary acquisition as an auxiliary tool for participants with an ASD and a highly delayed expressive language. In this study, the participants had to identify a set of photographs by tapping over the corresponding icons in the interface of Proloquo2go. The results of this study suggested that the use of a mobile device supplied with a communication application such as Proloquo2go could be related to the increase of correct answers.

Proloquo2go is only available in English and for iOS; however, there are several similar alternatives for Android systems, some of which are available in other languages, such as Portuguese. Some examples of applications which are available for free in Google Play Store include MyTalkMobile, Niki Talk, TalkInPictures and Tools for Autism. A detailed comparison regarding availability,
language and customization possibilities of each application can be found in Table A.1 of Appendix A.1. This comparison leaves out several applications since most applications of this kind were either less functional than those mentioned above or more difficult to use due to bugs that negatively affected user experience. Other inconveniences in tested applications include the fact that some of them required the installation of additional software, or provided a very limited set of features in their free versions. Furthermore, most applications that allowed sentence building failed to correctly conjugate verbs according to the subject.

Software for the development of language skills

The acquisition of academic skills has also been a concern for many researchers and companies. Software developed for educational purposes covers a wide range of skills; however, given that impairments in communication are one of autism’s main diagnostic criteria, and considering the importance of verbal skills, not only in communication, but also in socialization and learning processes, we will focus on educational software that aims the development of linguistic skills.

As a matter of fact, many researchers and companies have been concerned about providing the means to facilitate the development of verbal skills. In terms of research, some authors performed studies where they compared the outcome of using software versus non-computational solutions or human intervention, regarding children’s language development process, while others focused their work on evaluating the impact of a specific set of features. In some cases, researchers used software specifically designed to perform their studies [29,46], while in others, they used existing software [10,28,44], including applications whose original purpose was not purely educational [14,34].

One of the earliest attempts to study the impact of software in teaching verbal skills belongs to Heimann et al [28]. This study intended to assess whether the use of a certain application could contribute to improve several skills, such as reading, phonological awareness, sentence imitation and verbal behaviour, and to increase the motivation of children with cognitive impairments. The software chosen, Alpha, allows multi-channel feedback (voice, animation and video were the channels used in this study) and comprises four working modes that can be manually set by the teacher according to the child’s progress: Individual Words, Creating Sentences, Testing Words and Testing Sentences. Word selection and sentence construction are reinforced by animations that illustrate them (e.g., if the user builds the sentence “The bear jumps over the horse”, an animation showing a bear jumping over a horse is displayed). Three groups of children participated in this study, being one of them composed of children diagnosed with an ASD. The results of the evaluation were analysed for each skill and indicated improvements for the group with ASD in Reading skills and in some categories of Verbal Behaviour skills (“verbal expression”, “enjoyment” and “seeks help”); however, some skills either did not improve or had to be tested along with another groups (for being difficult to administer to children with ASD).

A few years later (1997), Lehman developed an application aiming the development of linguistic skills named Simone Says [40]. The interaction with this application comprised the following stages:

1. Presentation of a graphical stimulus representing an everyday action or object (e.g., an apple), in order to maximize the likelihood of transferring the practised words to a home or school setting. It was possible to vary the basic attributes of the object depicted on the stimuli, such as its colour or size, as a way to improve children’s ability to generalize;

2. Recording of the child’s response towards the stimulus (e.g., if the stimulus was an apple, it was expected for the child to say “Apple”). If the child was not able to respond to the stimulus, a virtual
character named Simone would model an appropriate response as a prompting cue;

3. **Reinforcement** of the child’s response with a stimulus animation that reflects such response. For instance, if the stimulus consists of a frog and an apple and the child says “Eat the apple!”, the frog will eat the apple as a reward.

An example of the interaction loop is shown in Figure 3.2:

![Figure 3.2: Simone Says' interaction loop](image)

A relevant feature in this application is that it automatically adjusts the level of difficulty of each exercise according to the child’s progress, starting with exercises that aim the development of basic vocabulary and evolving to exercises that promote the practice of short verbal scripts, wh-questions, turn-taking interactions, etc. A example of a growing difficulty sequence of exercises can be seen in Table 3.2:

<table>
<thead>
<tr>
<th>Exercise order</th>
<th>Stimulus</th>
<th>Expected response</th>
<th>Corresponding reward</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>An apple</td>
<td>“Apple”</td>
<td>The apple animates (e.g. jumps)</td>
</tr>
<tr>
<td>2</td>
<td>Several apples</td>
<td>“Apples”</td>
<td>Every apple animates</td>
</tr>
<tr>
<td>3</td>
<td>A frog and an apple</td>
<td>“Eat the apple”</td>
<td>The frog eats the apple</td>
</tr>
</tbody>
</table>

Table 3.2: Example of a sequence of Simone Says’ exercises

It is important to note that there are other combinations of responses and resulting rewards that can be considered in this example. For instance:

- In Exercise #2, if the child’s utterance is “Apple”, only one of the apples will animate;
- In the absence of an appropriate response, the Exercise #3 can be complemented with a prompt that intends to map a mental state to the communication act, by showing a thought bubble near the frog with a preview of the animation that should succeed the response (e.g., a picture of the frog eating the apple).

There are no evaluation results available regarding this software, nor the software itself is available; however, we considered this work worth to be detailed since it reveals a concern towards the user’s progress, a feature that, as we will see, is still rare in currently available software.

---

Following the promising indicators provided by initial studies [28,39,46], other researchers continued to perform experiments on the use of software for educational purposes. In 2003, Massaro et al studied the impact of using a software application to teach new vocabulary to children with ASD and children with hearing impairments [10]. Specifically, the authors intended to find out whether the acquisition of new words was due to the software used or other sources, and also to find whether children were able to generalize the words learnt to other contexts. The software used, Language Player, comprised a tutoring character named Baldi and an exercise area that could contain pictures and/or text (see Figure 3.3). Baldi was a 3D animated character capable of producing both natural and synthesized speech and moving his mouth and tongue accordingly. In the context of the study, Language Player was used to perform various kinds of exercises, namely picture selection, word typing and vocalization (after reading/hearing the word) and word identification (without previously reading/hearing the word). Once each child participating in the study achieved 100% correct answers in training exercises, these were repeated with different images randomly placed in order to assess whether they were capable of generalizing. This process was repeated until the child was able to identify at least 5 out of 6 words across 4 unique sets of pictures. The authors reported that children participating in the study were able to identify more words following training exercises rather than before initiating the training, and that they were also able to identify words regarding new images in random locations, as well as to other interactions besides the ones with Language Player.

![Figure 3.3: Language Player's interface, with the character Baldi on the left](image)

In 2006, Massaro and Bosseler performed a second study using Language Player, but this time focusing on the influence of Baldi’s face movements on children’s learning process [44]. In order to make such assessment, the authors performed alternated exercises with and without Baldi’s face (but keeping the speech). Results reported indicate that children were able to acquire vocabulary both with or without Baldi’s face movements; however, their learning rate was faster when Baldi’s face was present.

Back in 2004, Hetzroni and Tannous performed a study using an application specifically designed for such purpose (I can word it too), in order to assess whether computer-based intervention could improve certain communication functions, by minimizing delayed and immediate echolalia, as well as irrelevant speech, and increasing the occurrence of relevant speech and communication initiations [29]. Exercises featured in the study could belong to one of 3 categories (food, hygiene or play). Reported results indicate that the 5 children participating in the study decreased the number of sentences with delayed echolalia for every category and increased the occurrence of relevant speech and communication initiations, mostly in food and play categories. Authors also reported that all participants displayed interest in interacting with the computer software.
More recently, in 2009, Whalen et al carried a study featuring a commercial application, **Teach Town: Basics**, in a classroom setting [70]. Participants were kindergarten/preschool students and were split into a treatment group and a control group, for comparison purposes. The intervention procedure included both on-computer lessons and off-computer activities that aimed to generalize skills learnt on computer lessons. The application allowed to automatically track each child's progress. Reported results indicated that the treatment group displayed improvements in several skills such as receptive and expressive language, auditory memory, acquisition of general concepts, matching ability and social skills.

While the studies above described software directed towards the development of linguistic skills, there were also studies where software originally intended for a different purpose was used instead. Two examples of such studies are those performed by Coleman-Martin et al in 2005 [14], in which **Microsoft Powerpoint** was used in order to present slides with words for the purpose of vocabulary acquisition and vocalization, and by Kagohara et al in 2012 [34], who used **Proloquo2go** as a mean for the participants to respond during picture identification exercises, as mentioned in Section [3.1.2]. In both cases, participants have shown improvements in the target skills. A summary of the procedures for the studies described so far can be found in Table A.2 of Appendix A.2.

Similarly to communication software, in the recent years, many mobile applications focusing the development of language skills have been designed and made available through application markets. Given the large amount of such applications, we will only detail a subset of them, although none of them is available in Portuguese:

- **Autism iHelp**[^21]: Comprises a training module where the child can learn the term corresponding to each picture, and a gaming module where the child has to identify which of the presented terms matches the given picture or which of the displayed pictures corresponds to a given word. Allows a few configurations, such as toggling the audio descriptions, and keeps track of user’s progress, although none of these features is associated with an user account;

- **Autism Learning – Camp Discovery**[^22]: Includes many different games, although most of them constitute paid content. An example of a free game is **Matching identicals**, in which the child has to match the picture of an object with another picture of the same kind of object. The first iterations of this game provide prompting by displaying the correct picture in a larger size. During each round, the difficulty is increased as the child progresses, by adding more distracting pictures and displaying all the pictures in the same size (or decreased if the child started giving wrong answers). An example of interaction is show in Figure 3.4. This application includes some administration options, as well as the possibility of choosing the animation that will be shown for correct answers;

- **Autism & PDD (Lite)**[^23]: Comprises a series of applications devoted to diversified language topics such as grammar, categories, yes/no questions, etc. Exercises consist of illustrated multiple choice questions. Correct answers to those questions elicit a reinforcement animation. The only customization option available is the possibility to randomize the order in which the questions appear;

- **Lite Autism**[^24]: Consists of a collection of applications based on multiple choice exercises regarding diversified topics (e.g., fruits, objects, facial expressions). Each exercise is preceded by a

[^22]: http://campdiscoveryforautism.com/ (last visited on 22/12/2014)
[^23]: http://www.linguisystems.com/products/product/search (last visited on 09/01/2015)
Figure 3.4: Example of an interaction in Camp Discovery

training phase in which the child can learn the picture corresponding to the audio description. In the exercise phase, correct answers are reinforced by a brief animation. This application includes some administration options, such as the possibility to choose the reinforcement animation, and logs user progress;

- **Vizzle Player**: This application includes both educational exercises and utility modules, such as timers and schedulers. The educational exercises work in a similar fashion to the above described applications, but, in case a wrong answer is selected by the user, that answer is either deleted (see Figure 3.5b) or marked with a red cross. It is possible to associate game progress to a user account.

Figure 3.5: Example of an interaction in Vizzle Player

[last visited on 22/12/2014](http://govizzle.com/)
A more detailed comparison of highlighted educational applications reviewed so far can be found in Table A.3 of Appendix A.2.

In-house software

In the recent years, the development of software that could make the difference for individuals with special needs has been a concern for Portuguese researchers. Out of the research projects and theses that meet this concern, we highlight two applications developed at Instituto de Engenharia de Sistemas e Computadores - Investigação e Desenvolvimento (INESC-ID) in the context of two masters’ theses, since both target the same audience: children with ASD.

One of such applications is Troc@s\(^\text{26}\), a web platform developed in 2011, in the context of a special education’s master’s thesis [65]. This platform was intended to be easily configurable and adjustable, considering the children’s ASD level of severity, as well as their cognitive, social and linguistic development levels, their education level and age. Troc@s comprises the following modules:

- **Images**: a picture gallery (Figure 3.6a);
- **Songs**: a music player (Figure 3.6b);
- **Television**: a set of short animated cartoon videos (Figure 3.6c);
- **Message board**: allowing the children to exchange messages with each other, using both words and/or symbols (Figure 3.6d);
- **Story book**: a set of books belonging to the National Reading Plan, that the children can see, read or listen to (Figure 3.6e);
- A connection to Photo Story\(^\text{27}\) an open source application that can be used to retell stories or live events using pictures, audio or words (Figure 3.6f).

![Figure 3.6: Troc@s' categories screens](image)

The contents featured in the first three categories (Images, Songs and Television) can be classified by the children through the use of Like and Dislike buttons. As for the platform management, tutors can manage and customize the presented contents, browsing through a folder structure similar to the one in

---

\(^{26}\) [http://www.mytrocas.info](http://www.mytrocas.info) (last access in 11/10/2015)

Windows operating system.

This platform was evaluated with three twelve year old participants in the structured teaching unit they were attending. The procedure consisted of introducing sessions to interact with Troc@s in each child's daily schedule and keeping both written and photographic record of those sessions, taking in account the feedback provided by the children, behaviour changes, etc. The observations made showed several improvements regarding communication and establishment of relationships with peers, understanding and commenting information, retelling a story or life event, etc. Despite these results, since the initial version still presented some issues regarding performance and usability, those issues were addressed in the context of another master's thesis [20], in 2013. The resulting improved version, myTroc@s.net, also integrated two new features: the possibility of creating an unlimited number of individual or shared profiles regarding the contents and layout of the application so that the application was tailored to the needs of each child, and the possibility of logging all the interactions between each child and the application (external apps launched, messages exchanged, activities played, user attention during an interaction, etc.).

The other application to be highlighted is an iPad application that aims to allow children with [ASD] to learn linguistic skills in a customizable environment [61]. While the goal of this application is very similar to ours, since it was being developed by the time we would need to start implementing our proposal, we chose not to extend this one.

This application allows to add words and upload images in order to create new exercises, that can be of the following types:

- **Leitura** (Reading): A word appears on the screen for the child to read it, along with a sad and a happy smiley faces below the word. If the child correctly reads it, the therapist should press the happy smiley (Figure 3.7a[1]). Otherwise, they should press the sad smiley (Figure 3.7a[2]) and a audio recording of the word is played. There are two reading modes: one where the word should be read as a whole (see Figure 3.7a) and another where the word should be read syllable by syllable (see Figure 3.7b);

- **Encaixe** (Writing): The letters that compose a word are shuffled and the child has to drag each letter to the correct place in order to assemble the word (see Figure 3.7c). There is also a sound button that can be pressed any time in order to hear a recording of the target word, as well as a visual prompt to let the child know when they are dragging a letter to a wrong place;

- **Ligação** (Matching): A set of N images and N words appear on the screen and the child has to match each image with the correct word ((see Figure 3.7d)). The number of matches to be made per exercise increments by one as the child achieves three correct answers, to a maximum of five matches per exercise.

In every case, when the child provides a correct answer, both visual and sound reinforcements are used. This application also allows to customize several settings:

- Turning the reinforcements on and off, as well as selecting the visual and the sound reinforcement to use (see Figures 3.8a and 3.8b);

- Defining exercise-specific settings (e.g, the number of matches to start with in the Matching exercise, as shown in Figure 3.8c).
• Defining Interface customizations, among others (e.g., the font and capitalization of the words, as shown in Figure 3.8d).
to teach a new word using the Writing exercise. In the end of the three phases that composed this evaluation, the child was able to correctly write the word in four out of the five trials, thus allowing to conclude that the child was able to learn the word using this application.

Discussion

Having in mind the conclusions reached in Section 3.1.1 concerning user needs and expectations, it is safe to say that the market already offers systems that meet important requirements such as the possibility to customize the content and some appearance aspects of the application. However, it is still possible to identify some serious pitfalls:

- Aside from Rosa’s application [61], none of the applications that focus on the development of linguistic skills is available in Portuguese;
- Most applications utilized in previous studies are either discontinued, paid, or lack an educational purpose;
- Despite the fact that the reviewed mobile applications are available for free, most of them (marked with a * in Table A.3 from Appendix A.2) include paid features or constitute “lite” versions of paid applications;
- Some of the applications have several navigation bugs that negatively affect user experience;
- Only few applications allow the upload of custom multimedia content; the customization and administration options offered by most applications are quite limited;
- Very few applications are concerned about evolving according to the user’s progress (only Camp Discovery and Simone Says do it) and one of them is discontinued (Simone Says).

All these observations constitute challenges that, as we will see later, were taken in consideration in our work, specially the ones regarding customization.

3.2 Automatic generation and processing of content

Considering the goals we defined in Chapter 1, we reviewed existing works in automatic generation of natural language, specially those with an academic/educational focus, and we also analysed resources and tools that could support our generation tasks.

Automatic features such as generation and processing of content can reveal themselves useful in several contexts, from education (e.g., languages [11, 17, 36, 50, 58, 72], maths [45, 63, 68], programming [38, 59]) to entertainment [19]. Focusing on education, these can help teachers and educators by allowing them to save them time while creating content (e.g., when a great amount/variety of exercises are needed) [63], adapting content to each student’s characteristics (e.g., when content can be generated considering the student’s progress or other relevant information) [45], contributing to a fairer grading process (e.g, instead of grading on a “correct/incorrect” basis, automatic processing of content can allow partial grading) [37, 38, 62], make cheating more difficult (e.g, using several variations of the same kind of exercise, one per student) [59], etc.

Below we present a review of the works that reach closer to our goal (which is to automatically generate multiple choice exercises in Portuguese for children with special needs, namely ASD to develop language skills).
3.2.1 Automatic generation of multiple choice exercises

Several works address the challenge of generating exercises of several types, namely multiple choice exercises. A multiple choice exercise is a kind of exercise composed by one question and a set of possible answers, in which only one is correct. The remaining answers are usually referred to as *distractors*.

We can think of the generation of this kind of exercises in terms of different subtasks, namely: question generation, correct answer generation, and distractors generation. For the task of generating questions, some of the existing works follow an approach based on creating questions out of annotated corpora [7, 11]. Others use existing questions as seeds in order to learn patterns and generate new questions [18], and there are also works in which both approaches were tried (e.g., a rule based approach where the questions were generated from newspaper articles, and a pattern generation approach where patterns to generate the questions were extracted from existing pairs of questions and answers [19]).

As for the distractor generation, the approach followed in most works consists of selecting words that replicate features of the correct answer (e.g., words that have the same morphological class as the correct answer). In Curto [19], the generated distractors can be more or less similar to the correct answer given a threshold of the number of features shared with the correct answer. In Albade [7], context information regarding the question is taken into account to select the final distractors from the candidates.

3.2.2 Resources and tools

For the purpose of automatically generating content, we analysed a set of resources and tools that we could use in order to extract words and images, as well as to assure the correctness of the resulting content. Below we present our review of the considered resources.

**Word and image resources**

For the task of retrieving words in Portuguese and images to illustrate such words, we were looking for lexical and/or image resources. For that purpose, one well known lexical resource is Princeton’s *WordNet* [56]. In WordNet, words are grouped into sets of cognitive synonyms (also known as synsets). These synsets are related among themselves through conceptual-semantic and lexical relations such as hyperonymy/hyponymy, meronomy, etc. WordNet only refers to English lexicon, but there are other online resources that were built using a similar structure for Portuguese lexicon:

- **WordNet.PT** [33]: Offers an online search engine that allows the user to lookup for a word, along with its morphological category (e.g., noun, verb), as well as browse over the word’s relations with other words (e.g, hyperonymy). WordNet.PT Global [34] the Instituto Camões’s version, also allows to filter the results according to the Portuguese variant (European, Brazilian, etc.). None of these versions provides a downloadable resource;

- **MultiWordNet of Portuguese** [35]: Offers an online search engine that allows two ways of searching for terms: using a visual dictionary that graphically presents the queried word’s semantic connec-

---

[31] A word is hyperonym of another word if the first word is a superset of the second word (e.g., “rose” is an hyperonym of “gladiolus”). The second word is then an hyponym of the first. [54]
[32] Meronymy is a part-whole relationship among words. For instance, “rose” is a meronym of “petal” [64].
[34] http://cvc.instituto-camoes.pt/traduzir/wordnet.html (last visited on 05/01/2015)
tions to other words, or using the Italian MultiWordNet’s interface. In both cases, no downloadable resources are available;

- **OpenWordNet-PT**: A WordNet-like downloadable lexical database for Brazilian Portuguese;
- **Onto.PT**: A lexical database that was automatically built upon existing WordNets, thesauri and online dictionaries, thus providing a larger amount of information. It is available as both an online search interface and a downloadable resource in different formats.

None of resources described above facilitate the retrieval of images; the only one who does is **ImageNet**. This resource is organized in a similar fashion to Princeton’s WordNet, allowing to browse through nested hierarchies of synsets, in which the outer synsets contain the hyperonym of the inner synsets, as shown in the listing below.

```xml
<ImageNetStructure>
  <synset widn="n00021265" words="food, nutrient" gloss="any substance that can be metabolized by an animal to give energy and build tissue">
    <synset widn="n07566340" words="foodstuff, food product" gloss="a substance that can be used or prepared for use as food">
      <synset widn="n07849336" words="yogurt, yoghurt, yoghourt" gloss="a custard-like food made from curdled milk"/>
    </synset>
  </synset>
</ImageNetStructure>
```

Each synset is identified with a id, which is associated with a text file including a set of image URLs. These URL files can be accessed without restrictions, but the respective images can only be downloaded for free after a request for their use under certain terms for educational and/or non-commercial purposes.

An alternative approach for retrieving images would be the use of an image search Application Programming Interface (API) in order to search for images that correspond to certain terms or using an image database where images are annotated with the corresponding terms. However, the main options (Google Custom Search, Bing, and Yahoo BOSS Search) are either paid or offer a very limited number of transactions for free. In terms of free APIs, there is Flickr which is restricted to images uploaded to Flickr, and Faroo which presents no specific restrictions. Still, it should be noted that images retrieved by search APIs might be under copyright.

**Part of Speech tagging: TreeTagger**

TreeTagger is a tool for the Part of Speech (PoS) and lemma annotation of sequences of words. It can be used for any language as long as a lexicon and a manually tagged training corpus are given, which is the case of many languages, including Portuguese.

---

37. [https://github.com/arademaker/openWordnet-PT](https://github.com/arademaker/openWordnet-PT) (last visited on 05/01/2015)
39. As of 05/01/2015, the online search functionality is down.
40. [http://image-net.org/](http://image-net.org/) (last visited on 05/01/2015)
41. [https://developers.google.com/custom-search/json-api/v1/overview](https://developers.google.com/custom-search/json-api/v1/overview) (last visited on 05/01/2015)
43. [https://developer.yahoo.com/boss/search/](https://developer.yahoo.com/boss/search/) (last visited on 05/01/2015)
44. [http://www.flickr.com/services/api/](http://www.flickr.com/services/api/) (last visited on 05/01/2015)
45. [http://www.faroo.com/hp/api/api.html](http://www.faroo.com/hp/api/api.html) (last visited on 05/01/2015)
46. [http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger](http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger)
For each analysed word, TreeTagger returns a multi-letter tag in which the first letter indicates the morphological class of the word (e.g., noun, verb, determinant...). The meaning of the remaining letters depends on that class; for instance, if the word is a noun, the letter in the second position indicates the type (common, proper, etc.), the letters in the third and fourth positions indicate the gender and the number, respectively, and so on.\textsuperscript{47}

### 3.3 Summary

In this chapter, we went through several works that somehow relate to our goals, mainly in two distinct fronts: the existing technology for children with ASD and the state of the art regarding automatic generation of content for purposes of generating academic/educational content.

Regarding technology for children with ASD, several researchers have run surveys in order to assess the needs and expectations of children with ASD and their caregivers. The outcome of these surveys indicates that communication and social skills are the ones caregivers prioritize. Also, both children and caregivers display interest towards the use of technology; however, caregivers have reported many difficulties in their past experiences with technology, mainly regarding aspects such as device portability, content tailoring based on the child’s characteristics, ease of use and maintenance, among others.

Both researchers and companies have put efforts in developing software solutions targeting children with ASD. Although there is a wide offer of applications (specially mobile ones), there is a lack of applications that are simultaneously bug-free, highly customizable, and available for Portuguese at no monetary cost.

Concerning systems for automatic generation and processing of content, there are many advantages of using them in educational contexts, such saving time by generating many variations of the same exercise template, accounting for students specific information, etc. In order to develop such a system, word banks like ImageNet and tagging tools like TreeTagger can be useful.

\textsuperscript{47}The full list of tags is available in \url{http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/data/Portuguese-Tagset.html} (last access in 11/10/2015)
Recalling the goals specified in Chapter 1, our aim is to deliver a platform in which children can solve exercises that allow them to **develop language and generalization skills** and also where caregivers can create exercises and perform a set of **customizations** to meet the needs/characteristics of each child.

From the child’s point of view, our solution comprises the following features:

- Solving multiple choice exercises;
- Using prompting cues to help the child to choose the correct answer for a given exercise;
- Possibility of showing content tailored to the child’s preferences as a way to reinforce the production of correct answers;
- Use of an animated character to present the exercises and provide feedback according to the child’s answers;

As for the caregiver’s point of view, the delivered features include:

- Possibility of creating and managing different kinds of exercises, with full control over their content;
- Possibility of uploading and managing multimedia resources (images) to be featured in the exercises;
- Possibility to create and manage the application’s users;
- Possibility of customizing:
  - The content to use as reinforcement for each child;
  - The animated character’s utterances in each situation (e.g., when the child logs into the application, when a correct answer is given, when a wrong answer is given, etc.).
In order to implement our solution, we have considered the possibility of extending an existing in-house platform. Doing so would allow us to save time on features that might have already been implemented and to direct our effort towards new challenges. Out of the applications considered, Virtual Therapist for Aphasia Treatment (VITHEA) stood out as the most appropriate for our needs. It comprises both desktop (web) and mobile (Android) applications, and includes useful features such as multimedia files management and a talking animated character. Besides, VITHEA has shown to be versatile enough to be extended for different purposes. On the other hand, VITHEA’s client-server architecture implies a dependency on Internet connection, which limits the range of situations where the exercises can be prepared and solved. Nevertheless, this drawback is not that restrictive if we recall that this architecture allows therapists to remotely keep track of their patients’ progress, and manage exercises, users and resources.

Although extending VITHEA allowed us to save some time regarding infrastructure aspects, we still had to go through a considerable implementation effort in order to adapt it to our specific context. In the following sections, we introduce this platform, as well as the main modifications we had to perform in order to include the features introduced above.

### 4.1 VITHEA - Virtual Therapist for Aphasia Treatment

Virtual Therapist for Aphasia Treatment (VITHEA) is an award-winner platform developed at Spoken Language Systems Laboratory (L2F), originally targeted to people diagnosed with aphasia (a language disorder that causes difficulties in several tasks, such as word naming). This platform allows patients with aphasia to solve exercises in which they can practice oral naming skills. It also allows therapists to create new exercises for the patients to solve and to monitor each patient’s statistical data. In 2014, VITHEA integrated a module intended for automatic screening of cognitive impairments such as Alzheimer, using neuropsychological tests; however, we will not provide further details about it in this document, as the version that was provided for us to extend does not include this module.

Below we describe the type of exercises available in VITHEA as well as the platform’s modules (one for the patients and another for the therapists), its architecture and the underlying technology.

#### 4.1.1 Word naming exercises

VITHEA’s exercises aim for aphasia patients to recover their ability regarding word naming. Each exercise can be assigned with a different category that states the skill it targets (e.g., Naming object picture, Complete sayings, Part-whole associations). All the types of exercise follow the same structure, being composed of: (a) a stimulus in the form of text, image, video or audio; (b) a question about that stimulus; (c) a set of up to four correct answers. An example of an exercise is shown in Table 4.1. Each exercise can also be tagged as a training exercise (that can be solved by the patient as a way to practice) or as an evaluation exercise (that should be used by the therapists to assess the patient’s progress).

When solving an exercise, the patient is provided with the stimulus and the question about it, to which he/she should orally reply. For an answer to be considered correct, it does not have to be an exact match to one of the correct alternatives provided by the therapist. A list with the most frequent synonyms and diminutives is also considered, so, if the patient’s reply includes a word in this list, it is considered correct. Replies that include a correct keyword, even if they also include unrelated terms, will also be...
What is the name of this object?

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Question</td>
<td>What is the name of this object?</td>
</tr>
<tr>
<td>Stimulus</td>
<td><img src="image1.png" alt="Image" /></td>
</tr>
<tr>
<td>Correct answers</td>
<td>Bill, Cash, Money</td>
</tr>
</tbody>
</table>

Table 4.1: Example of an exercise in VITHEA

considered correct. Finally, if the reply is incorrect, the patient can try again for a user-defined number of attempts.

4.1.2 Therapist’s module

This module is intended to be used by therapists in order to create the exercises described in Section 4.1.1 being composed of the following sub-modules:

- **Exercises:** Allows to create, edit, preview and delete exercises. It also allows to manage the multimedia resources (images, audio or video) available to be used as stimuli in the exercises. The interface for exercise creation is displayed in Figure 4.1.

- **Users:** Allows to create, edit, and delete users (both therapists and patients). In the case of the patient, it stores not only basic user information, but also aphasia-specific information (e.g., the patient’s aphasia type and level) that is updated considering the results obtained by the patient in the evaluation exercises;

- **Statistics:** Allows to monitor statistical data about the patient (e.g., the user’s accesses to the system and start/end time of each training exercise).

![Figure 4.1: VITHEA's therapist's module](image2.png)
4.1.3 Patient’s module

This module is available as both a web application and a native Android application. It provides the exercise interface to the patients and stores statistical data about each session (e.g., the patient’s utterances when replying to a question). The main screen of the application lists all the exercise categories. When the user chooses a category (e.g., Naming object picture), one of the exercises in that category is displayed (the sequence of exercises is presented in a random order). The user can choose to answer the exercise, move to the next exercise or go back to the main screen. This module also features an animated talking character, called “Catarina”, which is responsible to present the exercises and provide feedback regarding the patient’s answers. Figure 4.2 shows the main screens 4.2a and 4.2c and an example of an exercise 4.2b and 4.2d for the web and the mobile applications, respectively.

![Figure 4.2: VITHEA’s patient’s module [53.]](image)

(a) Main screen (web)  
(b) Exercise (web)  
(c) Main screen (mobile application)  
(d) Exercise (mobile)

4.1.4 Architecture and underlying technology

The process behind the resolution of an exercise comprises the following steps: for each stimulus, the animated character asks the patient a question related to a given stimulus. The patient’s reply is recorded into an audio file that is sent to an Automatic Speech Recognition (ASR) module, which converts the audio into a textual representation to be compared with a set of possible correct answers. Finally, the patient receives both textual and audio feedback based on the comparison result. The auditory feedback is presented by the virtual character through text-to-speech synthesis). This process is depicted in Figure 4.3.

As stated before, VITHEA is a distributed application with a client-server architecture. The client can be either the user’s browser (when accessing either the therapist’s or the patient’s application) or the patient’s mobile application. The web applications are deployed with Apache Tomcat in the server, [http://tomcat.apache.org/](http://tomcat.apache.org/) (accessed on 06/01/2015)
where the database can also be found. The patient's web application is responsible for communicating with the ASR module and invoking the animated character (running in an Apache server).

Both the therapist's and the patient's web applications follow a Model View Controller (MVC) architecture composed of different layers, in order to separate presentation from data (see Figure 4.4). These layers are grouped in three main tiers:

- **Presentation tier:** In this tier, application data is collected, validated and displayed through a graphical user interface, composed of Java Server Pages (built using Apache Tiles). Here, the Struts framework is used to render HTML elements containing information coming from (or to be sent to) the server. The pages' appearance is defined with Cascade Style Sheets (CSS) and some client-sided operations (such as input validation) are coded in JavaScript;

- **Middleware tier:** This tier is composed of several layers, and includes the application logic (coded in Java), the bean objects (which carry the information to be displayed in/collected from the presentation tier) and the data access objects (which connect to the database entities). The mapping between Struts elements and the Java bean objects is made using the Spring framework, and the object-relational mapping is made using the Hibernate framework;

- **Data tier:** Comprises the persistent data, which is stored in a MySQL database.

Additionally, the patient's web application also uses Adobe Flash Player to record the patient's utterances. The ASR module uses AUDIMUS, a Portuguese speech recognition software developed at L2F. As for the animated character, it was developed in Unity3D and its voice is synthesized using DIXI, a Portuguese text-to-speech engine, also developed at L2F.

The patient's web application does not work on a mobile browser: the class responsible for acquiring speech input is not supported in Flash Player running on a mobile browser, and the animated character requires the installation of Unity Web Player which only runs in Windows and Mac, thus leading

---

35
to the development of a native Android application. This application uses RESTful web services, i.e., services that follow the Representational State Transfer (REST) principles; this means that the client-server communication is stateless – all the information needed to understand a request must be part of that request and the session state is kept solely on the client side. The data transferred between the mobile application and the web server is represented in JavaScript Object Notation (JSON). The server side services were implemented using Spring Security\(^{12}\) and Spring Web MVC\(^{13}\). On the client side, the communication with the web services is made through Spring for Android\(^{14}\). The virtual character was exported from Unity3D to Android, and the speech recognition remained performed by AUDIMUS, being the audio recording provided by the AudioRecord class of the Android API\(^{15}\). An overview of the communication between the mobile application and the web server is displayed in Figure 4.5.

---

\(^{12}\) [http://projects.spring.io/spring-security/] (accessed on 06/01/2015)
\(^{13}\) [http://projects.spring.io/spring-webflow/] (accessed on 06/01/2015)
\(^{14}\) [http://projects.spring.io/spring-android/] (accessed on 06/01/2015)
\(^{15}\) [http://developer.android.com/reference/android/media/AudioRecord.html] (accessed on 08/01/2015)
4.2 From VITHEA to VITHEA-Kids

Although we took advantage of VITHEA’s infrastructure, as well as of some of its features (such as the animated talking character (Catarina) and the multimedia file management), our target users differ from those of VITHEA, which means VITHEA-Kids also differs significantly from VITHEA concerning its purpose and functionalities. Below we present the types of exercises we implemented, our solution’s conceptual model and the main changes performed to VITHEA to reflect the desired functionalities.

4.2.1 Multiple choice exercises

In this work, instead of using VITHEA’s exercises, we present a new kind: multiple choice exercises. Each exercise is composed of a question, an optional stimulus (picture or text) complementing the question, and a set of possible answers (textual or pictures, respectively), in which only one is correct. The number of distractors per exercise can vary between zero and three, easing the task of creating exercises with variations in the number of distractors. For instance, a caregiver can create exercises with zero distractors in order for the child to strengthen a certain concept, and then, when appropriate, create exercises with one or more distractors to increase the difficulty.

We have implemented multiple choice exercises of two subtypes: 1) identification of an image; 2) choice of the correct picture to match a given word/expression. An example of each subtype of exercise is presented in Tables 4.2a and 4.2b.

![Table 4.2: Examples of multiple choice exercises](image)

(a) Image stimulus and a set of possible textual answers

(b) Textual stimulus and a set of possible answers (images)

The examples provided do not represent exhaustively all the possibilities that these exercises offer. Other examples include exercises where the goal is to find the “outsider” (i.e., the only answer that does not match the stimulus or the topic), or exercises with an empty stimulus, in which the target word is mentioned in the instruction. For instance, it would be possible to create an exercise in which the question would be “Which girl is happy?”, the word stimulus would be empty and the set of answers would be pictures of a girl expression different emotions (being one of them “happy”).

The reason behind the choice of using multiple choice exercises is that this kind of exercises has been used for children with ASD (recall Chapter 3) and might allow to work on skills such as vocabulary acquisition, word-picture association, and generalization. In particular, the structure of the proposed exercises allows the development of three out of the core functional language units of verbal behaviour mentioned earlier in Chapter 2: tact behaviours can be elicited by displaying a stimulus (an image or word) corresponding to the correct answer; echoic behaviours can be elicited when the animated character utters the correct answer; finally, Intraverbal behaviours can be elicited when the question is uttered.
by the animated character.

### 4.2.2 VITHEA-Kids’ conceptual model

Similarly to VITHEA, VITHEA-Kids is composed of two modules: the caregiver’s module and the child’s module. Both modules share a set of concepts, each characterized by certain attributes and capable of a specific set of actions. Starting with the concepts (and their attributes), we have:

- Two types of users: **child** and **caregiver**. Both require a unique **username** and a **password**. Child users are also characterized by generic information about them (*first and last name, birthday, sex*);
- **Resources** in the form of multimedia files (e.g., images), which are characterized by the file’s **name**, **path**, and **size**. There are two types of resources:
  - The ones to be featured in the exercises (either as the question stimulus or as an answer/distractor). These are also characterized by a **topic** and a **title**;
  - The ones to be used as **reinforcement** when the child picks a correct answer.
- **Multiple choice exercises**, characterized by a **topic**, a **level of difficulty**, an **instruction** and the **number of distractors**. There are two types of exercises:
  - Exercises where the stimulus is a multimedia resource and the set of possible answers are textual;
  - Exercises where the stimulus is a word and the set of possible answers are multimedia resources;
- **Logs** regarding the performance of a child in a session of exercises, comprising the following information:
  - The date and time when it was taken;
  - Total number of exercises the child went through;
  - Number of exercises that were correctly answered;
  - Number of exercises that were skipped;
  - Average number of times a distractor was hit per exercise.
  - Which exercises were gone through and the child performance on each of them.
- A set of **messages** to be uttered by the animated character in several situations, such as:
  - When the child logs into the application;
  - When the child picks the correct answer in an exercise;
  - When the child completes a session of exercises.

As for the relations among these concepts and actions that can be performed:

- Caregiver users are only able to log into the caregiver’s module and child’s users can only log into the child’s module;
- Caregiver users are able to:
  - Create and manage child users;
– Create and manage exercises;
– Upload and manage the resources to be used in the exercises;
– Upload and manage reinforcement images;
– Customize the messages to be uttered by the animated character.

- Child users are able to solve exercises and visualize reinforcement images when a correct answer is picked.

- Each child user is associated with:
  – A set of messages to be uttered by the animated character;
  – A set of reinforcement images;
  – The logs regarding their interactions with the child module.

### 4.2.3 Caregiver’s module

The caregiver’s module\(^{16}\) was based on VITHEA’s therapist’s module, however, it offers different functionality. Therefore, we had to modify this module in all its layers (from persistence to presentation), since the information to be recorded and displayed regarding the exercises, multimedia resources and users strongly differs from the information recorded in VITHEA.

After logging into this module, the caregiver is presented with the main screen, which features four sections (see Figure 4.6): Exercícios (Exercises), where the caregiver can manage both the exercises and multimedia resources; Utilizadores (Users), where the caregiver can manage child users’ accounts; Preferências (Preferences), where the caregiver can customize several aspects of the interaction between each child user and the child module; Estatísticas (Statistics), where the caregiver should be able to access the logs of each child user’s performance during a session of exercises. The latter is still being developed.

![Figure 4.6: Caregiver’s module main screen.](https://vithea.l2f.inesc-id.pt/vitheakids-admin/index.action)

The functionalities mentioned above and the main changes implemented in order to achieve them are described in the next subsections.

**Exercises and multimedia resources**

Regarding the exercises, the caregiver is now able to create the two types of multiple choice exercise presented in Section 4.2.1. When creating an exercise, the caregiver should specify the topic of

\(^{16}\)Available in [https://vithea.l2f.inesc-id.pt/vitheakids-admin/index.action](https://vithea.l2f.inesc-id.pt/vitheakids-admin/index.action)
the exercise (e.g., “Animals”) from a predefined set, its level of difficulty (Introductory, Intermediate or Advanced), the instruction, the stimulus (see Figure 4.7c), the correct answer and the set of distractors, as depicted in Figure 4.7a (multiple choice with image stimulus and textual answers) and Figure 4.7b (multiple choice with textual stimulus and image answers). When the caregiver submits the exercise, the input is validated in order to check if the mandatory fields (all of them except the stimulus and the distractors) are filled.

The caregiver can also list and edit the exercises available, as shown in Figure 4.8.

This implied the creation of new pages for creating, updating and listing the exercises, as well as two new classes (and tables), one per each exercise variation, which inherit their common attributes from a base class. The attributes for each of these classes are the ones described in Section 4.2.2.
Moreover, the exercises’ base class can be extended every time a new type of exercise that shares the same attributes is implemented.

Concerning the multimedia resources, the caregiver is able to upload images, also specifying the title of the resource and the topic it relates to (see Figure 4.9b). The image formats currently supported include, jpg, png, gif, bmp and tiff. As with the exercises, submission triggers input validation to check if all the fields are filled. It is also possible for the caregiver to list the available resources (Figure 4.9a) and well as to edit them (Figure 4.9c).

Implementation-wise, we were able to take advantage of existing code to deal with the multimedia files in terms of the file system. However, in terms of the application, we used our own logical repre-
sentation of these files as resources for exercises, so, similarly to the exercises, we had to create new pages for creating, updating and listing multimedia resources, as well as classes and tables to represent them.

**Users and preferences**

The caregiver module allows to create, list and manage child users. When creating a child user, the caregiver should specify a unique user name, a password, as well as the child’s first and last name, birth date and sex (Figure 4.10a). Submitting a child creation form triggers input validation to check if all mandatory fields are filled and whether both password fields match. It is also verified whether the specified user name is already in use by another child. Figure 4.10 shows the screen for creating, listing and editing child users.

![Create a new child user](image1)

![List child users](image2)

![Edit a child user](image3)

Figure 4.10: Creation and management of child users.

Our users have different characteristics than those in VITHEA. Therefore, we needed to store different information about them. Once again, we had to implement a new user creation form featuring such information and an interface to list the existing users, as well as to edit them. Similarly, we created new classes and tables to represent user information: one for the user credentials and another one for the remaining information provided on user creation.

Besides the basic features described so far, we added the possibility for the caregiver to customize certain aspects of the child’s module. One of those aspects is the animated character's utterances in specific situations (see Figure 4.11), namely:

- When the child logs in for the first time in the child’s module;
• After each subsequent login in the child's module;
• When presenting the list of exercise types;
• When the child answers correctly to an exercise;
• When the child finishes a session of exercises.

Figure 4.11: Animated character’s utterances customization.

If these utterances are never edited for a certain child, the following default utterances will be used: “Bem-vindo” (“Welcome”) when the child logs in; “Escolhe um exercício na lista à tua direita!” (“Pick an exercise in the list on the right!”) when presenting the list of exercise types, “Muito bem!” (“Very good!”) when the child answers correctly to an exercise and “Parabéns!” (“Congratulations!”) when the child finishes a session of exercises.

Another customizable parameter is the set of reinforcement images to be used for each child user when they choose a correct answer. The caregiver can then upload images with the format jpg, png, gif, bmp or tiff (see Figure 4.12a), list all available images (see Figure 4.12b) and select which images should be used for each child (see Figure 4.12c). When a new image is uploaded, it does not become immediately active for the existing child users, since the content might not be reinforcing for all of them.

Implementation-wise, there were no customization options in VITHEA (the animated character only uttered predefined sentences and there was no visual reinforcement concept), so we had to implement these features from scratch. However, in the case of the reinforcement images, since file management is also needed, we took advantage of the code we used to deal with the exercise resources.

4.2.4 Child’s module

The child’s module is based on the patient’s module. After trying both the web and the mobile versions, we decided to focus our efforts in adapting the mobile one due to the following reasons:

• The plug-in required to run the animated character (Unity Web Player) is about to be discontinued in most popular browsers (also, users might not be allowed to install it on certain computers);
The caregivers we consulted displayed preference towards using a mobile device for the child’s app, as it is more practical (and, as seen before, the web version of the child’s module is not supported in mobile browsers);

In the evaluation to the mobile version of VITHEA’s patient module, users mentioned that they preferred that version to the web one.

In what regards to interaction, after logging in, the child is presented with the main screen, which shows the list of types of exercises (Figure 4.13a). When choosing one of them, the first exercise shows up. The animated character utters the question, and the exercise area is filled with the stimulus and the possible answers in a random order (Figure 4.13b). Picking the correct answer on the first try will lead to a reinforcement image (Figure 4.13c). A click over any other answer will lead to prompting in order to help the child picking the correct answer: the clicked distractor disappears, the correct answer is highlighted and the remaining answers are uttered by the animated character (Figure 4.13c). If, after that, the child picks the correct answer, a weaker reinforcement screen, composed of the question stimulus and the correct answer, is shown (Figure 4.13d).

The child can also click on the arrow located in the bottom right corner to skip the current exercise or to proceed to the next exercise after solving the current one. In the application’s settings menu, there are also options to end the exercise session (saving all the child’s performance data regarding that session) and to go back to the main screen (without saving any information). When the exercise session ends (either because the user chose so or because all the exercises under the chosen type were solved), some information about child’s performance data is shown, namely: number of solved exercises; number of correctly answered exercises; number of skipped exercises; average number of times the child picked a
distractor per exercise (Figure 4.13f).

Another feature we added was the possibility of having the animated character repeating her most recent utterance by clicking a repeat button near the character.

Regarding implementation, we had to create new layouts, as well as some new buttons, in order to display the new exercise types list, the multiple choice exercise, the reinforcement views, the child’s performance and the settings menu. We also had to add some methods to ensure the information flow among different views and to perform the requests to the server, since all the information regarding exercises, resources, users and preferences is on the server’s side.

In order to perform requests to the server on the background, our request classes extend Android’s AsyncTask\(^\text{17}\) class. Figure 4.14 depicts the communication between the mobile application and the following web services: the service to log the child into the application and retrieve the messages that the animated character should utter when the child logs in and enters the main screen; the service to

\(^{17}\)http://developer.android.com/reference/android/os/AsyncTask.html (last access on 04/10/2015)
retrieve the exercise list from the type that was selected in the main screen, as well as the preferences for the reinforcement images and animated character utterances associated with the logged child; finally, the service that sends the child's performance data to a service that saves such data into the platform's database.

![Communication between the child application and the server.](image)

**Figure 4.14: Communication between the child application and the server.**

We also use `AsyncTask`'s subclasses to load an image from a given Uniform Resource Locator (URL) (this one was already available in VITHEA) and to send the utterances to the Unity player in which the animated character is displayed. The latter receives as a parameter zero or more sentences and sends each string in a different request to the Unity player. In order to make sure the character is able to utter a given sentence before receiving another request, we get the task to sleep in between requests, during an amount of time that depends on the sentence's length.

### 4.3 Summary

In this chapter, we presented an overview of Virtual Therapist for Aphasia Treatment (VITHEA), the platform that was the starting point for our work and we also described our solution, VITHEA-Kids, in terms of its features as well as the main modifications that were necessary in order to adapt VITHEA's modules to our problem.

In both cases, there are two modules: one where the focus is to solve exercises (the patient's/child's module) and another where the focus is to create the exercises to be solved. However, given that the target users for each platform are different (aphasia patients and their therapists in the case of VITHEA, children with ASD and their caregivers in the case of VITHEA-Kids), many differences arise between the two platforms:

- The types of exercise available are different: in VITHEA, they aim for the oral naming of words, while in VITHEA-Kids there are two variations of multiple choice exercises: one where the user should click on the right word and another where the user should click on the correct image;
- Each platform stores different information regarding multimedia resources and users;
- VITHEA-Kids introduces the concepts of prompting (helping the user to choose the correct answer) and visual reinforcement (rewarding the user after a correct answer by using an image that interests them);
- VITHEA did not allow for any customizations, while VITHEA-Kids allows for the customization of the animated character's utterances as well as of the visual reinforcement.
Automatic generation of multiple choice exercises

In this chapter, we present a system for automatic generation of multiple choice exercises. The motivation behind this system is to provide exercises, automatically generated according to certain parameters, aiming to ease the task of creating exercises.

In order to achieve this goal, we developed two different modules: the ExerciseGeneration module, responsible for generating an exercise composed of a correct answer, a question (based on a given template) and a set of distractors (incorrect answers), and the WordImageLookup module, responsible for extracting information (words and image URLs) organized according to relationships among words (synonymy, hyperonymy and hyponymy). The ExerciseGeneration module makes use of the WordImageLookup to retrieve the words and the URLs of the images to be used in the exercises. We intend to make both modules available so they can be integrated not only in VI THEA-Kids, but in any other project.

In the next sections, we present an overview of these two modules and describe the process of generating each component of the multiple choice exercise: the correct answer, the question and the set of distractors.

5.1 System overview

The main idea of our system is to allow the generation of content for multiple choice exercises of two types: one where the answers are in the form of text (to identify an image) and another where the answers are in the form of images (to illustrate a word). Each exercise is generated given as input the exercise topic (e.g., “Animals”), a template to guide the question generation (composed by an immutable part and a variable whose value can change depending on the exercise’s topic) and information regarding the number of distractors and their proximity to the given topic. The output is an exercise composed by a question/instruction, the correct answer to the question and a set of distractors.

In order to generate the content for the exercises, we need resources that associate words with
pictures. As discussed in Chapter 3, one of such resources is ImageNet, which comprises a structure of synsets (sets of synonyms) organized in a hierarchy based on the relationships of hyperonymy/hyponymy among them and then associates a set of image URLs to each synset. In our system, we consider that the resources containing the words and image URLs should be provided by the programmer of the platform which will use our system, as long as they follow ImageNet’s structure.

From this main idea, we identified two distinct responsibilities: 1) generating content that constitutes a valid exercise, and 2) retrieving words and/or images according to the relationships among them. Therefore, we decided to delegate these responsibilities into two different modules: ExerciseGeneration and WordImageLookup, respectively.

Figure 5.1 provides a black-box view of the main components of our system, in which the components with a thick line (ExerciseGeneration and WordImageLookup) constitute our contributions (TreeTagger is an external module).

An example of input could be the topic “Emoções” (Emotions), the question template “Que < $variable > é este?” (Which < $variable > is this?), “3” as the number of distractors and “true” as whether the the distractors should be related to the given topic. For this input, a possible output could be the question “Que emoção é esta?” (Which emotion is this?), the correct answer “alegria” (happiness) (together with an image of a happy person) and the distractors “tristeza” (sadness), “raiva” (anger) and “medo” (fear).

5.1.1 ExerciseGeneration module

The ExerciseGeneration module is responsible for generating the content for multiple choice exercises. As depicted in Figure 5.2 it features the following components:

- ExerciseGenerator: Provides the methods for generating the two types of exercises mentioned earlier. This component communicates with WordImageLookup to retrieve the words and images needed;
- TemplateParser: Responsible for parsing the question template and replace the variable with a word based on the exercise’s topic;
- GrammarModule: Responsible for making sure the generated question is syntactically correct. This component invokes TreeTagger to perform PoS tagging on the generated question;
• **SpellingModule**: Responsible for altering the gender and/or the number of a word. This component is used by GrammarModule when problems regarding gender and/or number agreement are found;

• **ResourceParser**: Responsible for parsing the files with the gender and number substitutions.

This module was implemented in Java, therefore we used a Java wrapper for TreeTagger (tt4j). The instructions to use this module in another project can be found in Appendix B.

### 5.1.2 WordImageLookup module

The **WordImageLookup** module is responsible for retrieving words and/or images from a hierarchy of synsets (sets of synonyms) organized according the relationships of hyperonymy/hyponymy among them. The need for this module arises from the fact that we could not find any tool that allowed to programmatically search through the ImageNet structure.

As depicted in Figure 5.3, **WordImageLookup** only comprises two components: the module’s homonymous class **WordLookupImage**, which provides a set of methods to retrieve the words and images, and the **ResourceParser** class, which converts an Extensible Mark-up Language (XML) representation of the synset hierarchy into a tree structure to be used in runtime.

---

1 Available in [https://github.com/reckart/tt4j](https://github.com/reckart/tt4j) (last access in 06/10/2015)
WordLookupImage class currently provides methods that traverse the tree obtained by ResourceParser in order to:

- Get all the leaf synsets;
- Get the root hyperonyms of a given word;
- Get all the leaf hyponyms of a given word;
- Get all the URLs of the images associated with a given synset identifier.

However, it can be extended to feature more ways to take advantage of the relations among words and images.

As for the XML file to be parsed by ResourceParser, it should follow a schema inspired in ImageNet’s schema, comprising a root tag called ImageNetStructure and a hierarchy of nested synset tags, in which the outer tags contain the hyperonyms of the inner tags. Each synset tag comprises a unique identifier (id), a set of synonyms separated by commas (words) and an optional definition for those synonyms (gloss). The schema for validation of the XML file is detailed in Appendix C.1 and below we present a very short example of a valid file:

```xml
<ImageNetStructure>
  <synset id="c100" words="animal">
    <synset id="101" words="dog"/>
    <synset id="102" words="cat"/>
  </synset>
</ImageNetStructure>
```

Similarly to ImageNet, the URLs for the images of a certain synset are kept in a text file named after the synset’s id (e.g., considering the example above, all the URLs of images of dogs should be in a file called 101.txt).

The WordImageLookup module was also implemented in Java. The instructions to use this module in another project can be found in Appendix C.2.

### 5.2 Generation process

As mentioned before, the ExerciseGeneration module provides the methods for generating two variations of multiple choice exercises. Both follow the same approach; the only difference lies in the set of distractors returned: one of the variations returns a set of textual distractors, while the other returns a set of image distractors.

As we can see in Figure 5.4, the exercise generation process starts with the generation of the the correct answer and the corresponding stimulus (being one of them a word and the other an image, or vice-versa, depending on the type of exercise), given a certain topic. Both the processes of generating a question and generating the set of distractors depend on having the correct answer: if the question template contains a variable, the variable will be replaced by a hyperonym of the correct answer; as for the distractors, they must be combined with the correct answer in such a way that it is possible to solve the exercise (meaning the set of distractors cannot contain the correct answer nor a synonym of it).
The question and the set of distractors can be generated in any order. In both cases, the exercise’s topic is given as input, but in the case of the question generation, a question template is also provided, while in the case of the set of distractors, the input includes the number of distractors to be returned and a boolean indicating whether the distractors should belong to the exercise’s topic. The question, the correct answer and the set of distractors are then assembled in a structure to be returned by either one of the two methods provided.

5.2.1 Correct answer generation

Our method to generate a correct answer consists in picking an hyponym of the topic provided. The process to achieve this goes as follows:

1. The ExerciseGenerator component asks WordLookupImage for all the leaf hyponyms of the topic;

2. If no hyponyms were found, an exception is thrown, and the process of generating a correct answer ends. Otherwise, a random synset is picked out of the returned leaf hyponyms;

3. The ExerciseGenerator component asks WordLookupImage for all the images associated with the id of the synset picked in Step 2;

4. If the set of images associated with that id is empty, we go back to Step 2. In order to avoid an endless loop, once the number of iterations exceeds the size of the hyponyms list, we pick the first synset that has a non-empty set of images associated;

5. A random word is picked out the last picked synset;
6. A random image is picked out of the image set associated with the picked synset.

Taking the example input from Section 5.1 in Step 1, WordImageLookup would retrieve the leaf hyponyms of the word “emoção” (emotion) (“Emoções” after being normalized). If any hyponyms were found, a random synset would be selected (e.g., the one composed by the words “alegria, contentamento” – happiness, joy). If this synset is associated with one or more images, a random word (e.g., “alegria”) and a random image associated with the synset would be selected and returned.

5.2.2 Question generation

The process of generating a question involves several steps (unless the template provided does not include a variable, in which case the template remains unaltered). This process is depicted in Figure 5.5 and it goes as follows:

1. The ExerciseGenerator component asks WordLookupImage for all the root hyperonyms of the correct answer;

2. The first synset to contain a word that matches the topic is picked;

3. If no synsets contain the topic, a synset is randomly picked from the root hyperonyms list. Otherwise, a random word is picked from the synset that contains the topic;

4. The template is parsed in order to find the variable (a word surrounded by `<%` and `>`) ;

5. If a variable is found, it is replaced by the word picked in Step 3. Otherwise, the process ends here and the template is returned as the final question;

6. The resulting sentence is PoS tagged using TreeTagger;

7. The resulting tagged sentence is sent to GrammarModule to be checked for inconsistencies regarding number agreement between the variable’s replacement and the rest of the sentence. If any inconsistency is found, it is fixed by SpellingModule;

8. The same as Step 7 but regarding gender agreement.

Again, taking the example input from Section 5.1 in Step 1, WordImageLookup would retrieve the root hyperonyms of the word “alegria”. If any hyperonyms are found, the normalized topic (“emoção”) is searched within the obtained hyperonyms. The first synset that contains the topic will be selected, and a random word will be selected from that synset (e.g. “emoção”). The variable in the template would then be replaced by “emoção”, resulting in the sentence “Que emoção é este?”. This sentence is PoS tagged in order to check the number and gender agreement. Since all the words are singular, no changes regarding number would be performed. However, there is a gender inconsistency: the noun “emoção” is feminine and the determinant “este” (this), which refers to “emoção”, is masculine. Therefore, the gender of “este” will be fixed by replacing the final “e” with an “a”, according to the substitution rules used by the SpellingModule.

Out of the three main tasks that compose the exercise generation process, the question generation is the most challenging, since it involves making sure a sentence is syntactically correct (which implies PoS tagging, inconsistency detection regarding gender and number agreement, and performing substitutions where needed), although our job was made easier by TreeTagger, which is responsible for the PoS tagging. In order to process the morphological tag returned by TreeTagger in a easier way, we created a class TaggedElement, which provides a set of methods that determine the morphological
class, gender and number of a word by matching the respective tag returned by TreeTagger against a set of regular expressions.

As for the detection of issues regarding gender and number disagreement, we focused on a set of combinations likely to occur in multiple choice questions in Portuguese. The combinations considered were the following:

- The variable is a **noun** preceded by a **demonstrative determinant**:  
  - *Como se chama esta planta?*
• The variable is a **noun** preceded by an **article**:
  – Como se chama a planta?

• The variable is a **noun** preceded by a **demonstrative determinant** and a **verb**:
  – Como se chamam estes animais?

• The variable is a **noun** preceded by an **article** and a **verb**:
  – Como se chamam os animais?

• The variable is a **noun** preceded by a **demonstrative determinant**, a **verb** and an **interrogative pronoun**:
  – Quais são estes animais?

• The variable is a **noun** preceded by an **article**, a **verb** and an **interrogative pronoun**:
  – Quais são os animais?

• The variable is a **noun** followed by a **demonstrative determinant**:
  – Que planta é esta?

• The variable is a **noun** followed by a **verb** and a **demonstrative determinant**:
  – Que animais são estes?

When an inconsistency is detected, all the words that should agree in gender and number with the variable but fail this criterion suffer substitutions in order to match the variable’s gender and number.

### 5.2.3 Distractor generation

Regarding distractors, we defined the following criteria:

• If the distractors are supposed to be related to the given topic, we collect them from the hyponyms of the topic, otherwise, we collect them from any synset at the lowest level of the synset tree;

• The distractors should not be synonyms of the correct answer, otherwise that would make the exercise impossible to be solved;

• There should be no repeated distractors.

Thus, for each textual distractor to be generated, the process goes as follows:

1. If the distractor should belong to a given topic, ExerciseGenerator asks WordImageLookup for the list of all the leaf hyponyms of the topic. Otherwise, it asks for a list of all the leaf synsets;

2. A random synset is picked from the list obtained in Step 1.

3. If the synset contains the correct answer or any previously generated distractor for the current exercise, we go back to Step 2. In order to avoid an endless loop, once the number of iterations exceeds the size of the list, we pick the first synset that does not contain the correct answer nor any previously generated distractor. If no synsets match these criteria, an exception is thrown, and the process of generating the current distractor ends;

4. A random word is picked out the last picked synset.
This process repeats itself as many times as the number of distractors specified.

Considering the example input from Section 5.1 since the distractors should belong to the topic “emoções”, *WordLookupImage* is asked for the leaf hyponyms of the normalized topic (“emoção”). A random synset is selected out of the hyponyms list (e.g., “tristeza, angústia” – sadness, anguish). Since the selected synset does not contain the correct answer and no distractors have been generated so far, a random word (e.g., “tristeza”) is picked. The remaining distractors are obtained through a similar process (which could return, for instance, “raiva” and “medo”).

As for the process of generating image distractors, it goes as follows:

1. If the distractor should belong to a given topic, ExerciseGenerator asks *WordImageLookup* for the list of all the leaf hyponyms of the topic. Otherwise, it asks for a list of all the leaf synsets;
2. A random synset is picked from the list obtained in Step 1;
3. The ExerciseGenerator component asks *WordLookupImage* for all the images associated with the id of the synset picked in Step 2;
4. If the synset contains the correct answer or any previously generated distractor for the current exercise, or if the set of images associated with the id of the picked synset is empty, we go back to Step 2. In order to avoid an endless loop, once the number of iterations exceeds the size of the list, we pick the first synset that does not contain the correct answer nor any previously generated distractor and that also returns a non-empty set of images. If no synsets match these criteria, an exception is thrown, and the process of generating the current distractor ends;
5. A random word is picked out the last picked synset;
6. A random image is picked out the last picked synset.

This process repeats itself as many times as the number of distractors specified.

An example of this process would be similar to the one for word distractors, but here the selected synsets would have to include images. The data returned would then comprise, for example, an image of a sad person, of an angry person, and of a scared person.

### 5.3 Summary

In this chapter we described a system for the automatic generation of multiple choice exercises, which can be integrated in VITHEA-Kids platform or in any other platform where such exercises could be useful. This system is composed of two independent modules: the *ExerciseGeneration* module, responsible for generating correct multiple choice exercises about a certain topic, and the *WordImageLookup* module, which provides a way to retrieve words and images based on a hierarchy of synonym sets.

The *ExerciseGeneration* module splits the generation process into three main tasks: 1) generating a correct answer; 2) generating a question about that answer given a template; 3) generate a set of distractors in the form of words or images. All of these tasks involve asking *WordImageLookup* for words and/or images. The second task, however, is more intricate, as it also involves checking the gender and number agreement of the generated question and fixing any inconsistencies that might be found.
In this chapter, we present the procedures and results of evaluating each component of this dissertation: VITHEA-Kids’ modules for caregivers and children (in Section 6.1), and the exercise generation module (in Section 6.2).

6.1 VITHEA-Kids

In order to evaluate our platform, we performed two distinct procedures: one to assess the quality of the caregivers’ experience while using their module, and another to assess the impact of using the child’s module in teaching a new word to a child with ASD.

6.1.1 Evaluation with caregivers

For the purpose of evaluating the caregiver’s module, we designed an experiment consisting of five tasks and a questionnaire to assess the participants’ satisfaction regarding the ease, effectiveness and efficiency of performing each task in our platform. We contacted sixteen caregivers from diverse backgrounds (therapists, parents, etc.) and sent the application’s URL, the task guide and the questionnaire URL to those who displayed interest in participating.

The task guide (presented in Figure D.1 of Appendix D.1.1) comprised the following tasks to be performed in the caregiver’s module:

1. Creation of a new multiple choice exercise in which the stimulus was an image and the answers were textual;
2. Modification of the title of a multimedia resource;
3. Creation of a new child user;
4. Customization of one of the animated character’s utterances for a specific child;
5. Visualization of the currently active reinforcement images for a specific child.
As for the questionnaire, it featured four main sections (as depicted in Figure D.2 of Appendix D.1.2), namely:

- An introductory section to characterize the participants, by asking them what is their role as caregivers and whether their children have any impairment (Figure D.2a);

- A section where each task is evaluated in terms of how easy and fast it was to complete the task and of whether there were errors or not. Participants were also able to write down their main difficulties, observations and suggestions regarding each task (Figure D.2b);

- A section where the overall user experience is evaluated through the classification of a set of affirmations in a scale of 1 (Totally disagree) to 5 (Totally agree) (Figure D.2c);

- A section where the participant is asked about their interest in a potential module for the automatic generation of exercises, as well as about other suggestions that they might have (Figure D.2d).

Out of the contacts established, we obtained eight participations, out of which only seven were valid (i.e., complete). Among the valid participations, there were three therapists, three parents, zero teachers or educators and one participant who did not match any of these roles. Also, in their role as caregivers, three participants deal with children with ASD, two participants deal with children with other impairments and three do not deal with children with special needs. The only invalid participation was from a therapist who deals with children with ASD.

For each task, participants classified their agreement with three statements: “I was able to easily complete the task”, “I was able to complete the task quickly” and “The task was completed without any error from the application” in a scale of 1 (Totally disagree) to 5 (Totally agree). As we can see in Figure 6.1, most participants agreed that the tasks were easy and fast to perform and were able to execute them without errors. However, the results differ according to the task. While Tasks 2 and 5 (Modification of an existing resource and Visualization of the reinforcement preferences, respectively) achieved full agreement, in Tasks 1 and 3 (Creation of a new exercise and Creation of a new child user, respectively) one participant reported errors. Also, Task 4 (Customization of the character’s utterances) there were two participants who did not fully agree with all the statements.

In order to better understand the agreement classifications, we also asked the participants to describe their experience regarding each task. Aside from the errors reported by one of the participants, which were immediately fixed, Tasks 1 and 3 did not receive any comment, nor did Task 2. However, in Tasks 4 and 5, some participants found it hard to understand how to select a child, given the instruction in the guide. Participants were also confused with the use of the name of the animated character (“Catarina”) in the application’s interface.

Regarding overall user experience, participants classified the following set of statements in a scale of 1 (Totally disagree) to 5 (Totally agree):

- “I found the application was easy to browse”;

- “The discourse used in the application was easy to understand”;

- “The feedback given by the application (in error or confirmation messages) was clear”;

- “The kind of exercises allowed to be created are useful for the children I deal with”;

- “Using this application would allow me to save time in my daily routine”;

58
Figure 6.1: Results on user satisfaction regarding each task

- “I would like to use this application again”;

We also asked the participants about their interest in the possibility of having exercises automatically created by the platform (while keeping the possibility of manually creating them), using a scale of 1 (Not interested at all) to 5 (Very interested).

As we can see in Figure 6.2a, all participants fully agree that the platform was easy to browse and makes use of a discourse easy to understand. However, some participants disagree concerning the clarity of the feedback given in error or confirmation messages. Furthermore, one participant considers that the kind of exercises supported is not useful for their children and there is also one participant that would not like to use this application again.

Additionally, as we can see Figure 6.2b, participants are either interested or very interested in the possibility of having exercises automatically generated by the platform.
Finally, we asked the participants an open question in order to get general feedback and further suggestions. One of the participants emphasized the importance of using a mobile application regarding the child (in the guide, we did not mention how would the exercises be presented to children). One participant was interested in the possibility of uploading other kinds of multimedia content, such as videos. Two participants provided suggestions of new exercises/activities to incorporate, namely:

- Exercises to train math skills (arithmetic operations, number identification);
- Exercises where it would be given the shape of a face and the child would have to add the remaining components (mouth, eyes, nose, etc.) to represent a certain emotion and would also identify the represented emotion;
- Exercises where it would be given an image of a place (e.g., a farm or house division) and the child would have to add the remaining components (e.g., animals, people, furniture), as well as to identify and describe the resulting scenario.
- Exercises to develop pragmatic skills;\footnote{Pragmatics is the study of the language usage given the relationship between the speakers and the surrounding context, as seen in \url{http://www.priberam.pt/dlpo/pragm%C3%A3tica} (last access in 14/10/2015)}
- Other kinds of exercises and activities that would be asked by therapists or parents.

Given that the amount of participations (seven) is not enough to have statistical relevance, we do not provide further statistical treatment of these results.

6.1.2 Evaluation with a child

The evaluation of the child’s module is currently taking place. For this evaluation, we chose to follow a research design called \textit{single-subject design}. In this design, the subject serves as their own comparison term, in opposition to group designs, in which there are two or more groups, and one of them serves as term of comparison (control group). The main differences between these types of design reside in the way research data is analysed (single-subject designs rely on visual inspection of the data collected, while group designs use measures of relationship strength, which makes it easier to summarize the results and compare them across different studies, but might omit relevant information at individual level) and in the external validity of the studies (studies that follow a single-subject design often feature few participants, which makes it difficult to generalize the results to other subjects in the population) \cite{55}. Despite the advantages of group designs, our choice goes to a single-subject design, which is the one used in the majority of the studies featuring subjects diagnosed with ASD or other impairments \cite{10,14,29,44}, since each individual with ASD has a unique set of symptoms, characteristics...
and needs, making it difficult to generalize results.

Single-subject design is based on the principle that, if an intervention is effective, a change in status from the period prior to intervention to the period during and post-intervention should be noticeable [23]. The three main components of this design are:

- **Repeated measurements** of the subject's progress regarding the target behaviour;
- **Baseline phase**: The period prior to the treatment. The measurements collected in this phase will serve as comparison term to those collected during and after the execution of the treatment. Baseline is represented by the letter \( A \);
- **Intervention phase**: The period during which the treatment is applied. It should last as long as the baseline phase, in order to facilitate the comparison against baseline measured values. This phase is represented by the letter \( B \).

Single-subject design comprises the following variants:

- **Basic Design (A-B)**: This design only features one baseline phase followed by one intervention phase, which means it does not allow to take conclusions about whether the cause of eventual improvements was the intervention or some external factor;
- **Withdrawal Designs**: These designs are based in the premise that, if a treatment is effective, the target problem should only improve during the intervention phase and it should worsen when the treatment is removed. With such assumption in mind, it would be possible to conclude that external events had no influence in the improvements; however, the primary goal of a treatment is to eliminate the target problem, so it is desirable that the therapy effect lasts beyond the intervention phase. Therefore, in order to use these designs for research purposes, the length of the intervention might need to be limited. There are two kinds of withdrawal designs:
  - **A-B-A**: This design allows to assess whether the effect of the treatment persists beyond the intervention phase and it might be indicative of for how long it persists (if the follow-up baseline period is long enough). However, it faces the limitation that the subject might not be interested in proceeding to a follow-up phase;
  - **A-B-A-B**: This variant allows to conclude that the treatment has improved the target behaviour, even if its effect is not reversed during the second baseline, as long as there are improvements during both intervention phases.
- **Multiple Baseline Designs**: This category of designs tries to solve the limitations of withdrawal designs by including additional **subjects**, **target problems** or **settings** (at least three of them) and running the evaluation for all of them at the same time (concurrent) or with a certain delay (non-concurrent). This way, the second and third cases serve as a control for external events in the first case, the third case serves as control for external events in the second case, and so on;
- **Multiple Treatment Designs**: This kind of designs consists in changing the nature of the intervention over time, in different phases. It can assume the following variants:
  - **A-B\(_1\)-B\(_2\)-B\(_3\)**: Changing the intensity of the intervention over time;
  - **A-B-C-D**: Changing the actual intervention over time;
- **Designs for monitoring subjects**: These designs should be used when it is not possible to establish an initial baseline, and might consist of only one phase (\( B \)) or include a follow-up period (\( B-A \)).
Out of these, we chose to follow the A-B-A withdrawal design, which comprises three phases – baseline, intervention and follow-up –, as depicted in Figure 6.3. In this figure, each point corresponds to the measurement of a different session.

This evaluation aims to assess whether the prompting and custom reinforcement features of our application, as well as the possibility of creating exercises with a variable number of distractors, are effective in teaching a new word to a child with ASD or not. The application is being tested with a male seven year old child diagnosed with ASD who is in the process of learning how to read and write, with the help of a therapist. The target word is “Javali”. The sessions are taking place in the child’s classroom, and are expected to last about fifteen days.

The first phase of the evaluation, the baseline, focused on accessing whether the child was able to select the word “Javali” (boar) when an image of a boar shows up, without prompting or reinforcement (see Figure 6.4). As expected, the child never selected the correct answer (“Javali”), so this phase ended after four sessions, which is the minimum number of sessions required to consider that a stable pattern was observed.

In the intervention phase, the child will have to solve exercises that will start without distractors and progressively increment the number of distractors. In this phase, prompting will be activated in case the child picks a distractor instead of the correct answer. Selecting the correct answer on the first try of a give exercise will lead to visual and audio reinforcement, while selecting it after having selected any distractor will only lead to the same image stimulus, followed by the correct answer. This phase will only end when the child selects the correct answer during two consecutive sessions, without prompting.
Finally, in the follow-up phase, prompting and reinforcement will be deactivated again in order to assess whether the child has learned the target word or not.

6.2 Automatic generation of exercises

The evaluation of our module for automatic generation of exercises consisted solely in the assessment of the quality of the generated distractors considering a question about a certain topic and the correct answer to that question, given that, in the context of VITHEA-Kids, the distractors are a key component in learning how to generalize a concept. For this purpose, we created an interface for the exercise generation module and set up the following resources:

- An XML file comprising three root synsets: “animal” (animal), “alimento” (food) and “transporte” (transportation). These synsets had twelve, twelve and seven hyponym synsets each, respectively. All the words featured were nouns;
- A set of twenty-three text files, each containing one or more image URLs related to a certain hyponym synset (which means six synsets did not have an URL file associated).

We asked thirty people to participate in the assessment process. Participants were mostly between 20 and 30 years old, resided in Portugal, their academic qualifications range from high school to master’s degree and were currently university students and/or researchers. All of them have a background in computer science and engineering, although there were participants who also had a background in electro-technical engineering, law, and arts/multimedia.

The evaluation interface was developed using Java Swing\[^2\] The generated questions, answers and distractors, as well as the responses given by the participants were saved into Comma-separated Values (CSV) files, to facilitate their processing. The evaluation comprised four steps (the full questionnaire is shown in Figure D.3 of Appendix D.2):

1. An initial screen with the context and the instructions regarding the evaluation process;
2. A screen to evaluate the quality of the word distractors in the context of an exercise where the stimulus would be an image and the answers would be textual (Figure 6.5a);
3. A screen to evaluate the quality of the image distractors in the context of an exercise where the stimulus would be a word and the answers would be images (Figure 6.5b);
4. An open question for any suggestions and observations from the participants.

In Steps 2 and 3, the input is a topic (e.g., “Animals”), a question about that topic (e.g., “What is the name of this animal?” or “Which image refers to this animal”, respectively) and the correct answer (e.g., the word “cat” or an image of a cat, respectively). In each step, once the participant clicks on a button to generate the distractors, two sets of three distractors each are displayed: the first set should only be composed of distractors related to the given topic, while the second set might include distractors about any topic (including the topic taken as input). For each distractor, the participants should indicate whether they think the distractor makes sense in the context of the given input or not (i.e., if it is possible to solve an exercise composed of the given question and a set of answers containing the given correct answer and the distractor being evaluated). Participants should also rate each distractor in terms of how

\[^2\] http://docs.oracle.com/javase/8/docs/technotes/guides/swing/index.html (last access on 14/10/2015)
adequate they find it, in a scale of 1 (Poor) to 5 (Very good).

Regarding the generation of word distractors, as we can see in Figure 6.6a and Table 6.1 out of the 180 generated distractors (6 per participant), 143 (79.44%) were marked as making sense. In what respects to quality, the most frequent rating was 5 (very good), given to 83 distractors (46.11%). The average quality rating was of 3.75 (std dev = 1.41).
As for the generation of image distractors, as we can see in Figure 6.6a and Table 6.2, 138 out of the 180 generated distractors (76.67%) were marked as making sense. In what respects to quality, the majority of the distractors (91, i.e., 50.56%) was rated with 5 (very good). The average quality rating was of 3.82 (std dev = 1.44).

Splitting the results for distractors that should be related to the topic and distractors that did not have a topic constraint, we can notice a significant difference in the results for each of these categories, regarding both word and image distractors (as we can see in Figures 6.8, 6.9, 6.10 and 6.11, and Tables 6.3, 6.4 and 6.5). 92.22% of the word and image distractors generated under a topic constraint were marked as making sense, and the majority of the word and image distractors were rated as very good (57.78% and 61.11%, respectively). The average quality rating was of 4.26 (std dev = 1.03) for word distractors.
distractors and 4.24 (std dev = 1.1) for image distractors. As for the distractors with no topic constraint, the percentage of distractors marked as making sense drops to 66.67% (words) and 61.11% (images), and there is a greater balance among each quality rating’s percentage. The average quality rating was of 3.24 (std dev = 1.55) for word distractors, and 3.39 (std dev = 1.59) for image distractors.

![Figure 6.8](image1.png)

(a) Word distractors’ adequacy  
(b) Word distractors’ quality rating

Figure 6.8: Results for adequacy and quality of the word distractors related to a given topic

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Average</strong></td>
<td>4.26</td>
</tr>
<tr>
<td><strong>Standard Deviation</strong></td>
<td>1.03</td>
</tr>
<tr>
<td><strong>Median</strong></td>
<td>5.00</td>
</tr>
<tr>
<td><strong>Mode</strong></td>
<td>5.00</td>
</tr>
</tbody>
</table>

Table 6.3: Results for quality of the word distractors related to a given topic

![Figure 6.9](image2.png)

(a) Word distractors’ adequacy  
(b) Word distractors’ quality rating

Figure 6.9: Results for adequacy and quality of the word distractors without a topic constraint

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Average</strong></td>
<td>3.24</td>
</tr>
<tr>
<td><strong>Standard Deviation</strong></td>
<td>1.55</td>
</tr>
<tr>
<td><strong>Median</strong></td>
<td>3.00</td>
</tr>
<tr>
<td><strong>Mode</strong></td>
<td>5.00</td>
</tr>
</tbody>
</table>

Table 6.4: Results for quality of the word distractors without a topic constraint

Finally, regarding suggestions and final observations, participants made the following comments:

- Two participants suggested that, in the sets of distractors that should be related to the topic, there should be a stronger relation between the correct answer and each distractor (e.g., if “parrot” is the correct answer, the distractors should also be birds, for instance);
Figure 6.10: Results for adequacy and quality of the image distractors related to a given topic

<table>
<thead>
<tr>
<th></th>
<th>Average</th>
<th>Standard Deviation</th>
<th>Median</th>
<th>Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adequacy</td>
<td>4.24</td>
<td>1.11</td>
<td>5.00</td>
<td>5.00</td>
</tr>
</tbody>
</table>

Table 6.5: Results for quality of the image distractors related to a given topic

Figure 6.11: Results for adequacy and quality of the image distractors without a topic constraint

<table>
<thead>
<tr>
<th></th>
<th>Average</th>
<th>Standard Deviation</th>
<th>Median</th>
<th>Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adequacy</td>
<td>3.39</td>
<td>1.59</td>
<td>4.00</td>
<td>5.00</td>
</tr>
</tbody>
</table>

Table 6.6: Results for quality of the image distractors without a topic constraint

- Three participants suggested that, in the sets of distractors that did not have a topic constraint, there should still be some kind of semantic connection between the correct answer and each distractor (e.g., if the correct answer is “horse”, a possible distractor could be “stable”);
- One participant reported that it was difficult to rate the quality of the distractors without information about the context in which such exercises would be used.

During the evaluation sessions, we were able to notice some difficulties in interpreting the aim of the study. For instance, some participants reported that they felt confused about the aim of the question about the adequacy of the distractor, which might have led to different interpretations of what it means for a distractor to make sense. In fact, given that we evaluated our generation module with a very restricted set, it is expectable that most generated distractors correctly satisfy the topic constraint and do not collide with the correct answer (we can confirm this expectation by inspecting the generated data). Another aspect that we noticed was that different participants had a completely different notion of what it could make an interesting distractor without topic contraints: some participants would penalize the
distractors that belonged to the given topic, while others would penalize the distractors that were too unrelated to the topic or the correct answer. This might explain the differences between the results for distractors under a topic constraint and for the distractors without a topic constraint that we observed in Figures 6.8, 6.9, 6.10, and 6.11.

Finally, according to the suggestions, one participant had in consideration the sets of distractors as a whole, instead of rating each distractor independently.

### 6.3 Summary

In this chapter, we presented the results of evaluating each module of our system, as well as the procedures followed in order to do so.

In the case of VITHEA-Kids, we evaluated our contributions in two ways: the caregiver’s module was tested by a set of seven caregivers, and the child’s module is currently being tested by a child. Regarding the evaluation with caregivers, we created a set of five tasks for them to complete and a questionnaire to assess their satisfaction towards several aspects of the interaction during the execution of the tasks. Most participants reported that they have been able to easily and quickly complete the task, without errors. They also gave positive ratings to several aspects of the overall interaction with the application and six out of seven participants agreed that they would like use the application again. Some participants provided suggestions that could constitute future work in our platform, such as other kinds of exercises in order to develop different skills.

As for the evaluation with the child, which is ongoing, we are following an A-B-A single-subject design in order to assess if the prompting and reinforcement features of our platform are of any help for the child to learn a new word. The baseline phase, which consists of assessing whether the child is able to solve the exercises featuring the new word or not, has just finished and we are now entering the intervention phase (in which prompting and reinforcement will be used in order to teach the new word).

As for the module for automatic generation of multiple choice exercises, we focused on the evaluation of the quality of the distractors generated. We set up an interface where each participant in the evaluation should classify the quality of the sets of distractors generated for a certain topic, question and correct answer. The participants, who were mostly university students and researchers, indicated that the majority of the distractors made sense in the context of the given exercise. Additionally, around 50% of the distractors generated were rated as being Very good in the context of the exercise. However, some participants reported that they would prefer distractors that had some semantic connection with the correct answer.
Conclusions and Future Work

7.1 Summary of the Dissertation

Motivated by the difficulties that individuals diagnosed with Autism Spectrum Disorder (ASD) face in social and communication contexts, several authors have performed studies and developed applications that either aim to provide an alternative mean of communication or a way to develop language skills, since these are also relevant to the development of verbal communication skills. With the increasing popularity of mobile devices, many applications of such kind have been made available in mobile application markets, some of which for free. Nonetheless, there are still difficulties and needs regarding the use of software for individuals with ASD that are not yet addressed. In particular, there is a desire for an individualized user experience, as well as for the accounting of each user’s characteristics, that is not fulfilled by currently available options yet. Additionally, there is a lack of applications that allow to develop language skills regarding the Portuguese language.

Given this context, we had set as our goal to develop a platform in which children with ASD could solve exercises aiming the development of language skills in Portuguese, and where their caregivers could create those exercises and customize several aspects of the interaction with each child. To achieve this goal, we decided to extend an existing platform, the award winner Virtual Therapist for Aphasia Treatment (VITHEA), originally targeted at patients with aphasia and their therapists. This platform was composed by two modules: a module where the patients could solve speech exercises, with the guidance of a talking animated character, and another module for the therapists to create the exercises to be solved. We took advantage of VITHEA's infrastructure, by adapting the Android version of the patient's module into the child's module, and the therapist's module into the caregiver's module, and designated the resulting platform as VITHEA-Kids. In the caregiver's module, we implemented the following features: creation and management of two variants of multiple choice exercises (one where the answers are textual and another where the answers are in the form of images); upload and management of images to be featured in the exercises; creation and management of child users; customization of aspects such as the animated character's utterances and the reinforcement images to display when the child correctly solves an exercise. As for the child's module, we adapted in order to make use of the
multiple choice exercises, as well as of the customizations performed on the caregiver's side.

Considering that the task of creating a considerable amount of exercises can be time consuming for caregivers, another goal we had set was to provide the possibility of automatically generating the exercises' content. For that, we implemented an independent module which, given a topic, a question template and some constraints regarding the distractor answers, returns a multiple choice exercise composed of a question, and a set of answers, in which one is correct and the remaining are distractors. The answers can be in the form of text or images, depending on the variant requested. The exercise generator obtains the words and images to be featured in the exercises through a second independent module.

Finally, we evaluated each part of our contribution as follows: regarding VITHEA-Kids, we asked a set of caregivers to perform some tasks on the caregiver’s module and reply to a questionnaire in which they could rate their experience and add further suggestions; we are also currently evaluating the child's module with a child with [ASD]. As for the exercise generation module, we set up a questionnaire in which participants would have to rate the quality of the generated distractors given a question about a certain topic and the correct answer to that question.

7.2 Final Conclusions and Contributions

In this work, we developed a platform that aims to meet the needs of children with [ASD] and their caregivers, specially those concerning customization. Our platform is composed of two modules: a mobile application for children to solve multiple choice exercises that allow them to acquire new vocabulary and to develop skills such as the ability to generalize concepts, and a web application for the children's caregivers to prepare those exercises and customize several interaction aspects in the child's application. The caregiver's application was evaluated with a set of caregivers; the majority of them agreed that the application was easy and fast to use, and would like to use the application again.

Additionally, in order to ease the task of creating new exercises, we developed a module to automatically generate two variants of multiple choice exercises given a set of parameters, such as the topic of the exercise and a question template. We developed this module independently so it can be integrated either in VITHEA-Kids or any other project. The words and images used in the generated exercises are obtained through a second module, which we also implemented independently so that it can be used in other contexts. The generation of exercises was evaluated as a whole, with the focus on the generation of distractors. In this process, a total of 180 word distractors and 180 image distractors were generated. Out of the distractors generated considering a topic, 92.22% of the word distractors were marked as making sense, achieving an average quality rating of 4.26 (std dev = 1.03), and also 92.22% of the image distractors were marked as making sense, achieving an average quality rating of 4.24 (std dev = 1.1). As for the distractors without topic constraints, 66.67% of the word distractors were marked as making sense, achieving an average quality rating of 4.24 (std dev = 1.55), and 61.11% of the image distractors were marked as making sense, with an average quality rating of 3.39 (std dev = 1.59).

Finally, VITHEA-Kids and the exercise generation module originated a poster that will be presented in ASSETS 2015, an international conference on the use of technology for accessibility.
7.3 Future Work

In what concerns to future work, there are several possibilities that can be explored. On short term, one of the goals should be the integration of the exercise generation module in VITHEA-Kids. Yet, both modules have the potential to be extended with new features.

7.3.1 VITHEA-Kids

Among the features to include in future versions of VITHEA-Kids, there are some that were emphasized by therapists as being extremely relevant, that include, but are not restricted to: the possibility of uploading other kinds multimedia resources (e.g., audio, video); the possibility of using animations as reinforcement; the possibility of sorting the existing exercises in order to control the order in which they appear in the child’s application; the customization of the way prompting is done. Before moving to more ambitious features, it would also be desirable to run a new evaluation process for both applications, featuring more participants.

In a medium term, VITHEA-Kids could be extended with new kinds of exercises and activities, such as the ones suggested by the caregivers that evaluated the platform (see Chapter 6), as well as exercises that make use of other kinds of input, namely ones where the answer can be provided by speech or text introduction (e.g, orally naming an image or correctly rewrite a given word that contained errors). It would also be desirable to have the animated character performing different facial expressions depending on the feedback she’s giving, or, in a longer term, turn the character into an intelligent interactive agent.

7.3.2 Automatic Generation of Exercises

In what regards the automatic generation of exercises, there is also some room for improvements; namely, it would be desirable to explore different approaches regarding the generation of questions in addition to the one we currently follow. It would also be important to run a new evaluation featuring the generation of correct answers and questions.

Concerning the word and image extraction module, in a short term it should provide a larger set of methods, namely methods to get hyperonyms and hyponyms from different levels of the resource tree, or methods to extract words given other criteria (e.g, orthographically similar words). Not only this would be relevant for this module, as it could contribute to the improvement of the exercise generation module, as it would allow to follow some of the suggestions given by the evaluation participants regarding the similarity of the distractors with the correct answer.

In a medium/long term, it would be interesting to add new features to the exercise generation module, namely the generation of paraphrases given a question. This feature could have a positive impact in applications such as VITHEA-Kids, as it could contribute to improve the child’s ability to generalize different formulations of the same question.


[26] Ilie Ghiciuc. Smartphone and tablet usage pattems: The how’s, the why’s and the where’s. in *Thin Slices*, December 2014.


[38] José Paulo Leal and Nelma Moreira. Using matching for automatic assessment in computer science learning environments.


A.1 Communication software

<table>
<thead>
<tr>
<th>Features</th>
<th>TalkinPictures</th>
<th>Tools for Autism</th>
<th>Proloquo2go</th>
<th>MyTalk Mobile</th>
<th>Niki Talk</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>Free</td>
<td>Free</td>
<td>Paid</td>
<td>Free</td>
<td>Free</td>
</tr>
<tr>
<td>Platforms</td>
<td>Mobile (Android, iOS)</td>
<td>Mobile (Android)</td>
<td>Mobile (iOS)</td>
<td>Mobile (Android, iOS)</td>
<td>Mobile (Android, iOS)</td>
</tr>
<tr>
<td>Multimedia content</td>
<td>text, audio, pictures</td>
<td>text, audio, pictures</td>
<td>Audio, pictures</td>
<td>text, audio, pictures, video</td>
<td>text, audio, pictures</td>
</tr>
<tr>
<td>Interface language</td>
<td>Portuguese, English</td>
<td>English</td>
<td>English</td>
<td>English</td>
<td>English</td>
</tr>
<tr>
<td>Text language</td>
<td>Custom, Portuguese, English</td>
<td>Custom</td>
<td>English</td>
<td>English, Custom</td>
<td>English</td>
</tr>
<tr>
<td>Audio language</td>
<td>Custom, Portuguese, English</td>
<td>English</td>
<td>English</td>
<td>English, Custom</td>
<td>English</td>
</tr>
<tr>
<td>Administration options</td>
<td>None</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Customizable content</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Customizable appearance</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Table A.1: Comparison of applications for communication purposes
### A.2 Educational software

#### A.2.1 Studies regarding the use of software to improve language skills

<table>
<thead>
<tr>
<th>Author (Year)</th>
<th>Participants</th>
<th>Skills targeted</th>
<th>Software</th>
<th>Type of exercise</th>
<th>Design</th>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heimann et al (1995) [28]</td>
<td>30 children divided in 3 groups; 1 ASD group with 11 children</td>
<td>Reading, phonological awareness, sentence imitation, verbal behaviour</td>
<td>Alpha</td>
<td>Word identification, Sentence creation</td>
<td>n/a</td>
<td>All participants made progress with Alpha’s lessons</td>
</tr>
<tr>
<td>Moore and Calvert (2000)</td>
<td>14 children diagnosed with ASD (3-6 years old) split in two treatment groups (a behavioural based group and a computer based group)</td>
<td>Vocabulary acquisition</td>
<td>Specifically designed for the study</td>
<td>Labeling objects, responding to verbal commands about the objects</td>
<td>n/a</td>
<td>Participants recall more nouns and feel more motivated after being exposed to the computer</td>
</tr>
<tr>
<td>Massaro et al (2003) [10]</td>
<td>8 children (7-11 years old) diagnosed with ASD</td>
<td>Vocabulary acquisition</td>
<td>Language Player</td>
<td>Picture naming</td>
<td>Within Subjects Design</td>
<td>Participants have learnt new words and generalised them with different pictures and outside the Language Player</td>
</tr>
<tr>
<td>Hetzroni et al (2004) [29]</td>
<td>5 children (7.8 - 15.5 years old) with ASD; verbal level: less than 50% use of functional speech, delayed and immediate echolalia, irrelevant speech</td>
<td>Reducing echolalia and increasing communication initiations</td>
<td>I can word it too</td>
<td>Multiple choice questions</td>
<td>Multiple Baseline Design across three topics (food, hygiene and play)</td>
<td>All participants reduced the number of sentences with delayed echolalia across all topics; there was an increase of relevant speech in food and play topics; all participants preferred the computer interactive application</td>
</tr>
<tr>
<td>Coleman-Martin et al (2005) [14]</td>
<td>3 participants, only one of them was diagnosed with ASD (12 years old)</td>
<td>Vocabulary acquisition</td>
<td>MS Powerpoint</td>
<td>Phoneme and word vocalization</td>
<td>Multiple Conditions Design + Drop-down Baselines: Intervention in three phases: With a teacher only, with both a teacher and the computer application, and with the computer application only.</td>
<td>All participants acquired target words during each of the three phases</td>
</tr>
<tr>
<td>Massaro, and Bosseler (2006) [44]</td>
<td>5 children (8-13 years old) diagnosed with ASD</td>
<td>Vocabulary acquisition (learning rate)</td>
<td>Language Player</td>
<td>Picture naming</td>
<td>Within Subjects Design</td>
<td>Participants learnt at a faster rate with an animated character</td>
</tr>
<tr>
<td>Whalen et al (2009) [70]</td>
<td>2 groups of children diagnosed with ASD from four different schools: a control group with 25 participants and a treatment group with 22 participants, both from pre-school and kindergarten, ranging from mild to severe autism</td>
<td>Body part naming, receptive language (comprehension and vocabulary), expressive language (labeling and expressive communication), auditory memory, general concepts, social skills, matching</td>
<td>TeachTown: Basics</td>
<td>n/a</td>
<td>Between and Within Subjects Group Design</td>
<td>The treatment group shown improvements in every evaluated skills</td>
</tr>
<tr>
<td>Kagohara et al (2012) [34]</td>
<td>2 male adolescents (13 and 17 years old) diagnosed with ASD</td>
<td>Vocabulary acquisition</td>
<td>Proloquo2go</td>
<td>Picture naming using the software as a speech generation device to reply</td>
<td>Two studies using Multiple-probe Across Participants Design</td>
<td>Both participants were able to correctly name the displayed pictures after the intervention</td>
</tr>
</tbody>
</table>

Table A.2: Studies regarding the use of software for language skills development purposes
### A.2.2 Educational applications for the development of language skills

<table>
<thead>
<tr>
<th>Features</th>
<th>Autism iHelp</th>
<th>Camp Discovery</th>
<th>Lite Autism</th>
<th>Autism and PDD Lite</th>
<th>Vizzle Player</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>Free</td>
<td>Free</td>
<td>Free</td>
<td>Free</td>
<td>Free</td>
</tr>
<tr>
<td>Platforms</td>
<td>Mobile (Android, iOS)</td>
<td>Mobile (Android, iOS)</td>
<td>Mobile (Android, iOS)</td>
<td>Mobile (Android), Web, Mobile (Android, iOS)</td>
<td></td>
</tr>
<tr>
<td>Multimedia content</td>
<td>Audio, pictures</td>
<td>Audio, pictures</td>
<td>Audio, pictures</td>
<td>Text, audio, pictures</td>
<td></td>
</tr>
<tr>
<td>Interface language</td>
<td>English</td>
<td>English</td>
<td>English</td>
<td>English</td>
<td></td>
</tr>
<tr>
<td>Text language</td>
<td>English</td>
<td>English</td>
<td>n/a</td>
<td>English</td>
<td></td>
</tr>
<tr>
<td>Audio language</td>
<td>English</td>
<td>English</td>
<td>English</td>
<td>English</td>
<td></td>
</tr>
<tr>
<td>Administration options</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Customizable content</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Customizable appearance</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Interaction logs</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Multiple difficulty levels</td>
<td>No</td>
<td>Automatically adjusted</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Prompting</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Reinforcement</td>
<td>Yes</td>
<td>Yes (custom)</td>
<td>Yes (custom)</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Tutoring character</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>

Table A.3: Comparison of applications for the development of linguistic skills
ExerciseGenerator: Usage instructions

ExerciseGenerator was built using Maven\(^1\) so that the resulting jar file could be used as a dependency in any other Maven project. This can be achieved by adding the following snippet to the pom.xml file:

\[
\begin{verbatim}
<dependency>
  <groupId>ExerciseGeneration</groupId>
  <artifactId>ExerciseGeneration</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <type>jar</type>
  <scope>compile</scope>
</dependency>
\end{verbatim}

To use this module, it is also required to install TreeTagger and download the parameter files for Portuguese\(^2\).

---
\(^1\)Maven is a software project management tool available in [https://maven.apache.org/](https://maven.apache.org/) (last access in 05/10/2015).
\(^2\)Installation files and instructions, as well as the parameter files, are available in [http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/](http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/) (last access in 05/10/2015).
C.1 ImageNet-based resource schema

```xml
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="ImageNetStructure">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="synset" type="SynsetType" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="SynsetType">
    <xsd:sequence>
      <xsd:element name="synset" type="SynsetType" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
    <xsd:attribute name="id" type="xsd:string"/>
    <xsd:attribute name="words" type="xsd:string"/>
    <xsd:attribute name="gloss" type="xsd:string"/>
  </xsd:complexType>
</xsd:schema>
```
C.2 Usage instructions

*WordLookupImage* was built using Maven. Therefore, the resulting jar file can be added as a dependency in another Maven project by placing the following snippet in the `pom.xml` file:

```xml
<dependency>
  <groupId>pt.inesc.l2f</groupId>
  <artifactId>WordImageLookup4j</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <type>jar</type>
  <scope>compile</scope>
</dependency>
```

In order to use *WordLookupImage* in another project, it is also necessary to provide the following resources:

- The [XML](#) file containing the synset hierarchy should be placed under the `src/main/resources` directory of the project that is using *WordImageLookup*, so that it can be loaded as a Maven resource in runtime;

- The text files containing the images’ URLs should also be put under `src/main/resources`. Each file should contain one URL per line.

The file paths for the synsets’ file and the URL files’ directory should be specified as arguments of the class constructor of *WordLookupImage*. 
D.1 VITHEA-Kids: Evaluation with caregivers

D.1.1 Tasks

Figure D.1: Task guide for caregivers
D.1.2 Questionnaire

(a) Information about the caregivers

(b) Satisfaction regarding each task

(c) Satisfaction regarding overall experience

(d) New functionalities and suggestions

Figure D.2: Questionnaire to evaluate the caregiver's module
D.2 Automatic generation of exercises: Tasks and questionnaire

(a) Questionnaire instructions

(b) Classification of the word distractors’ quality

(c) Classification of the image distractors’ quality

(d) Suggestions

Figure D.3: Questionnaire to evaluate the automatic generation of exercises