Modeling Language and CASE Tool for Communication Board Customization

Natalia M. Franco[1,2], Robson N. Fidalgo[1], Edson A. Silva[1], Ticia F. Cacalcante[3], Patrick H. S. Brito[2]

[1] Informatics Center
Federal University of Pernambuco
Recife, Brazil
{nmf, rdnf, eas4}@cin.ufpe.br

[2] Computing Institute
Federal University of Alagoas
Maceió, Brazil
patrick@ic.ufal.br

[3] Educational Center
Federal University of Pernambuco
Recife, Brazil
ticiaferro@hotmail.com

***Abstract*— Communication Board (CB) is an Assistive Technology (AT) that enables people who have speech impairment to communicate their needs or desires using visual-graphic symbols. Since each patient has individual needs, it is important to offer a customizable approach to define the vocabulary of their CB. In this context, this paper proposes a modeling language for diagramming the CB vocabulary. Its feasibility, expressiveness and usefulness were validated through the construction of a software that will be described later in this paper.**

Keywords—Assistive Technology; Augmentative and Alternative Communication; Communication Board; Modeling Language; Speech Impairment.

I. INTRODUCTION

Communication, in its various forms, is a basic human need for people to socialize, express ideas, feelings and desires. The absence of this necessity causes frustration, anger and social exclusion of these individuals. Patak [1] conducted a study with patients who received mechanical ventilation during the period in intensive care unit and noted that the level of frustration in trying to communicate decreased with the insertion of an Augmentative and Alternative Communication (AAC) device based on Communication Board (CB). We know emotions influence the physical and psychological health of the patient. In this case, it was observed they were calmer when they could communicate effectively their complaints, sensations and feelings.

Patak's study [1] involved only adults with temporary speech impairment and, in the same case, we can cite those limitations caused by post-traumatic stress, which may present as a general aphonia or inability to verbalize some subjects [2,3]. In the case of a child who suffered aggression, bullying or sexual violence it is common, for fear or shame, the speech limitation and, in such cases, psychologists who work with this situations often use recreational resources as drawings and low-tech CB (e.g., colored cards and paper) so that the child can feel comfortable and can start dropping hints of what happened.

The situation becomes more critical when the speech impairment is permanent, caused by disability or injury. With children, the situation is even more delicate. According to Palerm & Ruiz [4], until 6 years old, children have their cognitive development directly related to language development. In this way, the lack of communication with other people, in this stage of life, can irreversibly damage their intellectual capacity. Besides that, in some variations of autism is possible to find characteristics as restricted repertoire of interests, deficits in social interaction and in communication [5]. In these cases, the CB may also be used as a mean to promote the socialization of these individuals.

Despite all the potential of CB, some solutions do not allow vocabulary customization for a given patient. Solutions that allow vocabulary customization do this by filling out the fields in a Graphical User Interface (GUI), where each element is separately constructed, without cohesion relationship with other elements. Thus, in this scenario, a patient using a CB can make all combinations with the CB elements, some of them causing cohesion problems in communication. For instance, is possible to communicate something like *"I want to wear an apple"*. This information is not consistent. The verb *"wear"* needs an outfit as a complement, and the noun *"apple"*, a verb related to food. Moreover, the vocabulary customization made from the fulfillment of fields in a GUI, makes hard the job of users (e.g., an audiologist) who for example, and by mistaken, can insert the same word more than one time.

Aiming to overcome the previous shortcoming, in this article, we propose a modeling language and a Computer Aided Software Engineering (CASE) tool that allow us to diagram the CB vocabulary as a mental/conceptual map. The proposed modeling language and the CASE tool are called CBML and CBCASE, repectively. CBML and CBCASE organize the CB vocabulary using "Element Categories" and "Cohesion Relationships", so that, Element Categories allow modularization and Cohesion Relationships define the successors of a vocable. In this case, the CBML and CBCASE may avoid the occurrence of cohesion problems, because by giving a Cohesion Relationship link to a Category (e.g., *eat*) this link will indicate a correct complement/successor (e.g.,

This research was supported by the National Center for Scientific and Technological Development (CNPq) and the Coordination of Improvement of Higher Education Personnel (CAPES).

food). In this way, using CBML and CBCASE it is possible to customize the CB vocabulary in several domains (e.g., basic needs, education, feelings). Moreover, using CBCASE is possible to generate a configuration file with the modeled CB content that can be used as an input to customize the CB on various platforms (e.g., desktop computers and mobile devices). These features allow the CB customization for different stakeholders and for various existing technologies. Furthermore, with a modeling language the user can work directly with domain concepts, abstracting the language syntax details, which facilitate learning and using of the language.

The motivations for this work are: 1) Improving the quality of life of speech impairment people; 2) Current solutions for AAC based on CB, when allow customization of content, are limited to the fields of their GUI; and 3) Lack of solutions that enable CB vocabulary customization in a diagrammatic view that allows categorization and succession links between its communication elements.

This paper is organized as follows: Section 2 covers the theoretical foundation of this work; Section 3 discusses the correlated works; Section 4 shows the proposal modeling language; Section 5 discusses a practical application with the proposed modeling language in our CASE tool; and Section 6 shows the conclusions and future works.

II. Theoretical Foundation

A. Assistive Technology for Augmentative and Alternative Communication

According to the Brazilian Committee of Technical Assistance [6], Assistive Technology is a field of knowledge with interdisciplinary feature that includes resources to promote the participation of people with disabilities in social life, seeking autonomy, independence and quality of life. For this reason, it is considered Assistive Technology any initiative that aims lowering barriers imposed by the disability to the individual, his inclusion in the society, autonomously, and to improve his quality of life. Some examples of Assistive Technologies are prosthetics and orthotics, equipments for the rehabilitation of people with disabilities, mobility elements such as cane, wheelchair, ramps, adapters environmental and safety bars, communication boards, etc.

In this context, the area of Assistive Technology dedicated to expand communication skills are called Augmentative and Alternative Communication (AAC). According to the American Speech-Language-Hearing Association (ASHA) [7], AAC is the area of clinical practice that attempts to compensate difficulties or disabilities demonstrated (either temporarily or permanently) by individuals with severe disorders of communicative expression. AAC resources recognize and value all communication attempts of an individual – whether through gestures, facial expression, eye gaze, writing or drawings – and try to complement them.

In AAC context, Alternative Communication Board (CB) are resources to assist people who have difficulties in the use of speech to communicate, because they use a set of symbols and captions to represent objects, actions, feelings, etc. So, the communication is established through the selection of symbols that express what the user wants to say. Generally, to facilitate the location of symbols, they are grouped by categories and colors. The most used pattern is the Fitzgerald Key [8], a system original developed in 1926 to teach deaf children to structure the sentences through a visual guide, associating a color to each category of words. The division is based on six categories as follows: 1) Nouns in orange; 2) Pronouns in yellow; 3) Verbs in green; 4) Adjectives in blue; 5) Social Expressions in pink; and 6) Miscellaneous in white.

B. Modeling Language and CASE Tool

Since a diagram transmits information more effectively than an equivalent text, modeling the CB vocabulary becomes more precise, expressive and direct than if it was done via a GUI. That is, a diagram specified in a modeling language transmits information more effectively to non-technical people than an equivalent specification in text [9]. Moreover, it has been observed that: 1) end users learn to use modeling language in a short time because they realize they are directly working with the domain concepts [10]; and 2) computational tools based on modeling language (CASE Tools) produce higher quality diagrams [11], since the errors are automatically checked by the modeling language. So that, modeling languages improves communication among stakeholders, by providing a diagrammatic and most representative way to document and validate the concepts of a given domain.

In this context, Computer-Aided Software Engineering (CASE) [12] refers to methods that use tools to automate the process of developing systems. CASE tools have emerged as one of the most important innovations to manage the complexity in systems development projects, especially the large and complex ones which involve many components and people. For example, developers can use CASE tools to draw the graphical interface of a system, and from these drawings automatically generate the application codes. Moreover, designers can use CASE tools to specify the system requirements which can later be automatically processed in the system documentation [11].

The most visible benefits of using a CASE tool are decrease development time, improve the quality of the developed system, reduce development costs, and facilitate communication among developers of the system process [13].

III. Related Works

In this Section, we present and evaluate the most popular and important works related to CB. In order to do a systematic evaluation of these works, we are using the following features: F1) Supporting computational solution to be used in technological devices; F2) Supporting mobile solution to be used with touch screen devices; F3) Supporting usability customization (e.g., scanning speed, type of touch and number of icons per screen); F4) Avoiding problems of textual cohesion (e.g., *"I want to drink a chocolate cake"*); and F5) Allowing the CB customization as a diagram.

Initiatives such as Boardmaker [14] and Aragonese Portal [15] allow the generation of customized CB. However, the customized CB generated by these initiatives can only be printed for its use on a paper. That is, these proposals do not generate a computational solution.

In computational solutions, the proposals of Saturno [16] and Gatti [17] allow the generation of customized CB. These CB are for desktop computers, which must be adjusted to an alternative input to the mouse and the keyboard. That is, the user can not make use of the features presented in mobile devices (e.g., portability, accelerometer and touch screen) because they are desktop solutions.

Initiatives such as Livox [18], Grid Player [19], Que-Fala! [20] and Vox4All [21] generate solutions for mobile devices. In the particular case of Livox, its customization is limited to preconfigured modules by the development team, which does not give customization freedom for its users. Differently, Que-Fala!, Grid Player and Vox4All, even allowing a better CB customization, they are limited to the resources available by the GUI use.

Table I shows a comparison among correlated works and our purpose. The "+" indicates that the presented solution supports that feature and a "-", that it does not support the feature.

Eight relevant proposals were investigated and we have identified that two are no computational solutions, two are desktop computational solutions and four are mobile computational solutions. The presented solutions, when allow customization of content, do that through a GUI. Among the disadvantages of customization through a GUI, it is possible to highlight the limitations concerning the fields that must be filled and the impossibility of establishing cohesion relationships between words and symbols. The lack of cohesion of these relationships allows the combination of all symbols entered and it may cause cohesion problems related to gender, verbs/complements and nouns/adjectives (see Section V). We highlight that it is a problem in all computational solutions here presented.

IV. A Language to Customize CB Vocabulary

A. Overview

Fig. 1 shows an overview of our proposal for CB vocabulary customization using CBML and CBCASE. The proposal consists of two components: 1) a CASE Tool for modeling the CB vocabulary; and 2) the CB Generator to produce a touch screen device application (*app*). These two modules communicate themselves through an Extensible Markup Language (XML) [22] file, which is generated from the diagram modeled in the CASE tool. In other words, this XML file contain all the modeled vocabulary using CBML and CBCASE. This XML file serves as an input to the CB Generator customize the CB for a touch screen device. In this context, the contribution of this paper are described as following: 1) the definition of a modeling language (CBML) to diagram the CB vocabulary; and 2) the construction of a CASE tool (CBCASE) which is based on this modeling language. So, in this paper, the left side of Fig. 1 shows the focus and contribution presented in this manuscript.

TABLE I. RELATED WORKS COMPARATIVE

Purposes	**Features**				
	F1	***F2***	***F3***	***F4***	***F5***
Gatti's work	+	-	+	+	-
Saturno's work	+	-	+	+	-
Boardmaker	-	-	+	-	-
Que-Fala!	+	+	+	-	-
Portal Aragonês	-	-	+	-	-
Livox	+	+	+	-	-
Grid Player	+	+	-	-	-
Vox4All	+	+	+	-	-
Our purpose	+	+	+	+	+

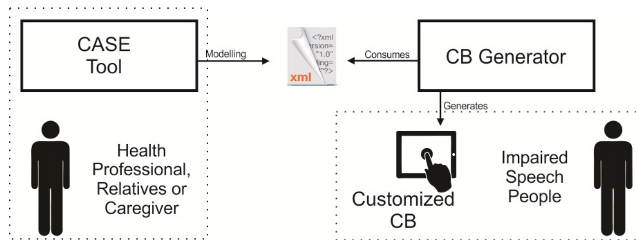

Fig. 1. Overview of the proposed solution

In Fig. 1, the CASE tool is based on a desktop computer and its users are health professionals, relatives or caregivers. The CB Generator of the same figure is based on a touch screen device (e.g., tablets and smartphones) and its users are impaired speech people.

Is important to mention CB customization (e.g., contrast color, size and number of icons per screen) and its usability requirements (e.g., sonorous feedback and scanning mode) are the responsibility of the CB Generator and the modeling language only defines the CB vocabulary.

B. Language Notation

Fig. 2 shows the CBML elements, which are based on Fitzgerald Key notation [8]. Such notation was chosen because it is a widely used pattern for professionals (e.g., audiologist, occupational therapist) in CB solutions. In this work, the white color is also used to represent the concept of “Context”.

The notational elements of CBML are: 1) “Categories”, represented by a white rectangle with a pictogram that is a folder (Fig. 3) with the initial letter of the category name and the color that follows the Fitzgerald Key pattern; 2) “Elements”, represented by a white rectangle with a pictogram that is a file (Fig. 3) with the initial letter of the category name and the color that follows the Fitzgerald Key pattern; 3) “Context”, represented by an white ellipse with a pictogram that is a circle with the letter “C”; 4) “Succession Link”, represented by a continuous line; and 5) “Contextualization Link”, represented by a dashed line. In relation to the pictograms (Fig. 3), a category is represented by a folder that can contain other folders (subcategories) and various elements, which are represented by files.

V. A CASE TOOL TO CUSTOMIZE CB VOCABULARY

So as to demonstrate the feasibility, expressiveness and usefulness of our language, we have developed the CBCASE. In Fig. 4 the GUI of CBCASE with a fragment of a hypothetical CB vocabulary is modeled. It is important to point out that our goal is to present a didactic and expressive example that covers all the constructors of the CBML. The GUI of CBCASE is divided into the three main areas: 1) Drawing Area (area "A" in Fig. 4) – space for modeling the CB vocabulary; 2) Element Palette (area "B" in Fig. 4) – contain all the notational elements of CBML; and 3) Options Area (area "C" in Fig. 4) – divided into three tabs (properties, overview and CB export). For each element, in the properties area, caption information, audio and image can be inserted. In the overview area, it is possible to see all the modeling vocabulary. In the CB export is possible to see the XML code responsible to customize the CB content.

CBCASE has a pallet (area "B" in Fig. 4) with the CBML elements. The modeling of a CB vocabulary starts with a click on the desired notational CBML element followed by other click in the drawing area (area "A" in Fig. 4). Next, the user can edit the properties (e.g., symbol, caption, audio) of the notational CBML element (one of the tabs in area "C", Fig. 4). Four well-formed rules are guaranteed by the semantic of the CBML. These rules are: 1) A subcategory can only be built within another category of the same type (e.g., nouns, pronouns, and miscellaneous); 2) An element can only be instantiated within a category of the same type (e.g., nouns, pronouns, miscellaneous); 3) A "Succession Link" can only exist between categories of "Verb" and "Adjective" and another category of a different type (e.g., nouns, pronouns, social expression, miscellaneous); and 4) A "Contextualization link" can only exist between "Context" and a "Category" different of "Verb" and "Adjective". Once finished the modeling, the user can save the project or automatically generate the XML file with the customized vocabulary.

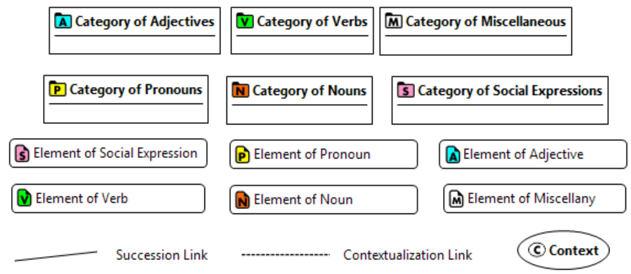

Fig. 2. Language elements of the CBML

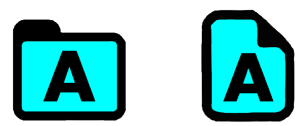

Fig. 3. Category and element pictogram for adjectives

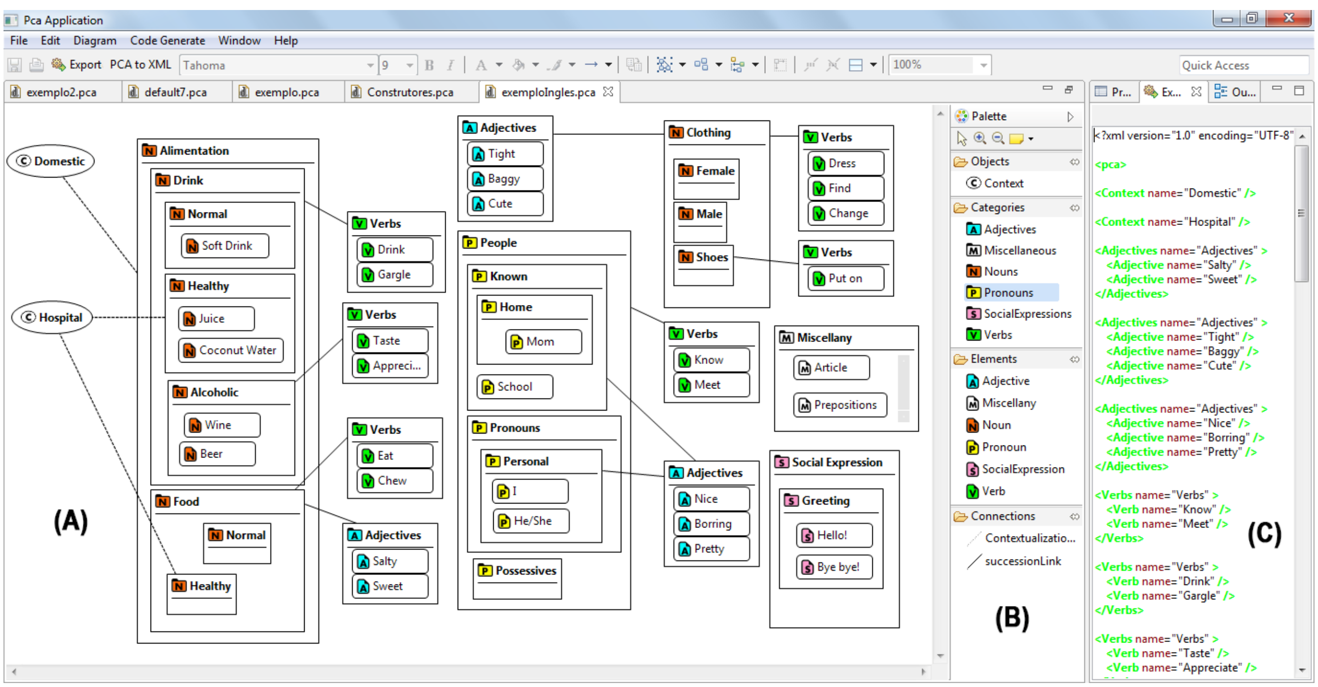

Fig. 4. GUI of our CASE tool

The hypothetical vocabulary showed in Fig. 4 covers concepts about “Alimentation” and “Clothing”. Through this example is possible to understand the CBML concepts and how to organize and connect its notational elements. In this case, “Alimentation” was categorized into “Drink” and “Food” and later in “Normal”, “Healthy” and “Alcoholic”, for drinks. The health professionals can choose the way to categorize the vocabulary but is important to know that each new subcategory implies in one more click until the communication is established.

Continuous lines (Succession Link) connect a list of “Verbs” or “Adjectives” to some “Category” which are responsible to solving cohesion problems. In Fig. 4, if the user selects the verb “*eat*” the CB will filter the complements, leaving visible only icons that represent food. Thus, it is not possible to construct phrases like *“I want to eat an orange juice”* or, in the case of the verb “drink”, *“I want to drink a chocolate cake”*. Relative to adjectives, we can cite the case of the word “*salty*” that can be used in the food context (*“This meat is salty”*) but it is not applied with clothing (*“That hat is salty”*). It is worth to emphasize that the continuous lines are bidirectional and can filter both on the category side as well the verb/adjective side.

The dashed lines connect “Context” and “Category”. In this case (Fig. 4), there are two contexts, “*Domestic*” and “*Hospital*”, which are respectively connected to “*Alimentation*” and “*Healthy*”. This means that in a “*Hospital*” context the user has access only to food in a balanced diet and in a “*Domestic*” context, to all food categorized in “*Alimentation*”. Another examples of context are: 1) filter the known people, by the place (e.g., home, school, hospital); and 2) filter the words by their topics in school (e.g., animals, fruits, places). The context can be manually selected or can be automatically modified by a geolocation device. The context can work in two ways: 1) restricting the other options that are not covered by the current context; or 2) giving priority to those icons that are covered by the context and the remaining icons stay visible.

VI. CONCLUSION

In this work, we presented the concept of Assistive Technology as a way to promote the inclusion of people with disabilities in a society and, more specifically, the Augmentative and Alternative Communication (AAC), focused on people unable to use speech to communicate.

The weaknesses of related works converge to the problem of vocabulary customization. When it is possible, it is made by filling the fields in a GUI. In this kind of customization each element of the vocabulary is separately constructed, without cohesion relationship with others elements. That is, in this scenario, the user can make all combinations of elements, some of them generating cohesion problems in communication (e.g., *"I want to wear an apple"*). Moreover, the vocabulary customization through GUI does not allow the modularized view of the CB vocabulary, which can complicate the user's work.

To give a contribution to solve this problem, we have proposed a new paradigm for vocabulary customization based on two components: 1) a CASE Tool for modeling the CB vocabulary; and 2) the CB Generator for creating an application (*app*) for a touch screen device. In this paper we focused on the first component and we have presented the modeling language (CBML) and the CASE tool (CBCASE).

Our proposal is original and provides a new paradigm for the CB vocabulary customization, because our paradigm is based on a modeling language that allows the content modeling with features like creating categories and build relationships that limit the constructions with cohesion problems. A modeling language can be defined to have expressiveness to allow CB customization in various contexts, taking into account different stakeholders in their use. Furthermore, by using a modeling language, it is possible to work directly with domain concepts, thus facilitating its learn and use. However, even with this facility, the use of a CASE tool is not simple and, as any new technology, needs to be taught to users (e.g., health professionals, relatives or caregivers). On the other hand, this customization problem could be solved by using an ontology editor (e.g., Protegé) but it is not domain-specific, thus makes harder the modeling work.

Our modeling language uses concepts well accepted by this research public, as the Fitzgerald Key and the categorization of contents in folders and files (here called Category and Element). To demonstrate the feasibility, expressiveness and usefulness of our modeling language, the evaluation of CBML was made through the implementation of our CASE tool.

In summary, with this work, we intended to create a new paradigm to customize CB content through a configuration file. But, the modeling language is only a piece of our solution. To become a viable technology, more efforts in developing our CB Generator are needed. As future work, we intend to deepen the studies in the development of CB and in the integration of these two components (CASE tool and CB Generator).

REFERENCES

[1] Patak, L., Gawlinski, A., Fung, N. I., Doering, L., Berg, J., & Henneman, E. A. (2006). Communication boards in critical care: patients' views. Applied Nursing Research, 19(4), 182–190. doi:http://dx.doi.org/10.1016/j.apnr.2005.09.006

[2] Jung, C. R. (1999). O Estresse e a Voz. CEFAC - Centro de Especialização em Fonoaudiologia Clínica.

[3] Mello, L., & Caramaschi, S. (2010). Intercorrências no desenvolvimento infantil. (Editora UNESP, Ed.) (p. 18). São Paulo: SciELO Books.

[4] Palerm, C. C. R., & Ruiz, P. A. L. (1992). Augmentative / Alternative System for Public Communication for Children with Cerebral Palsy. In 14th Annual International Conference of Engineering in Medicine and Biology Society of the IEEE (pp. 1525–1526). Paris.

[5] Gadia, C. A., Tuchman, R., & Rotta, N. T. (2004). Autismo e doenças invasivas de desenvolvimento. Jornal de Psiquiatria, 83–94.

[6] Secretaria Especial dos Direitos Humanos. Comitê de Ajudas Técnicas. (2009). Tecnologia Assistiva (p. 138). Brasília: CORDE.

[7] ASHA. (2013). American Speech-Language-Hearing Association. Retrieved October 30, 2013, from http://www.asha.org/

[8] Fitzgerald, E. (1949). Straight Language for the Deaf. A System of Instruction for Deaf Children. (p. 104). Alexander Graham Bell Association for the Deaf.

[9] Moody, D. L. (2009). The “Physics” of Notations: Towards a Scientific Basis for Constructing Visual Notations in Software Engineering, 35(5), 756–778.

[10] Hudak, P. (1996). Building domain-specific embedded languages. ACM Comput. Surv., 28(4es). doi:10.1145/242224.242477

[11] Kelly, S., & Tolvanen, J. P. (2008). Domain-Specific Modeling: Enabling Full Code Generation. Wiley. Retrieved from http://books.google.com.br/books?id=GFFtRFkuU_AC

[12] Andoh-baidoo, F. K., Kunene, K. N., & Walker, R. E. (2009). An Evaluation of CASE Tools as Pedagogical Aids in Software Development Courses, 369–375.

[13] Raphael, B., & Smith, I. F. C. (2013). Engineering Informatics: Fundamentals of Computer-Aided Engineering (Second Edi., p. 267).

[14] Mayer-Johnson. (2013). BoardMaker Software. Retrieved July 18, 2013, from http://www.mayer-johnson.com/boardmaker-software/

[15] Governo de Aragão. (2013). ARASAAC - Portal Aragonês de Comunicação Alternativa e Ampliada. Retrieved from http://www.catedu.es/arasaac/index.php

[16] Saturno, C. E. (2013). Protótipo de ferramenta de comunicação aumentativa e alternativa para crianças e adolescentes com paralisia cerebral. Universidade do Vale do Itajaí.

[17] Gatti, N., Matteucci, M., & Sbattella, L. (2004). An Adaptive and Predictive Environment to Support Augmentative and Alternative Communication. In J. Klaus, K. Miesenberger, W. L. Zagler, & D. Burger (Eds.), ICCHP (Vol. 3118, pp. 983–990). Springer. Retrieved from http://dblp.uni-trier.de/db/conf/icchp/icchp2004.html#GattiMS04

[18] Livox. (2013). Liberdade em Voz Alta. Retrieved July 17, 2013, from http://www.agoraeuconsigo.org/

[19] Sensory Software International. (2013). Grid Player. Retrieved July 18, 2013, from http://www.sensorysoftware.com/gridplayer.html

[20] Que-Fala. (2013). Que Fala! Retrieved July 18, 2013, from http://metodos-si.com.br/

[21] Imagina. (2013). Vox4all. Retrieved October 28, 2013, from http://www.imagina.pt/produtos/educacao-especial/vox4all/

[22] W3C. (2014). Extensible Markup Language (XML). Retrieved April 09, 2014, from http://www.w3.org/XML/