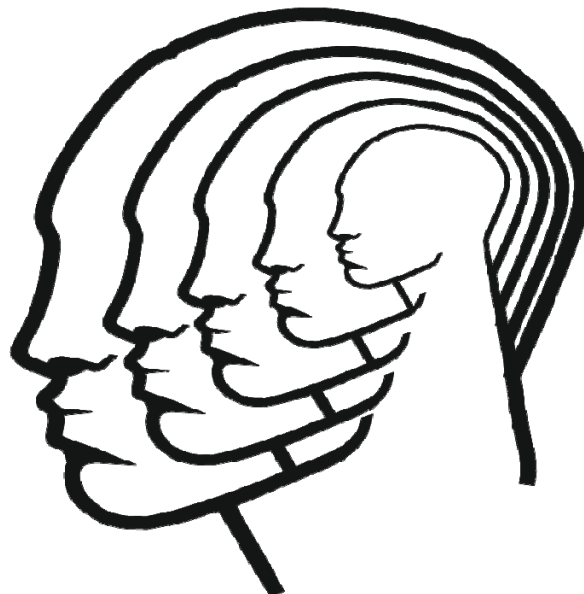


Departamento de Ingeniería de Sistemas Informáticos y
Telemáticos

TESIS DOCTORAL

DESARROLLO DE UNA NUEVA METODOLOGÍA PARA
LA EVALUACIÓN DE SOFTWARE ERGONÓMICO EN
EDUCACIÓN PRIMARIA



Autora: Elsa Chinita Soares Rodrigues

Director: José Moreno del Pozo

D. José Moreno del Pozo, Profesor Titular del Departamento de Ingeniería de Sistemas Informáticos y Telemáticos de la Universidad de Extremadura.

CERTIFICA

Que D^a Elsa Chinita Soares Rodrigues, ha realizado en el Departamento de Ingeniería de Sistemas Informáticos y Telemáticos, bajo mi dirección, el trabajo de investigación correspondiente a su Tesis Doctoral titulada:

“Desarrollo De Una Nueva Metodología Para La Evaluación De Software Ergonómico En Educación Primaria”

Revisado el presente trabajo, estimo que puede ser presentado al tribunal que ha de juzgarlo y autorizo la presentación de esta Tesis Doctoral en la Universidad de Extremadura.

Cáceres, a 4 de Julio de 2012

Fdo. D. José Moreno del Pozo
Titular de Universidad Departamento de Ingeniería
de Sistemas Informáticos y Telemáticos
Universidad de Extremadura

Esta tese está escrita com o novo acordo ortográfico da língua portuguesa.

Agradecimentos

Gostaria de expressar meu agradecimento ao meu orientador José Moreno del Pozo cuja competência, compreensão, amizade, e mais importante a imensurável paciência que teve ao longo destes anos. Mais acrescento que contribuiu para o meu conhecimento. Direi mesmo que tornou-se um amigo incontestável. O meu muito obrigado.

Agradeço ao aluno Daniel Grilo à designer Cristina Rodrigues pelo empenho, dedicação e amizade, sem eles o trajeto teria sido longo. Às professoras Patrocínio, Carmo, Dores, Otília, Antónia pela dedicação e empenho demonstrado e a todos os alunos que participaram neste estudo.

Uma palavra especial de agradecimento vai para Isabel Sofia Brito e para Fernanda Pedro pelas discussões úteis sobre versões anteriores desta Tese de doutorado e, mais importante, pela amizade demonstrada ao longo de tantos anos.

Tenho de agradecer a António Toucinho da Silva, e ao Luís Bruno pelas suas contribuições e comentários.

Agradecimento muito especial à revisora desta Tese Madalena Rodrigues.

Agradeço aos meus colegas e amigos, João Paulo Barros, Luís Garcia e João Paulo Trindade.

Finalmente, e mais importante, quero agradecer à minha família, André, Eduardo, Zé Maria, porque eles são "únicos, maravilhosos e incontestavelmente a minha razão de existir".

Agradeço à restante família e amigos pelo seu apoio, carinho e amizade, à minha mãe, aos meus irmãos, à Guida, Zeca, Ana, Francisco, Kiko e Maria, Ni, João Inês e António.

Abstract

The primary aim of this Project was to support elementary school teachers in the educational process (teaching-learning). To achieve such a goal, it was necessary to study and to analyze some quality and pedagogical software development methodologies, focused on the final product. However, to reach and stay focused in the material needed for the Project we kept in mind both *User Interface Designer* and *Human Computer Interface* models. The Study was focused in a specific target related to teaching and after the analysis of the requirements it was necessary to divide it in two groups: on the one hand we have students and on the other hand teachers.

The referred analysis enabled the awareness of the specific requirements of each group as well as the common features among them. Thus, it was possible to define the following objectives: to systemize a set of activities related to the “ergonomic analysis of the tasks” process; to develop an Interactive and Adaptable model in the process of teaching-learning; to develop Adjustable Interactive Oriented Method; to adapt the referred model to any user; to develop and evolve the capacity of the student at the same time that more complex concepts are introduced; and to prove the validity of this model in the development of the process teaching-learning.

For that, it was necessary to define a model that involved both the already referred analytical results and the proposed objectives. There was also the need to study methodologies contributing to the development of educational software, and to relate them with each other. Beyond these methodologies, it was convenient to study interface and cognitive models, in order to connect them with those methodologies.

We have yet to stress that after the study of the methodologies and evaluation models, we only used the specific parts of each of them we considered to add any value to the Project. Thus, we considered the created model a reflex of a good management between the quality of the products and the quality of the developing process.

We can look at the model as to a triangle, each *vertex* representing a variable important to consider: *vertex* IHC – Human Interface Computer (includes User Design Interface as well); *vertex* QUAL – Quality of the software (the analyzed methodologies adapted to our study being part of it); and *vertex* ERG – Ergonomic Models (analyzed and adapted to the study). It is important to point that each *vertex* has its own *sub-variables* and that there is interaction among them. All three *vertices* should be harmonically related and only after achieving this is our goal completed.

Our model is based on the Adjustable Interactive Model for Education (IAOE). In order to prove the viability of both the model and the method, a software was developed, related to the learning of the Portuguese language in the elementary school.

This “IAOE” method is developed following determined steps, being them sequential or not, enabling us to incorporate all the analytical and developmental phases of software in an interactive form, allowing the ideas and the processes to be verified and evaluated constantly.

The developed software suffered a great number of adjustments and improvements from the beginning as first users made their suggestions. After that it was tested by teachers and students over a twelve month period in three elementary schools from Beja region. During and after this period many more changes (in colors, interactions, messages, etc.) were made as a response to users’ requests and recommendations. After that and aiming to test its quality, usability and pedagogical interactive aspects were analyzed. Associations between image, text, sound, text *versus* sound and image *versus* movement and sound were also examined. After all the alterations and tests had been done, the software was again introduced in the school in order to be evaluated in a test.

The use of this software by students that are learning how to read will help them in an interactive way through the discovery of structures of both words and sentences. It shows them the word as a whole and through associations with images and words, making it easier to remember for posterior syllabic division and help in forming other words.

Teachers who use this software can systemize contents and meet students' difficulties, being able to change words and sentences as necessary for the students. Furthermore, the possibility to include tests and to graphically register the results for each student, make it a useful work instrument for management and evaluation of students' learning process.

Resumen

Esta tesis Doctoral está focalizada en la asistencia a los profesores del primer ciclo básico en el proceso enseñanza-aprendizaje. Por tanto, ha sido necesario estudiar y analizar diversas metodologías de desarrollo de software educativo, volcadas al producto final, de gran calidad y pedagógicas. Adicionalmente, para poder interconectar estas metodologías y extraer de ellas lo más importante para nuestro estudio, se han tenido en cuenta dos paradigmas derivados del mundo de las TIC's (Tecnologías Informáticas y de Comunicaciones), "User Interface Designer" y "Human Interface Computer". Nuestro estudio ha considerado un segmento de personas muy específico y relacionado con el ambiente enseñanza-aprendizaje y hemos optado por dividirlo en dos grupos. Uno de ellos el de los alumnos (discentes, receptores del mecanismo), el otro, profesores (docentes, emisores). Esta división surgió como una necesidad y como resultado lógico al efectuar el análisis de los requisitos.

Este análisis proporcionó la distinción de las necesidades específicas de cada uno de los grupos en el proceso enseñanza-aprendizaje y lo que tienen en común. A partir de este, fue posible determinar los siguientes objetivos generales:

- Sistematizar un conjunto de actividades relacionadas con el proceso de análisis ergonómico de las tareas en el ámbito educativo;
- Desarrollar un modelo interactivo adaptable al proceso enseñanza-aprendizaje ;
- Adaptar el referido modelo al utilizador, ya sea docente o discente;
- Desarrollar la capacidad evolutiva una vez que el alumno adquiera los conceptos más complejos;
- Comprobar la validez de este modelo en el desarrollo del proceso enseñanza-aprendizaje.

Por lo tanto, fue necesario definir un modelo que tuviera en cuenta los resultados del análisis antes explicado y los objetivos establecidos. Apareció, por tanto, la necesidad de estudiar metodologías que contribuyen en el desarrollo del software educacional, y relacionarlas entre sí mismas. Más allá de estas metodologías, ha sido conveniente estudiar

modelos de interface de usuario y modelos cognitivos, con la idea de interconectarlos con las metodologías indicadas.

Al final de todo el estudio de esta panóplia de metodologías y modelos de evaluación, tuvimos que integrarlas y extraer lo que consideramos la plusvalía de cada metodología, para crear nuestro propio modelo. Este, es reflejo de una buena gestión entre la calidad del producto y la calidad del proceso de desarrollo.

La simbología del modelo es representada por un triángulo, con los tres vértices simbolizando las tres variables que deben encontrarse en equilibrio perfecto : el vértice HIC – Human Interface Computer – que engloba también User Interface Design) , el vértice CUAL – “Calidad del Software” - que considera las metodologías analizadas y adaptadas a nuestro estudio y el vértice ERG – “Ergonomía” que representa los modelos ergonómicos analizados y adaptados a nuestro estudio en perfecta armonía. Sólo de esta forma, serán alcanzados los objetivos propuestos. No obstante, es importante subrayar que cada uno de estos vértices es compuesto por variables de segundo grado que a su vez se van relacionar con las variables de segundo grado de los demás vértices. Por lo tanto, existe un cruce entre todas las variables e variables de segundo grado, componentes de los vértices.

El modelo propuesto, tiene en su base al Método Interactivo Adaptable Orientado a la Enseñanza (en portugués MIAOE), diseñado y desarrollado en esta tesis. Para probar la viabilidad del modelo y del método fue creado y desarrollado un software específico, relacionado con el aprendizaje de la lengua portuguesa para el primer ciclo de enseñanza.

El método MIAOE es desarrollado teniendo en cuenta diferentes etapas, a veces cíclicas, a veces secuenciales, lo que nos permitió incorporar las diferentes fases de análisis y desarrollo de un determinado software de una forma interactiva , possibilitando que las ideas y los procesos fueran verificados y evaluados de forma continua.

El software desarrollado ha sido testado en tres escuelas de enseñanza básica de la provincia de Beja, durante un año. En este período han sido realizadas muchas alteraciones y ajustes, a petición de los usuarios (docentes y discentes). Las alteraciones y ajustes se focalizaron en diferentes puntos (los colores, las interacciones, los mensajes, etc.). Con el objetivo de testar la calidad del software, fueron analizados de forma recurrente diversos aspectos relacionados con: la interactividad, la usabilidad y facilidad de utilización por parte

del usuario y los aspectos pedagógicos. Después, fueron analizadas en profundidad las asociaciones entre la imagen, el texto, el sonido, texto frente a sonido, e imagen frente a movimiento y sonido.

Después de realizar los ajustes y alteraciones, el software fue nuevamente evaluado en las escuelas, iniciándose una segunda serie de pruebas, según los mismos parámetros de calidad; como resultado de este proceso una de las fichas de evaluación tuvo que ser reformulada.

La utilización de este software, por parte de los alumnos que se encuentran en fase de iniciación del aprendizaje de la lectura, les facilita un abordaje a la lengua escrita de forma muy interactiva, por vía de el descubrimiento de las palabras y las estructuras de las frases. El software les permite visualizar la palabra en su totalidad utilizando la asociación entre la palabra y la imagen y una progresiva memorización, para que posteriormente puedan ejecutar una división silábica y la construcción de otras palabras.

En resumen, a los profesores que lo utilizan, nuestro modelo y por ende nuestro software orientado al proceso de enseñanza-aprendizaje, les permite sistematizar los contenidos y disminuir las dificultades de los alumnos, en la medida que les es posible alterar palabras o frases, según las dificultades de cada alumno. Además de la posibilidad de incluir fichas, el registro de los resultados de cada uno de los alumnos, en forma gráfica, constituye un instrumento de trabajo muy útil y fácil a la hora de evaluar y gestionar el aprendizaje.

Sumário

O projeto em causa visou apoiar os professores do 1º ciclo básico no processo ensino aprendizagem. Para tal, foi necessário estudar e analisar algumas metodologias de desenvolvimento de *software*, centradas no produto final, de qualidade e pedagógicas. Contudo, para podermos interligar estas metodologias e retirar delas o mais relevante para o nosso estudo, tivemos de ter em conta os modelos de *User Interface Design* e *Human Interface Computer*. Porém, como o nosso estudo abrangeu um público específico e relacionado com o ensino, optámos por dividi-lo em 2 grupos. Sendo um deles o dos alunos e o outro dos professores. Esta divisão foi necessária, após termos efetuado a análise de requisitos.

Esta análise proporcionou distinguir as necessidades específicas de cada um dos grupos e o que tinham em comum. Assim, foi possível definir os seguintes objetivos: sistematizar um conjunto de atividades relacionadas com o processo de análise ergonómica das tarefas; desenvolver um modelo Interativo Adaptável no processo ensino aprendizagem; adaptar o referido modelo ao utilizador qualquer que seja ele; desenvolver a capacidade evolutiva à medida que o aluno aprenda conceitos mais complexos; provar a validade desse modelo no desenvolvimento do processo ensino-aprendizagem.

Para tal, foi necessário definir um modelo que englobasse os resultados da análise referida anteriormente e os objetivos referidos. Houve ainda a necessidade de estudar metodologias que contribuíssem para o desenvolvimento de *software* educacional, e relacioná-las umas com as outras. Para além destas metodologias, houve a conveniência de estudar modelos de interface e modelos cognitivos, de forma a interligá-los com as metodologias já referidas.

Após todo o estudo desta panóplia de metodologias e modelos de avaliação, tivemos que integrá-las, e extrair o que consideramos uma mais-valia em cada uma delas, para criarmos o nosso modelo. O modelo em causa é reflexo de uma boa gestão entre a qualidade do produto e a qualidade do processo de desenvolvimento.

A simbologia do modelo é um triângulo, os seus três vértices que simbolizam as três variáveis que devem estar em equilíbrio (o vértice IHC – “*Human Interface Computer*” e engloba também *User Interface Design*”; o vértice QUAL – “*Qualidade de software*” engloba as metodologias analisadas e adaptadas ao nosso estudo; o vértice ERG – representa, os modelos ergonómicos analisados e adaptados ao estudo em causa) em perfeita harmonia. Só dessa forma é que o nosso objetivo é alcançado. É de salientar, que cada um destes vértices é composto por subvariáveis que têm de se inter-relacionar com as subvariáveis dos vértices restantes. Como tal, existe um cruzamento entre todas as variáveis e subvariáveis.

O modelo em causa tem por base o Método Interativo Adaptável Orientado para o Ensino (MIAOE). Para provar a viabilidade do modelo e do método foi desenvolvido um *software*, relacionado com a aprendizagem da língua portuguesa para o 1º ciclo básico.

Este método MIAOE é desenvolvido seguindo determinados passos, sequenciais ou não. Permite-nos incorporar todas as fases de análise e desenvolvimento de um *software* de uma forma interativa, possibilitando que as ideias e os processos sejam verificados e avaliados constantemente.

O *software* desenvolvido, foi testado em três escolas do ensino básico do concelho de Beja, durante um ano. Durante este tempo foram feitas muitas alterações a pedido dos utilizadores. Essas alterações abrangeram diferentes pontos (cores, interação, mensagens, etc.) Com vista a testar a qualidade foram analisados os aspetos interativos, de usabilidade e pedagógicos. Logo foi analisado em detalhe as associações entre a imagem, texto, som, texto versus som, imagem versus movimento e som.

Após estas alterações o *software* voltou às escolas para ser novamente avaliado ao mesmo nível de qualidade, nesta altura uma das fichas de avaliação teve de ser reformulada. Pois só assim era possível obter resultados coerentes de acordo com os objetivos estabelecidos.

A utilização deste *software* pelos alunos que se encontrem em fase de iniciação da aprendizagem da leitura facilita-lhes uma abordagem à linguagem escrita de forma interativa através da descoberta da palavra e estrutura da frase. O *software* permite-lhes a visualização da palavra global e através de associações de imagens e palavras, uma progressiva memorização para posterior divisão silábica e construção de outras palavras.

Os professores que utilizarem este *software* poderão sistematizar conteúdos e suprir dificuldades do aluno, na medida em que poderão alterar palavras e frases conforme for necessário aos seus alunos. Para além disto o facto de ser possível a inclusão de fichas, o registo dos resultados de cada um dos alunos em gráfico constitui um instrumento de trabalho facilitador da avaliação e gestão das aprendizagens dos alunos.

Acrónimos

Esta seção apresenta a lista de acrónimos utilizada nesta Tese.

Acrónimos	Significado
ASD	Adaptive Software Development
CRC	Class-Responsibility-Collaborator
DFD	Diagrama de Fluxo de Dados
HCI	Human Computer Interface
HCI Concerns	Human Computer Interface Concerns
HCI Designers	Human Computer Interface Designers
IEEE	Association, IEEE Standards
I.S.I.D	Systems International Design
ISO	International Organization for Standardization
L.B.S.E	Lei de Bases do Ensino Educativo
Modelo IA	Modelo Interativo Adaptável
Modelo IAOE	Modelo Interativo Adaptável Orientado para o Ensino
MIAOE	Método Interativo Adaptável Orientado para o Ensino
MOLIC	A Modeling Language for Interaction as Conversation
TIC	Tecnologias de Informação e Comunicação
UID	User Interface Design
KIS	Keep it simple
XP	Extreme Programming

Índice

Capítulo 1.....	1
Introdução	1
1.1 Contextualização	3
1.2 Objetivos da Tese	7
1.3 Validação dos Resultados.....	8
1.4 O contributo desta Tese	8
1.5 Estrutura da Tese	12
Capítulo 2.....	17
Estado da Arte	17
2.1 O Processo Ensino Aprendizagem	18
2.2 Princípios para a Avaliação do Software Educacional/Educativo	29
2.2.1 Metodologia de Avaliação Pedagógica de Reeves.....	32
2.2.2 Metodologia de Avaliação Pedagógica de Bloom.....	38
2.2.3 Taxonomia Solo.....	46
2.2.4 Metodologias de Avaliação Pedagógica do Software Educativo	47
2.3 Metodologias de Avaliação da Qualidade de Software.....	53
2.3.1 Descrição dos Modelos McCall e Boehm.....	54

2.3.2	Modelos de Qualidade do Software Ergonómico, Princípios, Regras e Critérios.....	59
2.3.3	Normas Ergonómicas.....	70
2.4	Metodologias de Desenvolvimento de Software.....	75
2.4.1	Modelo em Cascata/Queda de Água/WaterFall	78
2.4.2	Modelo de Prototipagem.....	78
2.4.3	Modelo em V (Quality Assurance e Quality Control).....	79
2.4.4	O Modelo Mercado e o Modelo Híbrido.....	81
2.4.5	Processo de Desenvolvimento User Interface Design (UID).....	82
2.4.6	Modelos de Desenvolvimento Evolutivos.....	84
2.4.7	Metodologias de Desenvolvimento Ágil.....	88
	Conclusões.....	93
	Capítulo 3.....	95
	Proposta do Novo Modelo	95
3.1	Metodologias Adoptadas	96
3.1.1	Teorias de Aprendizagem.....	96
3.1.2	Metodologias de Avaliação de Software Educativo.....	98
3.1.3	Modelos de Qualidade.....	101
3.1.4	Metodologias de Desenvolvimento de Software.....	103
3.2	O Novo Modelo Interativo Adaptável	110

3.2.1	Descrição do Método Interativo Adaptável Orientado para o Ensino (MIAOE).....	113
	Conclusões.....	119
Capítulo 4.....		121
Análise Sistemático do Contexto.....		121
4.1	Análise de Requisitos.....	124
4.1.1	Requisitos funcionais do Software "Palavras e Palavras".....	129
4.1.2	Requisitos Não Funcionais do Software "Palavras e Palavras".....	130
4.1.3	Diagramas de Fluxo de dados.....	131
4.1.4	Descrição dos processos, dos depósitos de dados e dos fluxos de dados do Software "Palavras e Palavras".....	138
4.1.5	Especificação de Processos do Software "Palavras e Palavras" através de Fluxogramas.....	141
4.2	Desenvolvimento do Software "Palavras e Palavras".....	158
4.2.1	Equilíbrio Visual.....	159
4.2.2	Ponto Focal.....	160
4.2.3	Unidade/Harmonia.....	160
4.2.4	A cor.....	161
4.3	Protótipos de Alta-Fidelidade.....	162
4.3.1	Zona de Interesse.....	164
4.3.2	Texto Escrito.....	165
4.3.3	As Imagens versus Cor.....	167

4.3.4	As Imagens versus Movimento e Som.....	170
4.3.5	Desenvolvimento dos Protótipos de Alta-Fidelidade (versão do aluno)	173
4.3.6	Desenvolvimento dos Protótipos de Alta-Fidelidade, para a versão do professor.....	187
4.4	Tecnologias Utilizadas no Desenvolvimento do Software “Palavras e Palavras”	194
	Conclusões.....	197
Capítulo 5.....		199
	Testes e Resultados.....	199
5.1	Testes e Resultados Obtidos a Nível Quantitativo	201
5.1.1	Testes e Resultados Obtidos Através da Ficha de Avaliação de Software Educativa Parte III de Rodrigues.....	202
5.1.2	Resultados Obtidos Através da Nova Taxionomia de Avaliação de Software Educativa (Parte III).....	208
5.2	Testes e Resultados obtidos a Nível Qualitativo	216
5.2.1	Resultados da Avaliação Heurística.....	217
5.2.2	Resultados Obtidos Através da Ficha de Avaliação de Software Educativa Parte III de Rodrigues.....	219
	Conclusões	221
Capítulo 6.....		223
	Conclusões e Trabalho Futuro	223
6.1	Conclusões.....	223

6.2 Trabalho Futuro.....	227
Bibliografia.....	229
Apêndices.....	250
Anexos.....	266

Índice de Tabelas

Tabela 1.1 – Turno das Escolas.....	6
Tabela 2.1 Atividades em Contexto de Sala de Aula, (Antunes C. , 2005)	28
Tabela 2.2 Taxonomia de Bloom, Níveis Cognitivos e seus objetivos fonte: (Telles, 2005)	40
Tabela 2.3 Taxonomia de Bloom, Área Afetiva e seus objetivos, fonte: (Telles, 2005)	41
Tabela 2.4 Estudo comparativo da Taxonomia de Bloom e da Nova Taxonomia de Anderson	41
Tabela 2.5 Hierarquia dos cinco níveis da Taxonomia Solo (Biggs J. , Teaching for Quality Learning at University, 2003).....	47
Tabela 2.6 Descrição da grelha classificativa, (Rodrigues E. , A Contribution for the Evaluation of Educational Interactive Software, 2002).....	51
Tabela 2.7 <i>Software</i> de Ferramentas, (Rodrigues E. , A Contribution for the Evaluation of Educational Interactive Software, 2002).....	52
Tabela 2.8 Objetivos das características de nível II	58
Tabela 2.9 Descrição da norma ISSO 9126, adaptado (International standards for HCI and usability, 2003).....	72
Tabela 3.1 Teorias e Modelos de Aprendizagem.....	97
Tabela 3.2 Objetivos dos Quadros de Costa.....	100
Tabela 3.3 Significado dos Vértices no Modelo Interativo Adaptável.....	112

Tabela 3.4 Descrição do Método Interativo Adaptável Orientado para o Ensino	118
Tabela 4.1 Tabela de Requisitos Funcionais - Utilizador Aluno.....	128
Tabela 4.2 Tabela de Requisitos Funcionais - Utilizador Professor/Educador	128
Tabela 4.3 Tabela dos Requisitos Não Funcionais do <i>Software</i> “Palavras e Palavras”	130
Tabela 4.4 Simbologia do DFD.....	131
Tabela 4.5 Diferenças entre os Níveis dos DFD’S.....	132
Tabela 4.6 Descrição dos Processos dos níveis 0 e 1 do <i>Software</i> “Palavras e Palavras” .	138
Tabela 4.7 Entradas/Saídas de Fluxo de Dados do nível 0 do <i>Software</i> “Palavras e Palavras”	139
Tabela 4.8 Teorias do Reconhecimento de Padrões	145
Tabela 4.9 Leis percetivas.....	146
Tabela 4.10 Níveis do Processo Percetivo da Seleção Visual	146
Tabela 4.11 Funções do texto no <i>Software</i> “Palavras e Palavras”	165
Tabela 4.12 Valores compreendidos do tamanho da letra em relação à idade, adaptado (Diéguez, 1995).....	166
Tabela 4.13 Papel da imagem versus função no design do ecrã, adaptado de (Rivlin, Lewis, & Davies-Cooper, 1990).....	167
Tabela 4.14 Funções didáticas da imagem adotadas no <i>Software</i> “Palavras e Palavras” .	169
Tabela 4.15 Funções do som (através do discurso direto) no <i>Software</i> “Palavras e Palavras”	171
Tabela 5.1 Resultado do Feedback no <i>Software</i> “Palavras e Palavras”	202

Tabela 5.2 Resultado do Grau de participação do aluno no Software “Palavras e Palavras”	202
Tabela 5.3 Resultado do Grau de controlo do aluno no Software “Palavras e Palavras” .	203
Tabela 5.4 Resultados Materiais de Apoio na Clareza de Instruções no Software “Palavras e Palavras”	204
Tabela 5.5 Resultados Materiais de Apoio na Quantidade de Informação no Software “Palavras e Palavras”	204
Tabela 5.6 Resultado da avaliação dos aspetos visuais em relação aos aspetos pedagógicos do Software “Palavras e Palavras”	206
Tabela 5.7 Resultado da avaliação da interatividade e motivação em relação ao aluno nos aspetos pedagógicos do Software “Palavras e Palavras”	206
Tabela 5.8 Resultado do item estimula a criatividade em relação aos aspetos pedagógicos do Software “Palavras e Palavras”	206
Tabela 5.9 Resultado do item resposta exige reflexão por parte do aluno em relação aos aspetos pedagógicos do Software “Palavras e Palavras”	207
Tabela 5.10 Resultado do item Interatividade - Função do Texto – Apelativa no Software “Palavras e Palavras”	208
Tabela 5.11 Resultado do item - Função do Texto – Interativa no Software “Palavras e Palavras”	208
Tabela 5.12 Média obtida pela Função Informativa no Software “Palavras e Palavras” .	209
Tabela 5.13 Média obtida pela Função Apelativa no Software “Palavras e Palavras”	210
Tabela 5.14 Média da Função Poética no Software “Palavras e Palavras”	210
Tabela 5.15 Média da Função Indagadora no Software “Palavras e Palavras”	210
Tabela 5.16 Média da Função Absoluto no Software “Palavras e Palavras”	212

Tabela 5.17 Média da Função de Colocação Estratégica no Software “Palavras e Palavras”	212
Tabela 5.18 Média da Função de Reação do Utilizador no Software “Palavras e Palavras”	212
Tabela 5.19 Resposta ao item 1 do Software “Palavras e Palavras”	219

Índice de Figuras

Figura 2.1 Interação dos 4 subsistemas do Processo Ensino-Aprendizagem, adaptado de (Altet, 2000).....	19
Figura 2.2 ZDP Zona de Desenvolvimento Proximal, adaptado (Martins, 2003)	26
Figura 2.3 Critérios Pedagógicos de Reeves, adaptado (Campos, 1994)	36
Figura 2.4 Critérios de Interface de Reeves.....	37
Figura 2.5 Representação da ligação de dois critérios da avaliação pedagógica de Reeves	38
Figura 2.6 Os seis níveis da Taxonomia de Bloom.....	39
Figura 2.7 Nova Taxonomia, (Marzano & Kendall, 2008).....	42
Figura 2.8 Comparação entre as duas Taxonomias, (Marzano & Kendall, 2008).....	43
Figura 2.9 Comparação entre a Taxonomia de Bloom e de Anderson (Wilson, 2006)	44
Figura 2.10 Esforço Mental - adaptado (Corrêa, 2003).....	45

Figura 2.11 Objetivos a Avaliar (Costa, 1999), adaptado (E.Rodrigues, J.Moreno, & D.Grilo, 2009).....	49
Figura 2.12 Elementos Estruturantes da Avaliação de Software (Costa, 1999), adaptado (E.Rodrigues, J.Moreno, & D.Grilo, 2009).....	50
Figura 2.13 Modelo de Qualidade de Software de MacCall, adaptado (MacCall, Richards, & Walters, 1997).....	56
Figura 2.14 Modelo de Boehm (Boehm, Brown, Kaspar, Lipow, Machleod, & Merritt, 1978) de Qualidade de Software (Herbert & Price, 1995), adaptado.....	59
Figura 2.15 Abordagem Ergonómica das Tarefas, adaptado (Eason K. , 1991).....	64
Figura 2.16 Modelo Human Computer Interface (HCI)	65
Figura 2.17 Envisage communication between HCI designers and software engineers, (Paula, Barbosa, & Lucena, 2005).....	66
Figura 2.18 Relação entre as três abordagens de software, fonte (Bevan N. , Quality in Use: Meeting User Needs for Quality, 1999).....	73
Figura 2.19 Abordagens para a qualidade de produtos de software, fonte (Bevan N. , Quality in Use: Meeting User Needs for Quality, 1999).....	73
Figura 2.20 Fases do processo de qualidade de desenvolvimento de software, fonte (ISID, 1999).....	74
Figura 2.21 Modelo em Cascata/Queda de Água/WaterFall, fonte (Sommerville, 2001) ..	78
Figura 2.22 Modelo de Prototipagem	79
Figura 2.23 Modelo em V descreve o paralelismo entre as atividades de desenvolvimento e testes ao software, fonte (Neto A. C., s/data), cita (Craig & Jaskiel, 2002)..	80
Figura 2.24 Regra de 10 de Myers, fonte (Tamashiro, 2010)	80
Figura 2.25 Modelo Híbrido, fonte (Silva & Figueiredo, 1994).....	82

Figura 2.26 Processo UID, 1991-1999 Interface Systems international Inc Last Modified: March 20, 1999, (ISID, 1999) , referido por (Rodrigues & Moreno, 2006).....	83
Figura 2.27 Modelo em Espiral, fonte (Sommerville, 2001).....	85
Figura 2.28 Etapas do Modelo de Entrega Incremental, (Sommerville, 2001)	87
Figura 2.29 Modelo de Entrega Incremental, adaptado de ((Miguel, 2006), (Desconhecido, engenharia de software).....	88
Figura 2.30 Desenvolvimento de Software Adaptativo, (Pressman R. , 2005).....	91
Figura 2.31 Extreme Programming Process, fonte (Pressman R. , 2005).....	92
Figura 3.1 Conjugação do Processo UID e do Modelo HCI	105
Figura 3.2 Modelo Interativo Adaptável	111
Figura 3.3 Modelo Interativo Adaptável em Desequilíbrio	112
Figura 3.4 Modelo Interactivo Adaptável Orientado para o Ensino.....	113
Figura 3.5 Método Interativo Adaptável Orientado para o Ensino	114
Figura 3.6 Pormenorização do Método Interativo Adaptável Orientado para o Ensino...	116
Figura 4.1 Interligação de Contextos.....	122
Figura 4.2 Modelo da definição de contexto de uso, retirado de (Sousa, 2008)	123
Figura 4.3 Elementos de um requisito, adaptado de (Dahlstedt, 2003)	125
Figura 4.4 - Representação dos Requisitos.....	125
Figura 4.5 Características dos Requisitos Não Funcionais, (Sommerville & Sawyer, 1997) adaptado.....	126
Figura 4.6 Etapas do processo de criação dos requisitos	127

Figura 4.7 Representação da decomposição de Diagramas de Fluxo de Dados (DFD'S), adaptado de (Baldas, Campos, & Coutinho, 2006).....	133
Figura 4.8 Diagrama de Contexto do nosso estudo, Software “Palavras e Palavras”	134
Figura 4.9 DFD Nível 0 do nosso estudo <i>Software</i> “Palavras e Palavras”	135
Figura 4.10 DFD Nível 1 do Processo P1 Inserir Jogo do <i>Software</i> “Palavras e Palavras” .	136
Figura 4.11 DFD Nível 1 do Processo P6 Jogar Jogo do <i>Software</i> “Palavras e Palavras” ...	137
Figura 4.12 Fluxograma Geral do DFD nível 0 do <i>Software</i> “Palavras e Palavras”	141
Figura 4.13 Fluxograma Jogos que representa o processo P6 do <i>Software</i> “Palavras e Palavras”	142
Figura 4.14 Fluxograma Fichas de Trabalho do <i>Software</i> “Palavras e Palavras”	143
Figura 4.15 Fluxograma Geral Professor/Educador do <i>Software</i> “Palavras e Palavras”..	151
Figura 4.16 Fluxograma respeitante à atividade do professor no jogo “Descobre Palavras” do <i>Software</i> “Palavras e Palavras”	152
Figura 4.17 Fluxograma respeitante à atividade do professor no jogo “Constrói Palavras”	153
Figura 4.18 Fluxograma respeitante à atividade do professor no jogo “Constrói Frases”	154
Figura 4.19 Fluxograma respeitante ao professor na atividade “Pontuação” no <i>Software</i> “Palavras e Palavras”	155
Figura 4.20 Fluxograma respeitante ao professor na atividade “Fichas de Trabalho” no <i>Software</i> “Palavras e Palavras”	156
Figura 4.21 Equilíbrio visual e desequilíbrio visual, adaptado de (Boyle, 1997)	158
Figura 4.22 Variação na obtenção do equilíbrio visual, adaptado de (Boyle, 1997)	158

Figura 4.23 Ponto focal por contraste, adaptado de (Szabo & Kanuka, 1999)	159
Figura 4.24 Ponto focal por isolamento, adaptado de (Szabo & Kanuka, 1999).....	159
Figura 4.25 Exemplo de uma composição desunificada (à esquerda) e uma composição unificada (à direita), adaptado de (Szabo & Kanuka, 1999).....	160
Figura 4.26 Estrutura de Formatação da Interface do Software “Palavras e Palavras”	162
Figura 4.27 Zona de interesse e áreas adjacentes de interesse para o aluno no Software “Palavras e Palavras”	163
Figura 4.28 As áreas delimitadas a tracejado correspondem a campos de intensidade visual, adaptado (Castela, 2004).....	164
Figura 4.29 Execução do setup da nossa aplicação.....	195
Figura 5.1 Metodologia dos testes de aceitação do Software “Palavras e Palavras”	200

Índice de Imagens

Imagem 4.1 Áreas do Cérebro Humano (Drews, 2007).....	147
Imagem 4.2 Detalhe do Córtex Visual adaptado de (Hyvärinen, s/Data)	148
Imagem 4.3 Via ótica para V1, (Hyvärinen, s/Data)	148
Imagem 4.4 Ecrã de Inicial do Software “Palavras e Palavras” (Rodrigues, Grilo, & Rodrigues).....	174
Imagem 4.5 Segundo ecrã do Software “Palavras e Palavras” (Rodrigues, Grilo, & Rodrigues).....	175

Imagem 4.6 Ecrã de inserção do aluno no Software “Palavras e Palavras” (Rodrigues, Grilo, & Rodrigues).....	176
Imagem 4.7 Ecrã das tarefas a realizar pelo aluno no Software “Palavras e Palavras” (Rodrigues, Grilo, & Rodrigues).....	177
Imagem 4.8 Ecrã de seleção da letra para o Jogo DP do Software “Palavras e Palavras”	178
Imagem 4.9 Ecrã do jogo DP nível 1 do Software “Palavras e Palavras”, versão aluno	179
Imagem 4.10 Ecrã do jogo CP do Software “Palavras e Palavras” (Rodrigues, Grilo, & Rodrigues).....	182
Imagem 4.11 Ecrã do jogo do Software “Palavras e Palavras”	183
Imagem 4.12 Ecrã Fichas de Trabalho do Software “Palavras e Palavras”	184
Imagem 4.13 Ecrã da Pontuação do Aluno no Software “Palavras e Palavras”	185
Imagem 4.14 Mocho gestor de conteúdos do Software “Palavras e Palavras”	186
Imagem 4.15 Ecrã da Inserção dos dados do Jogo DP do Software “Palavras e Palavras”	187
Imagem 4.16 Ecrã de inserção de dados para o jogo CF do Software “Palavras e Palavras”	189
Imagem 4.17 Ecrã do jogo CF do Software “Palavras e Palavras”, versão do Professor	191
Imagem 4.18 Ecrã da Inserção das Fichas de Trabalho do Software “Palavras e Palavras” versão do professor.....	192
Imagem 4.19 Ecrã da Pontuação do Software “Palavras e Palavras” versão do professor	193

Índice de Ilustrações

Ilustração 4.1 Excerto do código da canção do <i>Software</i> “Palavras e Palavras”	173
Ilustração 4.2 Excerto do código que mostra a lista dos alunos no <i>Software</i> “Palavras e Palavras”	174
Ilustração 4.3 Excerto do código para criar um jogador do <i>Software</i> “Palavras e Palavras”	176
Ilustração 4.4 Excerto do código da utilização de timer no jogo DP do <i>Software</i> “Palavras e Palavras”	180
Ilustração 4.5 Código para a associação das janelas no jogo DP do <i>Software</i> “Palavras e Palavras”	180
Ilustração 4.6 Excerto do código de contagem de respostas certas e erradas para o jogo DP do <i>Software</i> “Palavras e Palavras”	185
Ilustração 4.7 Excerto do código referente à inserção de imagens no jogo DP do <i>Software</i> “Palavras e Palavras”	188
Ilustração 4.8 Excerto do código referente à inserção no jogo CP do <i>Software</i> “Palavras e Palavras”	190

Índice de Gráficos

Gráfico 5.1 Média dos Resultados da Interatividade do Software “Palavras e Palavras” .	203
Gráfico 5.2 Média dos itens dos Materiais de Apoio no Software “Palavras e Palavras” .	205
Gráfico 5.3 Média obtida na avaliação dos aspetos pedagógicos do Software “Palavras e Palavras”	207
Gráfico 5.4 Resultado das funções do Texto no Software “Palavras e Palavras”	209
Gráfico 5.5 Resultado obtido pela Função do Som no Software “Palavras e Palavras” ...	211
Gráfico 5.6 Resultado obtido por Funções da Imagem no Software “Palavras e Palavras”	213
Gráfico 5.7 Média obtida em cada item da usabilidade no Software “Palavras e Palavras”	214
Gráfico 5.8 Média obtida em cada item das Funções didáticas do Som no Software “Palavras e Palavras”	215

Capítulo 1

Introdução

Na sociedade atual a utilização das Tecnologias de Informação e Comunicação (TIC) é um dado adquirido e de vital importância que faz parte do nosso dia a dia. Sendo um “veículo” de conhecimento, o seu papel no contexto educacional/educativo é muito valioso, pois fortalece o sistema educacional/educativo. Contudo, é visto como um material de apoio interativo e não como um substituto do professor.

Segundo Roberto Santiago (Santiago, 2000) ao citar Caivano *“A conjugação das Tecnologias de Informação e Comunicação estão a envolver o mundo numa potente rede de difusão de conteúdos. A irreversível rapidez das mudanças económicas está a modificar as relações sociais em todas as áreas e não parece que vá parar nas portas da escola. A suposta centralidade do conhecimento como variável directamente produtiva, que configura a chamada economia do conhecimento”, põe as instituições educativas sob a incómoda lupa do mercado. E desde o balcão do mercado se afirma que a escola está obsoleta e que há que rever a função própria da escola, a definição, transmissão e legitimação dos conhecimentos socialmente considerados relevantes”*.

A partir desta afirmação podemos concluir que embora a presença das TIC na Escola seja já uma constante, o seu papel no processo educativo ainda não atingiu os limiares desejados a nível administrativo e pedagógico, pois *“ a escola é uma instituição envelhecida numa sociedade moderna e em contínua mudança”*, refere Roberto Santiago (Santiago, 2000).

Para Robert Taylor *“ (...) a utilização da informática em contexto educacional, permite ao computador servir de tutor (professor), de ferramenta e como aprendiz (aluno) ”*, (Taylor, 1980). Taylor analisa a utilização dos meios informáticos em contexto educacional sob três perspetivas:

- Como meio de ensino/aprendizagem – reflete-se na aprendizagem apoiada por computador;
- Como ferramenta de trabalho – reflete-se na aprendizagem complementada com o computador;
- Como objeto de estudo – reflete-se na aprendizagem relacionada com o sistema computacional.

Todavia, os atributos do computador de nada serviriam para criar ambientes de aprendizagem, se não existissem, subjacentes a estes atributos, modelos para apoiar e colocar em prática a apresentação de conteúdos pedagógicos, o tratamento da informação, diferentes formas de interatividade e a reprodução do conhecimento. Para tal, a psicologia da aprendizagem tem contribuído de forma indicativa na aplicação dos conceitos das teorias comportamentais e cognitivas, assim como os modelos de ensino/aprendizagem centrados no professor e no aluno e as metodologias de desenvolvimento de *software*, entre outras, tal como apresentado no capítulo 2.

A inserção dos computadores no processo educacional permite que os educadores passem de um modelo de educação cartesiano linear (modelo de transmissão de conhecimento), para um modelo horizontal (modelo de diálogo), refere Luciano Gamez (Gamez, 1998). Assim, é importante salientar que a utilização de *software* educacional/educativo, como instrumento de ensino, deve ir ao encontro dos objetivos e das metodologias de ensino/aprendizagem adotado pelas instituições no seu projeto pedagógico.

Como tal, deve-se avaliar o ambiente educacional em que os alunos estão inseridos, tendo sempre presente a maneira como eles aprendem e as estratégias utilizadas para facilitar essa aprendizagem.

Segundo Elsa Rodrigues (Rodrigues E. S., 2002) “ (...) *qualquer processo educativo deve ser dinâmico, constantemente repensado e renovado à medida que se vai desenvolvendo, não se devendo submeter a um modelo estático (...)* ”.

O “Projeto de Investigação / Ação na Área do *Software* Educacional/Educativo” estudo realizado por Elsa Rodrigues no qual participaram alunos e professores na avaliação do *software* educacional/educativo, existente no mercado Português para o 1º ciclo básico, visou detetar possíveis lacunas a nível pedagógico (indo de encontro a Santiago) e lacunas a nível de

interface, usabilidade e da satisfação do utilizador final, seja ele aluno ou professor (Rodrigues E. S., 2002).

De salientar que a faixa etária destes alunos se situa entre os 6 e os 10 anos de idade. No entanto, prevendo a integração do ensino especial nas ditas “escolas normais”, englobaram-se os alunos e os professores do ensino especial.

Para Elsa Rodrigues (Rodrigues E. S., 2002) “ (...) Este estudo baseou-se em recomendações ergonómicas e pedagógicas, visou o desenvolvimento de uma ficha de avaliação, composta por um conjunto sistemático de questões a utilizar no processo de avaliação de conformidade ergonómica de Software Interactivo Educativo (...). A sistematização deste processo deu origem à ferramenta de trabalho – Ficha de Avaliação de Software Interactivo Educativo (...) Há inúmeros aspectos que não devem ser esquecidos durante todo este processo, e, neste sentido, a nossa ficha de avaliação teve como principal função salientar as características importantes a ter em conta nos procedimentos de avaliação. Foi nesta perspectiva que o nosso trabalho se tornou pertinente. Pois provou as limitações e os problemas que existem no Software Educativo, que circula no nosso mercado. ”

Com base neste projeto de investigação e face aos problemas encontrados, o presente estudo iniciou-se com vista a resolver as lacunas detetadas.

1.1 Contextualização

Tendo como partida o estudo anterior “Ação na Área do Software Educativo/Educativo”, iniciou-se novo estudo para colmatar as lacunas anteriormente detetadas, denominado “Metodologia de Desenvolvimento para Software Interativo Adaptável Orientado para o Ensino”.

Em qualquer estudo, a participação dos utilizadores deve ser tida em conta, pois quanto maior for a sua participação maior será a sua satisfação como utilizador final. Desenvolver uma metodologia de ensino interativa, pedagógica e adaptável implica a necessidade de incorporar no modelo uma teoria interna de aprendizagem. Além disso, as teorias de interação com o utilizador, como sugerem Norman e Draper (Norman & Draper, User Centered System Design:

New Perspectives, 1986), apontam para a importância do utilizador ter uma perceção integrada sobre o sistema e a inclusão da representação mental do utilizador.

Assim sendo, optamos por efetuar um " estudo de campo" no qual a participação dos alunos e dos professores foi imperativa, visto que todos os alunos têm características, interesses, capacidades e necessidades de aprendizagem que lhe são próprias. A participação dos professores foi muito útil, pois só eles podem identificar as necessidades dos alunos e selecionar metodologias do ensino que se podem aplicar para resolver essas necessidades.

Visto ser um estudo de campo, torna-se necessário descrever as estruturas das escolas (onde decorreu o estudo), quer a nível dos recursos humanos e dos recursos materiais, quer a estrutura do ensino vigente em Portugal.

Em Portugal, a educação inicia-se num âmbito não obrigatório com o Pré-escolar, dirigido a crianças com idades compreendidas entre os 3 anos e os 6 anos, idade que corresponde à entrada na escolaridade obrigatória. A escolaridade obrigatória é designada como Ensino Básico e tem a duração de 9 anos. Este ciclo de obrigatoriedade inicia-se aos 6 anos de idade e termina aos 15 anos. Organiza-se em três ciclos sequenciais, o primeiro de quatro anos, o segundo de dois anos e o terceiro de três anos, sendo as suas competências:

- 1º Ciclo - O ensino é global e visa o desenvolvimento de competências básicas. A Lei de Bases do Ensino Educativo (L.B.S.E), no seu artigo 8º (Educação, 2005), apresenta como objetivos para o 1º ciclo "*...o desenvolvimento da linguagem oral e a iniciação e progressivo domínio da leitura e da escrita, das noções essenciais da aritmética e do cálculo, do meio físico e social e das expressões plástica, dramática, musical e motora*". Funciona em regime de mono docência, com recurso a professores especializados em determinadas áreas, caso existam alunos do ensino especial;
- 2º ciclo - Está organizado por áreas de estudo interdisciplinares., funciona principalmente em regime de professor por área. Tendo como objetivos "*a formação humanística, artística, física e desportiva, científica e tecnológica e a educação moral e cívica, visando habilitar os alunos a assimilar interpretar crítica e criativamente a informação, de modo a possibilitar a aquisição de métodos instrumentos de trabalho e de conhecimento que permitam o prosseguimento da sua formação, numa perspetiva do desenvolvimento de atitudes ativas e conscientes perante a comunidade e os seus problemas mais importantes*";

- 3º Ciclo – Funciona *“segundo um plano curricular unificado, integrando áreas vocacionais diversificadas, e desenvolve-se em regime de um professor por disciplina ou grupo. a aquisição sistemática e diferenciada da cultura moderna, nas suas dimensões humanística, literária, artística, física e desportiva, científica e tecnológica, indispensável ao ingresso na vida ativa e ao prosseguimento de estudos, bem como a orientação escolar e profissional que faculte a opção de formação subsequente ou de inserção na vida ativa, com respeito pela realização autónoma da pessoa humana”*. Aos alunos que completem com sucesso o 3.º ciclo ser-lhes-á atribuído o diploma do ensino básico.

Em Portugal existem mais dois níveis de ensino, sendo eles:

- Secundário - Continuação do 3º ciclo com a duração de três anos;
- Ensino Superior – Continuação do secundário, cuja duração depende do curso escolhido pelo aluno.

Este estudo abrangeu somente o 1º ciclo básico e decorreu ao longo de quatro anos consecutivos e, posteriormente, mais dois anos de forma irregular (entre 2005 – 2009 e 2010-2011), em três escolas do distrito de Beja.

Durante o estudo, existiram algumas dificuldades relacionadas com recursos humanos e com recursos de materiais entre outras. Os problemas relacionados com os recursos humanos foram causados essencialmente - pela lei de recrutamento de professores vigente em Portugal. Esta lei obriga a maioria dos professores a participarem num concurso de seleção anual, segundo o Decreto-Lei nº27/2006 (Constitucional, 2006), esta seleção tem por base, um conjunto de critérios a que os professores estão sujeitos.

De salientar que o professor ao apresentar a sua candidatura seleciona uma região demográfica de Portugal onde pretende lecionar e não uma escola. Como tal, o professor pode não ficar na escola que lecionou no ano anterior.

Para o estudo em causa, este processo de recrutamento de professores foi negativo, porque todos os anos surgiram professores diferentes, com métodos de ensino diferentes e, por vezes, com objetivos diferentes. Esta situação acarretou alterações constantes aos requisitos funcionais.

As turmas devem ser constituídas por alunos do mesmo ano de escolaridade, em número de vinte e dois alunos. No entanto, se na turma for integrado um aluno do ensino especial, esta deve diminuir o número de alunos em dois.

Convém referir, que a obrigação dos professores é o cumprimento do programa educacional indicado pelo Ministério da Educação. Como tal, a este estudo os professores e os alunos dedicaram uma manhã ou uma tarde por semana, nas horas definidas como área de projeto. Perante tal, optou-se por criar uma tabela de turnos para as escolas, que se apresenta na tabela 1.1.

Escola	Dia de Semana	Período
Escola A	Terça-Feira	Manhã
Escola B	Quarta-Feira	Manhã
Escola C	Quinta-Feira	Tarde

Tabela 1.1 – Turno das Escolas

Os recursos materiais eram escassos, a nível de computadores, impressoras, *data show*, scanners, papel, etc.

Para o estudo, duas escolas foram selecionadas de forma aleatória e em cada escola foi selecionada uma turma do primeiro ano (1º ano), visto o primeiro ano corresponder à iniciação da aprendizagem da leitura e da escrita. A terceira escola selecionada foi por motivos muito especiais: a turma era constituída, na totalidade, por alunos do ensino especial. Estas turmas participaram no estudo durante o ciclo de vida do mesmo.

A partir deste momento as escolas passam a ser identificadas como:

- Escola A / Turma A;
- Escola B / Turma B;
- Escola C / Turma C.

Na Escola A / Turma A, a turma era composta por alunos de vários anos de escolaridade (1º ano, 2º ano, 3º ano e 4º ano) em número de 12 alunos. Na Escola B / Turma B a turma era constituída por vinte alunos, sendo um, do ensino especial. Enquanto a Escola C / Turma C era constituída por oito alunos do ensino especial. Estas situações alteraram o esquema estrutural do estudo.

Este estudo foi bastante complexo pois teve de ter em conta várias áreas de estudo e posterior necessidade de as combinar e relacionar.

1.2 Objetivos da Tese

Esta investigação teve como objetivo geral apoiar os professores do 1º ciclo no processo de ensino aprendizagem, em contexto de sala de aula, desenvolvendo uma ferramenta de apoio.

A ferramenta de apoio tem de estar relacionada com as Novas Tecnologias de Informação e Comunicação. Como tal, tem de contemplar a interatividade, usabilidade e adaptabilidade para o utilizador. Por outro lado, deve ser didática e pedagógica de forma a refletir o processo ensino/aprendizagem. Por outras palavras, a ferramenta de apoio tem de espelhar o processo ensino/aprendizagem. Portanto, esta ferramenta é “um *software* “ interativo adaptável orientado para o ensino, com vista a apoiar o professor no processo ensino aprendizagem, em contexto de sala de aula. Como tal, para o desenvolvimento dessa ferramenta o “*software*” implicou a criação de um novo modelo e método de desenvolvimento de *software* interativo, adaptável e orientado para o ensino.

O processo de aprendizagem por si só é bastante complexo e adicionar um artefacto cognitivo como o *software* educacional/educativo ao ambiente de aprendizagem, faz com que esta complexidade obrigatoriamente se torne presente, de acordo com David Squires e Anne McDougall (Squires & McDougall, 2003).

A identificação por Donald Norman (Norman D. A., Cognitive artifacts, 1991) de duas visões do uso de um artefacto cognitivo – a visão de sistemas e a visão pessoal – demonstra esta mesma noção de complexidade. Da visão de sistema, o *software* educacional, contribui, de alguma maneira, para o aumento do desempenho cognitivo. Neste sentido, a cognição é

partilhada entre o utilizador e o *software*, o que, teoricamente, faz com que a tarefa seja concluída de maneira mais fácil. Delegar algumas tarefas ao *software*, associando a isto a necessidade do utilizador em apreender novos conceitos e habilidades faz com que esta relação se torne num processo muito complexo.

Esta complexidade fez com que os objetivos fossem perfeitamente identificados e definidos. Para tal, tivemos que definir os seguintes objetivos específicos:

- Sistematizar um conjunto de atividades relacionadas com o processo de análise ergonómica das tarefas, quando esta está implícita no desenvolvimento de *software* interativo e adaptável para o ensino;
- Desenvolver uma metodologia interativa e adaptável para o ensino;
- Desenvolver um modelo interativo adaptável no processo ensino aprendizagem;
- Adaptar o referido modelo ao utilizador qualquer que seja ele;
- Testar a capacidade evolutiva do aluno face à metodologia desenvolvida;
- Testar a capacidade evolutiva do aluno face à evolução do modelo, ou seja à medida que o modelo evolui o aluno deve aprender conceitos mais complexos.

1.3 Validação dos Resultados

O método desenvolvido foi testado durante todo o seu ciclo de vida. Ao longo deste tempo, foram feitas muitas alterações a pedido dos utilizadores, que abrangeram diferentes pontos (cores, interação, mensagens, etc) e que serão apresentados no capítulo 4. Após esta fase de teste, esteve mais um ano a ser avaliado em termos de qualidade e resultados de aprendizagem.

Estes resultados foram adquiridos através de uma grelha de avaliação, elaborada especificamente para este fim. A grelha apresenta uma estrutura de *checklist*, que funciona como uma lista de verificação de itens a serem observados. De acordo com o nosso estudo, os resultados obtidos são visíveis através de um gráfico. Existem alguns itens, em que as questões não permitem a criação de um gráfico, tais como: *Software Identification, Tools of Exploration*(y/n), visto as perguntas terem um carácter não mensurável. Enquanto outras questões têm um carácter mensurável, portanto são avaliadas segundo uma escala

compreendida entre 0- 5 pontos; “zero” indica que o *software* não contempla a pergunta; “um” - corresponde ao muito mau; “dois” corresponde ao não satisfaz ; “três” corresponde ao satisfaz, ou seja satisfaz minimamente os objetivos pretendidos; “quatro” corresponde ao bom, o que indica que o *software* apresenta características acima da média; “cinco” corresponde ao muito bom, o que revela que o *software* está de acordo com os objetivos e satisfaz bastante o utilizador.

1.4 O contributo desta Tese

A utilização da metodologia de desenvolvimento de “*Software* Interativo Adaptável Orientado para o Ensino”, contribuiu de forma significativa no desenvolvimento de *software* para o 1º ciclo de aprendizagem. Visto que a utilização do *software*, com as características que desenvolvemos, permite aos alunos, que se encontram em fase de iniciação da aprendizagem da leitura, uma abordagem à linguagem escrita de forma interativa através da descoberta da palavra e estrutura da frase.

Esta metodologia permite-lhes a visualização da palavra global através de associações de imagens e palavras, assim como uma progressiva memorização para posterior divisão silábica e construção de outras palavras.

Os professores que utilizarem em contexto de sala de aula o *software* com esta metodologia, poderão sistematizar conteúdos, suprir as dificuldades do aluno na medida em que poderão alterar as palavras e frases conforme for necessário aos seus alunos, de uma forma individualizada. Para além disso, o facto de ser possível a inclusão de material didático, como fichas, e o registo dos resultados obtidos por cada um dos alunos, e ainda o facto de o resultado ser visualizado através de um gráfico, o qual permite seleccionar um intervalo de tempo, constitui um instrumento de trabalho facilitador da avaliação e gestão das aprendizagens dos alunos.

De forma a validar este projeto de investigação foram publicados alguns artigos e um poster em conferências internacionais, seminários e em revista de opinião que contribuíram para o seu desenvolvimento. Apresentamos a lista de publicações nesta área:

Conferências Internacionais:

- *IV International Conference on Multimedia and Information and Communication Technologies in Education (m-ICTE2006), Espanha – Sevilha, 2006. Tema “An Approach for Interactive Educational Software in Classroom Setting.”*

“Abstract: This paper presents a systematic set of activities related to the ergonomic analysis process of interactive educational software in a classroom setting. It uses the User Interface Design (1991-1999 Interface Systems International) with the aim of developing an ergonomic model for Human Computer Interaction (HCI) in the context of interactive educational software. The model should be adaptable to any kind of user and evolutive, i.e., the model should be able to switch to more advanced concepts once the user has learned the previous ones.”

- *International Conference of Education, Research and Innovation (ICERI 2009), Espanha – Madrid 2009. Tema “Main Path To Learning Using Interactive Adjustable Software”.*

“Abstract: This document aims at, to present a summary of project that it led to the development of Adjustable Interactive Model, which has as base the guided adjustable interactive method for education. This project had as objective to support the teachers in the education process learning, in context of classroom. For such, we had that to define the following objectives: Systemize a set of activities related with the analysis process ergonomic of the tasks, when this is implicit in the development of Software Adaptable Interactive for Education; To develop Adjustable Interactive Model in the process education learning; To adapt the cited model to the user any that is it; To develop the evaluative capacity to the measure that the students learns more complex concepts. To reach these objectives, we make detailed study to some methodologies of development of software related with ergonomics pedagogical aspects, and methodologies to evaluate the related quality of software in [2]. Our challenge consisted of relating these methodologies, in the process to teach “of interactive, ergonomic and adjustable form”, using the Technologies of Information and Communication (TIC). So that this was possible, we decide to

divide this analysis in two groups: Group - teacher; Group - student. Thus, we not only verify the necessities you specify of each one of the groups, but also common aspects. These two groups had presented, in common, the TIC, what it took in them to question its use in the processes to teaching/to learn, of form to reach the considered objectives. For such, we opted to analyzing the methodologies related with pedagogical aspects, ergonomics methodologies, methodologies of quality, Human Computer Interface, User Interface Design. In such a way, Interactive Model was possible to create the appointed model "Adjustable" that involves all these methodologies, and that it allowed in them to create a "Guided Adjustable Interactive Software of Education."

- *V International Conference on Multimedia and ICT in Education (m-ICTE2009) - Lisboa, 2009. Tema: "New Practices of Learning Using Adjustable Software".*

"Abstract: We can perceive that in our society "learning" today is not the same as to just acquire knowledge. We must realise that, in our present society, "learning" is not the same as just acquire knowledge, but it also means to use elementary tools, so that learning can be applied in real life. In this context the teacher has an important role in the definition process of new teaching practices. We have proposed a project that develops a new type of software, which contributes to a more efficient teacher's performance, inside the classroom. This software generates new mechanisms in order to allow the definition of new practices, according to the teacher wants to teach. This new approach allows students to minimize their difficulties. This software was built using Human Computer Interface and User Interface Design models in order to fulfil the goals we want to reach."

- *VII Seminário de Formação Pessoal e Social, na Escola Superior de Educação de Beja, 2007. Tema "Aprendizagem com Recurso às Tecnologias".*

"Resumo: Este artigo pretende visar o trabalho realizado na cadeira de Projeto, da licenciatura em Engenharia Informática da Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Beja. O projeto aspira tornar mais atrativo ensino da leitura e escrita nas escolas básicas. Surgiu de um estudo e análise efetuados pela Dra. Elsa Rodrigues com a duração de mais de um ano em

algumas escolas básicas de Beja, tomando contacto com alunos e professores a fim de entender a metodologia de ensino atualmente utilizada e perceber de que forma se poderia desenvolver um software que pudesse realizar essas tarefas de uma forma mais cómoda, divertida, eficaz e prática. O software em desenvolvimento abrange dois tipos de utilizadores diferentes: os alunos e os professores, educadores e/ou encarregados de educação. Com esta estrutura é possível aos professores gerir todos os conteúdos que a aplicação disponibiliza.”

Revista de opinião:

- Artigo de opinião publicado na revista cienciaportugal.net (Rodrigues & Grilo, CienciaPt - A Educação, Ciência, Tecnologia e Inovação em Portugal - Revista, 2007), 3 de Maio de 2007, nº137- tema: “Desenvolvimento de Software Interativo Educacional, para o 1º ciclo básico”

“ Resumo: A inserção dos computadores no processo educacional permite que os educadores passem de um modelo de educação cartesiano linear (modelo de transmissão de conhecimento), para um modelo horizontal (modelo de diálogo), (Gálvis, 1997). Assim, é importante salientar que, como instrumento de ensino, a utilização dos computadores e do Software Educacional não deve ser isolado dos objetivos e das metodologias de ensino/aprendizagem adotado pelas instituições no seu projeto pedagógico. Portanto, deve-se avaliar o ambiente educacional em que os alunos estão inseridos, tendo sempre em presente a maneira como o indivíduo aprende e as estratégias utilizadas para facilitar essa aprendizagem.”

1.5 Estrutura da Tese

Esta tese apresenta seis capítulos, 16 anexos e três apêndices, ambos identificados pelas letras do alfabeto. O capítulo 1 corresponde à introdução, em relação aos restantes capítulos, anexos e apêndices, apresentam-nos uma breve descrição dos mesmos.

Capítulo 2. Estado da Arte. A motivação para o trabalho de investigação em curso, no âmbito do *software* educacional, acabou por obrigar a desenvolvê-lo em duas vertentes bem distintas, mas justificáveis:

- Por um lado, foi fundamental perceber e avaliar as metodologias que as Ciências de Educação têm vindo a desenvolver e aplicar ao longo das últimas décadas, quais os resultados obtidos e as razões de procurar novas formas de ensinar. Ao longo da 1ª parte deste capítulo, procuramos rever o estado da arte nas Ciências de Educação e a forma como as mesmas se têm vindo a afirmar no dia-a-dia do professor, permitindo-nos optar por aquela ou aquelas que, a nosso ver, melhor se adaptam à condução desta investigação;
- Por outro lado, a evolução das tecnologias de informação e comunicação (TIC) e o seu impacto nas Ciências de Educação abriram oportunidades únicas para ajustar algumas metodologias enunciadas à realidade da sala de aula e aos modelos de aprendizagem. Situando-se este projeto no campo da Engenharia Informática e no desenvolvimento de *software* aplicado aos métodos de aprendizagem em ambiente de sala de aula, tornou-se imperioso estudar as metodologias de desenvolvimento de *software*, especialmente no que diz respeito a *software* para Web e *software* adaptativo. Assim, na 2ª parte deste capítulo faz-se uma incursão pela realidade do que hoje está subjacente à Engenharia de Requisitos, como parte integrante da Engenharia de *Software*.

A leitura das duas partes deste capítulo podem ser efetuadas separadamente e abordam temáticas que, na realidade, poucas vezes ou nunca se cruzam, dada a especificidade e características de cada uma. Mas o trabalho de investigação não pôde avançar sem o esforço desenvolvido para entender o estado da arte das Ciências de Educação, como o grande requisito para o desenvolvimento de *software*.

Esta é uma realidade com que o engenheiro de *software* se depara frequentemente – a sua capacidade de satisfação de requisitos obriga a um esforço complementar no entendimento do problema e das alternativas que os especialistas (do problema) advogam como solução. Só estudando o problema aprofundadamente, o engenheiro de *software* fica com o conhecimento necessário a projetar uma solução. E isto obriga-o, na maior parte dos

casos, a entrar em áreas de conhecimento que, pressupostamente, estariam fora do âmbito das suas competências.

Em resumo, a investigação ficaria incompleta se só se avaliasse o que hoje são as metodologias para o desenvolvimento de *software*, os requisitos para o interface pessoa-computador ou a acessibilidade de um aplicativo, tirando importância ao problema, o porquê de existir problema, as alternativas para solução do problema e os objetivos que se pretendem ver atingidos com a solução tecnológica.

Capítulo 3. Proposta do Novo Modelo. Neste capítulo descreve-se a proposta para o novo modelo que teve como base o estudo descrito no capítulo 2. Como tal, apresenta e explica as metodologias e métodos adaptados para a criação do modelo. O novo modelo inter-relaciona três áreas, sendo elas: *Human Interface Computer*, ergonomia, qualidade de *software* com as teorias de aprendizagem.

Para que o “Modelo Interativo Adaptável “ tivesse “*corpus*” foi necessário criar um método subjacente ao modelo, que se designou como “Método Interativo Adaptável Orientado Para o Ensino (MIAOE)”.

Capítulo 4. Análise Sistemático do Contexto. Foca-se o processo de desenvolvimento de sistemas de informação, que engloba a engenharia de requisitos e a fase de análise, que são descritos os processos de modelagem. Descreve-se o processo de desenvolvimento e as suas fases. Cada fase é apresentada detalhadamente.

Capítulo 5. Teste e Resultados. A importância dos testes no desenvolvimento de *software* e as implicações à qualidade do software, fazem destes, elementos fulcrais. Como tal, este capítulo apresenta os testes efetuados durante todo o estudo, assim como os resultados obtidos através do modelo desenvolvido, representado pelo *software* “Palavras e Palavras”.

Capítulo 6. Conclusões e Trabalho Futuro. Apresentam-se as conclusões a este estudo e apresentam-se alguns caminhos para futuros estudos.

Apêndice A. Ficha de Avaliação de Software Educacional (Parte III) de Rodrigues. Este apêndice tem como objetivo testar o *software* “Palavras e Palavras” em termos de usabilidade, interface, interatividade e aspetos pedagógicos.

Apêndice B. *Nova Taxionomia de Avaliação de Software Educacional (Parte III).* Este apêndice tem como objetivo testar o software “Palavras e Palavras” em termos de interface (cores), usabilidade (controle por parte dos alunos e dos professores, etc.) , interatividade (som, imagens, etc.) e pedagogicamente.

Apêndice C. *Resultado das Funções Didáticas do Som no Software “Palavras e Palavras”.* Este apêndice apresenta quatro tabelas com os resultados obtidos no teste de avaliação às funções didáticas do som.

Anexo A e A1. *As Inteligências Múltiplas.* Estes anexos descrevem as inteligências múltiplas, o seu relacionamento entre elas, as habilidades que cada individuo realiza em cada uma das inteligências.

Anexo B e B1. *Atividades em Contexto de Sala de Aula.* Estes anexos descrevem as atividades a efetuar pelos alunos, os materiais de ensino a usar e as ações que o docente deve realizar, tudo isto em cada uma das inteligências múltiplas.

Anexo C e C1. *Níveis e Objetivos Gerais da Nova Taxonomia.* Descrição e objetivos de cada um dos níveis desta nova taxonomia.

Anexo D. *Taxonomia SOLO – verbos associados a cada SOLO.* Definição dos verbos a utilizar no SOLO 2, SOLO 3, SOLO 4 e SOLO 5.

Anexo E. *Quadro I – Requisitos Técnicos.* Avaliação dos requisitos técnicos para um determinado *software*.

Anexo F. *Quadro II - Conteúdo e Aspetos Pedagógicos da Aplicação.* Descrição dos itens respeitantes aos elementos constituintes dos conteúdos pedagógicos e aos aspetos pedagógicos.

Anexo G. *Quadro III – Interface Gráfica, Interatividade e ferramentas de Exploração.* Este anexo reporta os itens referentes: à interface gráfica, à interatividade e ferramentas de exploração.

Anexo H. *Quadro IV – Usabilidade da Aplicação.* Neste anexo estão descritos os itens referentes à usabilidade

Anexo I. *Origem das normas ISO e/ou IEC.* Significado do título e da origem de uma norma ISO e/ou IEC.

Anexo J. *Normas ISO e outras.* O anexo em causa especifica os princípios e recomendações das normas ISO/IEC, em relação ao contexto de utilização, interface e interação, documentação, processo de desenvolvimento, e capacidade entre outros.

Anexo L. *Palavras e Palavras.iss – Avaliação Heurística.* Este anexo traduz o resultado da avaliação heurística ao *software* “Palavras e Palavras”.

Anexo M. *Código para tratar os clicks no jogo DP do software “Palavras e Palavras”.* Este anexo mostra o código em *C#* para o tratamento dos *clicks*.

Anexo N. *Código para tratar de obter os resultados do aluno.* Este anexo ilustra o código em *C#* para o tratamento dos resultados obtido pelo aluno através das respostas certas e erradas.

Capítulo 2

Estado da Arte

“Embora o homem seja moldado pelas necessidades da estrutura económica e social da sociedade, ele não é infinitamente adaptável. Não só existem certas necessidades fisiológicas que exigem imperiosamente a sua satisfação, mas há ainda, certas qualidades psicológicas inerentes ao homem, que têm de ser atendidas e que provocam certas reacções caso sejam frustradas. Quais são essas qualidades? A mais importante parece ser a tendência para crescer, para desenvolver e realizar potencialidades que o homem formou através da história – como, por exemplo, a faculdade de pensamento criador e crítico e a variar as experiências emocionais e sensoriais.” (Fromm, 1983), refere (Faria, 1994).

A forma como a informação se apresenta nos nossos dias, demonstra uma nova maneira de pensar e estar (Lévy, 1994). Independentemente das mudanças que ocorrem numa sociedade, e das décadas passadas, este conceito é sempre actual, pois a sociedade está em mutação constante. Toda e qualquer alteração numa sociedade reflecte-se no processo educacional, pois este é o alicerce de todas as sociedades. A demanda educacional surge em cada passo do ser humano.

Esta demanda passa por todos os educadores sejam eles pais, avós, tios, professores . Neste capítulo, faremos referência ao papel do professor, do aluno e de produtos multimédia, como partida para os nossos objetivos referidos no capítulo I. Ressalvamos, que ao referir produtos multimédia, focamos especificamente software educativo.

De salientar, que o âmbito deste estudo não foi avaliar a qualidade do *Software* Educacional/Educativo, já que neste caso a qualidade foi um dos requisitos para avaliá-lo quanto às necessidades da população alvo do nosso estudo.

Para tal, tivemos que estudar metodologias de avaliação pedagógicas, de qualidade de *software* e de desenvolvimento de *software*, sendo elas:

- Metodologias de Avaliação Pedagógica
 - Metodologia de avaliação pedagógica de Reeves;
 - Metodologia de avaliação pedagógica de Bloom;
 - Taxonomia de Bloom revista;
 - Taxonomia SOLO;
 - Metodologias de Avaliação Pedagógica do Software Educativo.
- Metodologias de avaliação da qualidade de *software*;
- Modelos Ergonómicos;
- Metodologias de desenvolvimento de *software* ;
- Normas de desenvolvimento de *software*.

No entanto, antes de enveredarmos pelas avaliações referidas, teremos de perguntar a nós próprios, o que é ensinar? O que é aprender? E o que é o processo ensino/aprendizagem?

2.1 O Processo Ensino Aprendizagem

“A grandeza de um ser humano não está no quanto ele sabe mas no quanto ele tem consciência que não sabe”, Augusto Cury (Cury, 2009).

Para nós ensinar é o ato de transmitir conhecimentos e competências a outrem, pedagógica e didaticamente. Salientamos que o conceito pedagogia e conceito didático são plurívocos e não consensuais.

Segundo Marguerite Altet (Altet, 2000) *“ ensinar é comunicar um conteúdo, (...) a pedagogia é este campo da transformação da informação em saber pela mediatização do professor, (...) a pedagogia debruça-se sobre a articulação do processo ensino-aprendizagem ao nível da relação funcional professor-alunos e da ação do professor em situação; a didática*

estuda a articulação do processo ensino-aprendizagem ao nível da estruturação do saber e da sua apropriação pelo aprendente”.

De acordo com Marguerite Altet (Altet, 2000) o processo ensino-aprendizagem a nível pedagógico recai na relação funcional professor-alunos e da ação do professor. Marguerite advoga que existem dois subsistemas que se relacionam numa situação, os quais são interdependentes mas autónomos, designando-os como o subsistema ensino e subsistema aprendizagem. Classifica o (s) professor (es) e os alunos como processadores, pois gerem energia (Altet, 2000). Cria quatro subsistemas que se inter-relacionam, originando um processo ensino-aprendizagem interativo. A figura 2.1 demonstra a interação entre os quatro subsistemas.



Figura 2.1 Interação dos 4 subsistemas do Processo Ensino-Aprendizagem, adaptado de (Altet, 2000)

O campo da didática analisa a ligação da situação ensino-aprendizagem ao nível da estruturação do saber e da retenção desse saber por parte do aluno. Enquanto o campo da pedagogia analisa a ligação da situação ensino-aprendizagem ao nível da comunicação.

Segundo os estudos do sócio construtivismo (Vygotsky, 2007), é possível elaborar e diferenciar formas de avaliação do processo ensino-aprendizagem, pois o aluno em contacto com pessoas mais experientes vê as suas potencialidades transformadas e são ativados esquemas processuais cognitivos, originando-lhe novas capacidades e conhecimentos.

Vygotsky distingue os esquemas processuais cognitivos em funções cognitivas de baixo nível e de alto nível. As de baixo níveis são uma herança genética e funcionam involuntariamente, enquanto as de alto nível são socialmente adquiridas e controladas voluntariamente, (Subbotsky, 1998) referido por Isabel Chagas, Teresa Bettencourt e Luís Marques (Chagas, Bettencourt, & Marques, sem data).

De referir que a motivação está diretamente relacionada com a teoria cognitiva. A motivação, neste caso, representa o desejo do aluno pelo sucesso ao atingir as melhores notas, sendo que esta, por sua vez, depende dos incentivos fornecidos.

Como tal, a motivação está relacionada com a satisfação, a atenção, a ansiedade, o reforço e o *feedback*. O *feedback* que recebe pode e deve ser mais uma motivação - poderá ser um prémio ou uma simples frase que o ajudem a continuar motivado. O que queremos dizer é que o aluno necessita de ser motivado para prestar atenção e, ao mesmo tempo, aprender e apreender.

O reforço e o *feedback* são focados na teoria *behaviour* proposta por Watson, na qual estes tinham de ser objetivos. Como tal, tinha de estudar o comportamento, que ele aclara como “*um conjunto de respostas objectivamente observáveis que o organismo executa face a estímulos também objectivamente observáveis*”, (Green, s/data). Assim sendo, perante um estímulo há a previsão da reação subsequente e perante a resposta há que determinar o estímulo que a despertou.

Watson defende que os estímulos externos são os responsáveis pelo comportamento, pelo que será possível controlar o comportamento humano manipulando as situações do meio ambiente, (Watson, 1913) referido por (Abrunhosa & Leitão, 2009) . Advoga Watson “ *Dai-me uma dúzia de crianças sadias, bem constituídas, e a espécie de mundo que me é necessário para educá-las, e eu comprometo-me, tomando-as ao acaso, a formá-las de tal maneira que se tornem um especialista da minha escolha, médico, comerciante, jurista, e, mesmo, mendigo ou ladrão, independentemente dos seus talentos, inclinações, tendências, aptidões, assim como da profissão e da raça dos seus antepassados*”, (Abrunhosa & Leitão, 2009).

Skinner concorda com Watson na medida em que defende ser possível controlar o comportamento através de reforços ou castigos. Os reforços podem ser premiados por palavras, ações ou mesmo contrapartidas, (Neto M. B., 2002).

Outro psicólogo *behaviourist*, Bandura, não tão radical quanto Skinner ou Watson, defende que o indivíduo é capaz de aprender através da observação do comportamento dos outros, não havendo contacto direto com o reforço. Considera que entre o estímulo e a resposta existe o espaço cognitivo de cada indivíduo. Ao relacionar a cognição ao behaviorismo, originou outra teoria de aprendizagem conhecida por Cognitivismo Social. (Bandura, 1977), tendo efetuado várias experiências (observações) para fundamentar a sua ideia.

Segundo ele, um comportamento também pode ser adquirido através de “*um conjunto de instruções fornecidas ao indivíduo*”, ou seja, podemos aprender a utilizar um programa computacional através das suas próprias indicações, por um manual com a descrição de todos os processos ou por outro indivíduo. Mantém o reforço, mas subdividi-o em reforço direto e reforço vicariante.

No reforço direto, o indivíduo ouve um elogio perante um determinado comportamento, enquanto no reforço vicariante o elogio é recebido pelo modelo, quer dizer, que um indivíduo ao observar outro a receber um elogio, tenta imitá-lo, pois sente-se incentivado a ter o mesmo comportamento.

Bernard Weiner (Weiner, 1990) indica que as teorias comportamentais focam a motivação extrínseca, que nos recompensa através de prémios, enquanto a motivação intrínseca nos realiza pessoalmente. Não podemos de forma alguma separá-las, pois ambas se interligam quando atingimos um objetivo.

De acordo com esta informação, foram surgindo modelos e quadros relacionados com a motivação. Alguns deles estão relacionados com o desenvolvimento de *software*, podendo abranger a área de jogos computacionais. Segundo Thomas Malone (Malone, 1981) houve a necessidade de criar um quadro teórico para a motivação intrínseca no contexto da conceção de jogos de computador para se obter instruções, que evocassem a motivação. Este quadro é composto por três variáveis: desafio, fantasia e curiosidade.

- O desafio ocorre da incerteza nos resultados provocada pela complexidade, aleatoriedade ou informação incompleta dos ambientes, pois, para os alunos é extremamente motivador depararem-se com um conjunto diversificado de desafios, e obterem um *feedback* concreto e sucinto. Para tal, os critérios têm de estar bem definidos e claros para uma melhor execução das tarefas;
- Por sua vez, a fantasia depende das competências requeridas pela aprendizagem;
- A curiosidade é despertada quando os alunos reconhecem lacunas no seu conhecimento. Citado por (Ribeiro C. R., 2006), (Martins, 2003).

Ao reconhecermos lacunas no nosso conhecimento, somos conduzidos à tão conhecida frase de Sócrates “Eu só sei que nada sei”, (Platão, s/data) e orientados para o que assevera (Cabral, 1990), cita (Martins, 2003), “*O homem vem ao mundo com um impulso lúdico que lhe permite repetir uma acção a que acha graça, tentando vencer as resistências que esse objecto apresenta*”.

O termo “objeto” aplicado por Cabral, para nós, é assumido como tarefa, pois o aluno ao ultrapassar uma dificuldade apresentada por uma tarefa, passa para a próxima e assim por diante, vendo os objetivos realizados em cada tarefa. Para que isso aconteça é obrigatório que o aluno se sinta motivado para tal.

Quando estamos envolvidos numa tarefa que nos motiva, ficamos imersos nela, ao ponto de, por vezes, nos alhearmos do que se passa ao nosso redor, até nos podemos esquecer de comer, dormir, tal é o nosso empenho. Este estado de envolvimento perante uma tarefa é conhecido como a “Teoria de Fluxo”, (Csikszentmihalyi & Csikszentmihaly, 1998). Perante tal, cabe ao professor motivar o aluno para que o mesmo atinja este estado de fluxo. O professor poderá fazê-lo através do método tradicional, pelo *software* educativo ou por ambos. Cabe-lhe essa decisão.

É de realçar que só os aspetos motivacionais não são suficientes para que se possa alcançar o sucesso na realização dos objetivos do aluno. A memória tem um papel muito importante no desenrolar do conhecimento. Pois, não nos recordamos do que aprendemos, significa que essa informação não foi processada ou seja não ficou apreendida e ao não ser apreendida não foi aprendida. Ou seja, a memória não processou nem reteve nenhuma informação.

Existem vários modelos relacionados com a memória, designados por modelos mentais, que são representações da realidade que as pessoas usam para compreender determinados fenómenos. Como descrevem Donald Norman (Norman D. A., 1990), Debre Gentner e Albert Stevens (Gentner & Stevens, 1983) *"In interacting with the environment, with others, and with the artifacts of technology, people form internal, mental models of themselves and of the things with which they are interacting. These models provide predictive and explanatory power for understanding the interaction"*.

Entretanto, Johnson-Laird (Johnson-Laird, 1983) propõe modelos mentais baseados em estruturas cognitivas *"It is now plausible to suppose that mental models play a central and unifying role in representing objects, states of affairs, sequences of events, the way the world is, and the social and psychological actions of daily life"*.

Marco Moreira (Moreira, 2001) faz referência a Johnson-Lari (Johnson-Lari, 1983) em que os modelos mentais e as imagens que são representações de alto nível, são fundamentais para o entendimento da cognição. As representações de alto nível *"podem ser comparadas às linguagens de programação (...) são traduzidas pelos computadores em códigos binários quando compiladas (...) as imagens e os modelos mentais poderiam ser traduzidos pela mente em algum código semelhante ao do código binário (...) o código mentalês"*.

Moreira alerta-nos para o facto de Johnson-Lari (Johnson-Lari, 1983) focar a consciência nos modelos mentais *"a consciência depende de cálculos do sistema nervoso (p. 450). Tais cálculos requerem um cérebro de certo tamanho e complexidade porque a capacidade computacional depende de memória e porque a velocidade computacional depende do tamanho das unidades que podem ser processadas simultaneamente, i.e., do número de processadores que podem ser postos em acção em uma tarefa. Tanto a capacidade como a velocidade computacional são importantes para a consciência. Basta lembrar o uso da linguagem (que requer grande habilidade mental computacional) e o fato de que decisões conscientes são tomadas em tempo real. (...) a mente opera, então, em três níveis computacionais principais: processadores que computam em nível não-consciente, mecanismos que constroem modelos mentais do mundo externo e dispositivos com a habilidade recursiva de construir modelos dentro de modelos"*.

Devido a esta enorme complexidade da aplicabilidade da consciência nos modelos mentais, esta foi de certa forma ignorada até à data.

Para Holland, Holyoak, Nisbett e Thagard (Holland, Holyoak, Nisbett, & Thagard, 1986) o conceito que os modelos mentais são a base para os processos de raciocínio "*Models are best understood as assemblages of synchronic and diachronic rules organized into default hierarchies and clustered into categories. The rules comprising the model act in accord with the principle of limited parallelism, both competing and supporting one another.*"

O estudo dos modelos mentais envolve uma análise pormenorizada, o que permite analisar e perceber como funciona a nossa mente ao interagir com um sistema informático, didático e pedagógico. Para que possamos perceber e entender os modelos mentais, temos de nos debruçar sobre a psicologia cognitiva.

Temos obrigatoriamente de começar por Jean Piaget, pois a sua teoria cognitiva assenta na sistematização de ideias acerca do processo pelo qual o ser humano constrói as estruturas do conhecimento.

Piaget ao observar a evolução de um grupo de crianças até à fase de adolescência concluiu que o desenvolvimento cognitivo se processaria por estádios ou fases. Descreveu esses estádios como, "*etapas qualitativamente diferentes que se desenham no desenvolvimento cognitivo*" (Abrunhosa & Leitão, 2009), (Piaget J. , 1997).

Os estádios são definidos por um intervalo de idades, a sua ordem é sequencial e não se passa para um determinado estádio sem antes se ter percorrido os anteriores. Piaget definiu quatro estádios, que passamos a indicar (Piaget J. , 1997):

- O estádio sensório-motor - corresponde dos zero aos dois anos de idade nesta fase a criança serve-se de perceções e movimentos;
- O estádio pré-operatório - caracteriza-se pelo surgimento da linguagem e pelo início do pensamento e a faixa etária está situada entre os dois e os sete anos de idade;
- O estádio das operações concretas - inicia-se aos sete anos e vai até aos onze anos de idade. Nesta altura, as operações enquanto atividades mentais são concretas e reversíveis (capacidade de regressar mentalmente ao ponto de partida). No entanto,

para que a reversibilidade exista, o pensamento necessita do auxílio dos objetos manipuláveis e das situações vividas;

- O estágio das operações formais - situa-se entre os onze e os quinze anos. O jovem adolescente é capaz de raciocinar dedutivamente a partir de hipóteses e formular hipóteses.

Após a passagem pelos quatro estádios, o indivíduo torna-se adulto e todas as estruturas intelectuais estão definidas.

O grande acérrimo defensor da psicologia cognitiva foi sem margem de dúvida Jerome Bruner. Com ele, houve a reintrodução do conceito da mente.

“Quero começar com a revolução cognitivista como meu ponto de partida. A revolução pretendeu trazer a “mente” de volta para as ciências humanas, depois de um longo Inverno de objectivismo” (Bruner, 2008).

Este movimento cognitivo teve importância na área computacional, pois exerceu associações entre processos mentais e o computador. Segundo Maria Abrunhosa e Miguel Leitão (Abrunhosa & Leitão, 2009) *“o cérebro seria semelhante ao hardware, o software os processos mentais”*. Perante esta situação, Bruner demarca-se, não concorda com a visão homem-máquina. Para ele, a essência do homem é a cognição, e esta é um processo mental, social, biológico e cultural. Compactuar com estas ideias seria limitar o pensamento do homem a um conjunto de organizações e reorganizações sintáticas. Assim, origina o movimento, que designou como Psicologia Cultural.

É neste âmbito, que ele desenvolve vários trabalhos para provar que a mente tem capacidade de atribuir sentido à realidade, atribuindo significado aos eventos que nos rodeiam. De acordo, com o referido *“Bruner prossegue com a convicção de que a mente não é uma capacidade isolada, mas, antes que se constitui com influência da cultura”* (Abrunhosa & Leitão, 2009). Todo o seu trabalho teve particular influência na educação e compartilha com Vygotsky a importância dos fatores sociais como elementos constituintes da mente.

Vygotsky desenvolve conceitos muito relevantes na área da aprendizagem, sendo um deles a Zona de Desenvolvimento Proximal (ZDP). A ZDP relaciona a diferença entre o que a criança consegue realizar sozinha e aquilo que não consegue fazer sozinha, mas que é capaz de

aprender a fazer com a ajuda de outrem mais experiente, (Martins, 2003). A figura 2.2 relaciona a distância entre as duas zonas de desenvolvimento, a Real que indica o nível que a criança atinge sozinha e a Potencial que é o nível que ela atinge com a ajuda de terceiros.

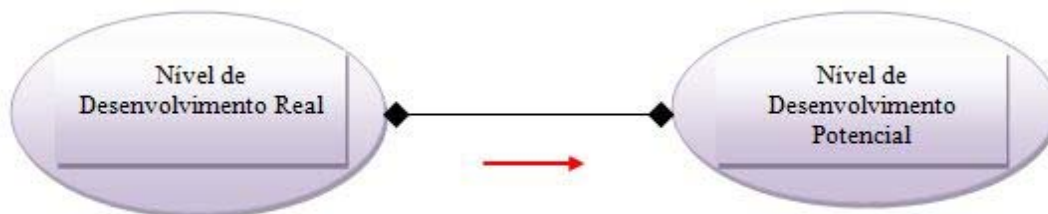


Figura 2.2 ZDP Zona de Desenvolvimento Proximal, Adaptado (Martins,2003)

Para Vygotsky, o desenvolvimento consiste num processo de aprendizagem de uso das ferramentas intelectuais, através da interação social com outros mais experimentados no uso dessas ferramentas, (Palincsar, Brown, & Campione, 1993) referido por Carlos Fino (Fino C. , 2001) e Germano Martins (Martins, 2003). É de salientar que o uso de ferramentas pode ser qualquer material didático, desde o tradicional ao *software* educativo.

No entanto, em 1993, surge-nos uma visão da mente profundamente diferente, pluralista, separada da cognição, que considera que as pessoas diferem umas das outras a nível das forças cognitivas e dos estilos cognitivos. Esta visão baseia-se na ciência cognitiva (o estudo da mente) e na neurociência (o estudo do cérebro), a qual é conhecida pela “ Teoria de Inteligências Múltiplas”, (Gardner, 1995) onde descreve as sete inteligências. São elas:

- Inteligência musical - capacidade de distinguir diferentes sons e ritmos, criar composições por meio de notações musicais ou de reproduzir composições musicais através de instrumentos musicais;
- Inteligência corporal-cinestésica - capacidade de coordenação e controle dos movimentos do corpo e na manipulação de objetos com destreza;
- Inteligência lógico-matemática - está na capacidade de lidar com séries de raciocínios, reconhecer problemas e resolvê-los, explorar relações padrões e categorias através da manipulação de objetos ou símbolos, (Gama, 1998);
- Inteligência linguística - está na capacidade de produzir sentenças gramaticais, através de uma zona específica do cérebro denominada “Centro de Broca”. Investigações recentes demonstram que “ (...) a notar que o que as pessoas

chamavam a área de Broca abrange vastas áreas do lobo frontal", diz (Dronkers, 2000). As imagens observadas dos cérebros Lelong e Leborgne através de ressonância magnética mostraram que nenhum dos cérebros antigos tinham danos que afetassem toda a região agora conhecida por área de Broca, mas os estragos estendiam-se para o interior de outras regiões para além delas (Desconhecido, News of the Wild, 2007);

- *Inteligência espacial - está na capacidade de visualizar mentalmente imagens de objetos, percursos, linhas trajetórias;*
- *Inteligência interpessoal - está na capacidade de perceber e entender os outros. O que os motiva, as intenções, o temperamento, o seu estado de espírito e o que provoca essas alterações;*
- *Inteligência intrapessoal - está relacionada com os aspetos particulares de uma pessoa, a capacidade de discriminar emoções, identificá-las, utilizá-las de forma a orientar o seu próprio conhecimento.*

Para Howard Gardner (Gardner, 1995) a inteligência é um *"potencial biopsicológico"*, ou seja todos os seres humanos têm a capacidade de treinar um conjunto de capacidades intelectuais, a partir da interação com os outros. Assim, qualquer uma destas inteligências pode ser trabalhada e melhorada. No entanto, qualquer indivíduo pode sofrer lesões numa destas inteligências, o que não implica que todas as sofram, e portanto algumas podem ficar inalteradas. Como cita Gardner *"Há idiotas sábios que realizam grandes façanhas de cálculo, mesmo que continuem sendo tragicamente deficientes na maioria das outras áreas"*, (Gardner, 1995).

Segundo Howard Gardner (Gardner, 1995) no processo ensino-aprendizagem uma inteligência serve como conteúdo da instrução e como meio para comunicar esse conteúdo. Perante uma dificuldade do aluno, cabe ao professor descobrir um meio alternativo de transmitir esse conteúdo. Salientemos que qualquer uma das inteligências abrange uma determinada zona do cérebro.

De acordo com Antunes (Antunes C. , Revista Educação, 2005) *"As pesquisas de Gardner representam verdadeiro símbolo educacional contemporâneo, ao sinalizar que o que se descobre sobre a mente humana, constitui não apenas saber académico, mas instrumento de*

acção pedagógica imprescindível”. (...) ” Gardner falava em sete inteligências, mas estudos e pesquisas posteriores elevaram esse número para nove”

Antunes descreve a sua grande paixão pelo cérebro humano. Considera que até aos anos noventa o cérebro era considerado *“como uma caixa preta, intransponível e realce ”*. A partir dessa data, começaram a desenvolver-se vários estudos, originando aquilo que se designou como *“a década do cérebro”*. A importância desses estudos trouxe melhorias e benefícios para a medicina e para a educação, visto que *“o estudo do cérebro é um estudo de educação”*, (Antunes C. , 2008).

Celso Antunes (Antunes C. , Revista Educação, 2005), desenvolveu uma tabela com vista a aplicar a teoria de Gardner, em contexto de sala de aula, a qual se apresenta como tabela 2.1.

Inteligências	Algumas Atividades	Materiais de Ensino	Ação do Docente

Tabela 2.1 Atividades em Contexto de Sala de Aula, (Antunes C. , Revista Educação, 2005)

Posteriormente, houve o desenvolvimento de outra tabela, que o professor pode e deve usar em simultâneo com as tabelas respeitantes ao Anexo A e ao Anexo A1 também desenvolvidas por Antunes. Estas tabelas apresentam as diferentes inteligências e o quanto elas são diferentes se as relacionarmos com as habilidades envolvidas e com a relação existente entre as inteligências. As tabelas apresentam não as sete, mas nove inteligências descobertas *“a posterior”*, (ver anexo A).

A utilização das inteligências múltiplas depende de aluno para aluno, pois cada um tem a sua forma própria de as utilizar. O professor deve ter isso em conta, de forma a colocar situações diferenciadas para cada aluno. Como forma de o auxiliar, Antunes enumerou um conjunto de atividades que estimulariam as várias inteligências, (Antunes C. , Revista Educação, 2005), conforme os anexos B e B1.

2.2 Princípios para a Avaliação do Software Educacional/Educativo

O conceito de *software* educacional e *software* educativo não é linear, pois para alguns, o *software* educativo foi desenvolvido somente para ser utilizado em situações educativas (Ramos, 1998). Para outros, *software* educativo é todo aquele que é usado no processo ensino-aprendizagem (Patrocínio, 1994, citado por (Ramos, 1998)).

Também os autores Marcelino e Mendes (Marcelino & Mendes, 1994) definem programa educativo, como qualquer programa de computador que possa apoiar o processo de ensino-aprendizagem, desde que tenha sido especificamente concebido para tal fim, com objetivos pedagógicos didáticos subjacentes.

Segundo Maria Nascimento (Nascimento, 2007) *“O software educativo (...) é primordial a identificação da concepção teórica de aprendizagem que está subjacente a ele, a sua compreensão enquanto programa educativo, (...) um ambiente virtual de aprendizagem significativa, capaz de gerar um conhecimento novo, com um potencial para prover mudanças no cotidiano escolar e fora dele”*.

De acordo com Carlos Fino (Fino C. N., 1998) *“ o software adequado à teoria é o que dá aos aprendizes acesso a uma exploração diversificada, permitindo-lhe assumir o controlo sobre o curso dos acontecimentos e negociar a sequência das operações envolvidas”*, ou seja, o *software* para ser educativo deve estimular o desenvolvimento cognitivo e metacognitivo.

Para, Fábria Vieira (Vieira F. M., 2001) qualquer *software* que se proponha ser educativo, deve ser desenvolvido a pensar como é que o aluno aprende, ou seja, ele deve oferecer um ambiente interativo que proporcione ao aluno investigar, levantar hipóteses, testá-las e requintar as suas ideias iniciais, pois dessa forma o aluno construirá o seu próprio conhecimento. Para o *software* (um qualquer) oferecer um ambiente interativo deve comunicar com o utilizador, e o utilizador com o *software*.

Para as editoras, todos os seus produtos de *software* educativo são *“fantasticamente educativos”*, que segundo Fino, *“ curto-circuitam o professor e quando executados começam a ensinar a torto e a direito”*, (Fino C. N., 2003).

O contexto educativo que o *software* educacional apresenta é a mais-valia do próprio *software*. Esse contexto segundo Carlos Fino (Fino C. N., 2003) deve permitir um conjunto de atividades, conforme a tabela 2.6 (ver Anexo C).

Segundo Dalvina Ayres (Ayres, 2009) *software* educativo é um *software* que pode ser utilizado para fins educacionais. É uma ferramenta capaz de inovar as ideias dos professores para a prática pedagógica através dos quais os alunos têm acesso a ambientes informatizados e às novas formas capazes de auxiliar no processo de ensino aprendizagem.

O ***“Software educacional, não tendo características próprias de aplicação educativa, depende da acção pedagógica do utilizador para adquirir propriedades educacionais: São exemplo desse software os processadores de texto e os softwares de desenhos gráficos. Já o software educativo, ao ser desenvolvido a partir de perspectivas educacionais específicas, possui um objecto educacional e direcciona-se para o auxílio da aprendizagem de conteúdos e habilidades, através da interface computadorizada.”***, Teixeira (Teixeira, 2006) citam Oliveira, Costa e Moreira (Oliveira, Costa, & Moreira, 2001).

No entanto, todo e qualquer *software* educativo só terá interesse se aprendermos e solidificarmos os nossos conhecimentos com ele. Para que o *software* educativo responda a esses objetivos, é extremamente importante efetuar uma avaliação da qualidade desse *software*. A avaliação da qualidade pode ser vista sob vários prismas desde a avaliação de critérios pedagógicos até à avaliação de critérios ergonómicos. Esta avaliação também nos permite ter uma maior noção das necessidades dos utilizadores perante fatores pedagógicos e de usabilidade do próprio *software*. Assim, houve que analisar as necessidades dos alunos, e as dos professores, para que pudéssemos desenvolver um *software* diferenciável, de acordo com as dificuldades de cada aluno.

Visto que a aprendizagem ativa o desenvolvimento, o *software* educativo deve dirigir-se não para etapas intelectuais já obtidas, mas sim para estágios de desenvolvimento ainda não alcançados pelos alunos. Assim, o *software* deve ter em conta como ponto de partida o nível de conhecimento em que se situa o aluno em relação aos conteúdos, como ponto de chegada, os objetivos reais apontados pelo professor, em contexto de sala de aula, de forma a, que o *software* participe nessa evolução do aluno. Como tal, o *software* deve ser interativo e adaptável a cada aluno, pois as suas necessidades diferem de aluno para aluno.

Segundo Luciano Gamez (Gamez, 1998) os objetivos da avaliação ergonómica de um dispositivo de *software* interativo educacional são:

- “Avaliar as funcionalidades (as necessidades dos utilizadores) ”;
- “Avaliar o efeito do interface sobre os utilizadores, que se traduz na facilidade de aprendizagem do *software* (intuitivamente) e na facilidade e eficiência de uso”.

Walter Cybis (Cybis, 1997) afirma “a avaliação dos efeitos da interface sobre os utilizadores é um trabalho de elaboração que pode ser realizado, sem a participação directa de utilizadores.” Tal afirmação baseia-se no conhecimento ergonómico nas suas diversas formas (modelos cognitivos, critérios e recomendações ergonómicas), tendo como diretrizes as técnicas de avaliação analíticas, sendo estas, a avaliação heurística, exploração cognitiva e listas de verificação, que permitem identificar “*a priori*” os problemas de facilidade de interação do *software*.

No entanto, para obter uma avaliação mais correta sobre as capacidades do *software* é fundamental a participação dos utilizadores, neste caso dos alunos, em todo o processo. Diremos mesmo imperativa! Pois, desta forma acreditamos que o grau de satisfação dos utilizadores irá determinar o potencial deste, passando desta forma a aplicar o conceito de ergonomia *Human Interface Computer- IHC*, perante um *Software* Interativo Adaptável Orientado Para o Ensino.

O *software* interativo educativo deve possuir características particulares e a sua utilidade é avaliada através da sua função didática e pedagógica. Como tal, deve ser inserido num contexto de aprendizagem pré-definida e proporcionar autonomia, cooperação, pensamento crítico, criatividade, descoberta e desenvolvimento do conhecimento.

Portanto, além de ser intuitivo, fácil e eficaz, o *software* interativo educativo deve ser didático. E a didática envolve a implementação de recursos técnicos ou funcionalidades específicas, como refere Walter (Cybis, 1997).

A avaliação ergonómica do *software* interativo educacional deve envolver a definição de métricas objetivas como a eficácia, e subjetivas como a satisfação e motivação dos alunos/professores/educadores no processo de aprendizagem.

2.2.1 Metodologia de Avaliação Pedagógica de Reeves

Este método é considerado uma mistura de uma *checklist*, com a avaliação heurística (que desenvolveremos mais a diante) e ensaios de interação.

Thomas (Reeves, *Systematic Evaluation Procedures for Interactive Multimedia for Education and Training. Multimedia computing: preparing for the 21 st century*, 1994) propõe duas abordagens que se complementam, sendo uma delas composta por catorze critérios pedagógicos (figura 2.3) e a outra por dez critérios (figura 2.4) relacionados com a interface. Os critérios são avaliados através de um gráfico, sobre uma escala com dois sentidos, e em que nas extremidades das setas são colocados os conceitos opostos que representam o critério a avaliar. Nas extremidades à direita são colocados os aspetos positivos e nos da esquerda os negativos. A avaliação é analisada pela interligação desses aspetos. Cada um destes critérios pedagógicos tem significados muito próprios, que passamos a descrever:

- 1 A Epistemologia é dividida em objetiva e construtiva. Sendo cada uma composta por um conjunto de aspetos não mensuráveis, que se apresenta:
 - Epistemologia objetivista estabelece os seguintes aspetos:
 - Conhecimento é independente do saber;
 - Realidade existe independente da existência da experiência;
 - O conhecimento é adquirido de forma objetiva através dos sentidos;
 - A aprendizagem consiste em adquirir conhecimento;
 - O aluno pode ser avaliado através de testes.
 - Epistemologia construtivista estabelece:
 - O conhecimento não tem sentido sem a participação do homem;
 - Embora a realidade exista independentemente, o que conhecemos dela, é individualmente construído;
 - O conhecimento é construído subjetivamente baseado em experiências anteriores e em processo metacognitivo ou de reflexão;
 - O aluno baseia-se na aquisição de estratégias até atingir um objetivo;
 - O aluno pode ser estimulado através de observações, estratégias e diálogos.

- 2 Filosofia Pedagógica é subdividida em filosofia pedagógica instrutiva e construtivista, estas por sua vez, apresentam os seguintes critérios:
- A filosofia pedagógica instrutiva:
 - Enfatiza a importância de metas e objetivos independentes do aluno;
 - Está baseada na teoria do comportamento;
 - O aluno é visto como um agente passivo e recetivo, na aquisição de conhecimento.
 - A filosofia pedagógica construtivista:
 - Realça a intenção, experiência e estratégias metacognitivas do aluno;
 - Permite um conhecimento construído individualmente pelo aluno;
 - Garante um ambiente de aprendizagem o mais rico possível;
 - Difere da instrutiva, já que o aluno é visto como um indivíduo activo, participativo, dotado de conhecimentos pré-existentes, atitudes e motivações.
- 3 A Psicologia Subjacente tem como subcritérios a psicologia subjacente comportamental e a psicologia subjacente cognitiva, tendo estas por sua vez mais subcritérios:
- A psicologia subjacente comportamental:
 - A instrução consiste na moldagem do comportamento desejável obtido através de estímulo respostas.
 - A psicologia subjacente cognitiva:
 - Dá ênfase aos estados mentais internos ao invés do comportamento psicológico;
 - Reconhece que uma ampla variedade de estratégias de aprendizagem deve ser empregue considerando o tipo de conhecimento a ser construído.
- 4 A Objetividade é subdividida em aspetos precisamente evidenciados e não evidenciados:
- Precisamente evidenciados – refere a forma utilizada nos programas tutoriais e de treino;
 - Precisamente não evidenciados – refere a forma utilizada em simulações e ambientes virtuais.
- 5 A sequência instruída divide-se em reducionista e construtivista:

- A sequência instruída reducionista - o aluno deverá ter aprendido previamente os conteúdos, conhecimento minucioso;
 - Na sequência instruída construtivista – o aluno é colocado num contexto realístico, o qual irá requerer soluções a problemas. O apoio é introduzido de acordo com a necessidade individual do aluno, sendo este um elemento ativo.
- 6 A validade experimental é focada em dois critérios o abstrato e o concreto:
- No subcritério abstrato - o aluno é colocado perante situações que não pertencem ao seu mundo real;
 - No subcritério concreto - o aluno está perante situações reais que fazem parte da sua vivência.
- 7 O papel do instrutor é composto por 2 critérios. O provedor de materiais e o agente facilitador, que por sua vez se tornam a decompor noutros papéis:
- Provedor de materiais - está na posse do conhecimento, é considerado “o dono do conhecimento”;
 - Agente facilitador - é visto como uma fonte de orientação e consulta.
- 8 Valorização do Erro. A subdivisão deste critério passa pela aprendizagem sem erro e pela aprendizagem com a experiência:
- Na aprendizagem sem erro - as instruções são organizadas de maneira que o aluno seja induzido a responder corretamente.
 - Aprendendo com a experiência - apoia-se no adágio popular que “a experiência é o melhor professor”. Prevê oportunidades para que o aluno aprenda com seus próprios erros.
- 9 Motivação - este critério é dividido na motivação extrínseca e intrínseca:
- A motivação extrínseca – desenvolve-se fora do ambiente de aprendizagem do aluno.
 - A motivação intrínseca – desenvolve-se dentro do próprio aluno, das suas próprias motivações.
- 10 Estruturação, pode ser alta ou baixa:
- Na estruturação alta - todos os caminhos são previamente determinados.
 - Na estruturação baixa - uma série de opções é oferecida ao aluno, de modo, que ele possa escolher a sua ordem livremente.
- 11 A acomodação das diferenças individuais, dividem-se em não-existentes e multifacetadas:

- Na acomodação das diferenças individuais não-existentes - consideram-se os alunos iguais (homogéneos);
 - Na acomodação das diferenças individuais multifacetada - tem-se em consideração a diferença entre os alunos.
- 12 Controlo do aluno, este critério pode ser não existente ou irrestrito:
- O controlo do aluno não existente - todo o controle pertence ao programa;
 - O Controlo do aluno irrestrito - o aluno decide que secções a estudar, que caminhos seguir e que material utilizar.
- 13 Atividade do utilizador pode ser dividida em atividades de matematização e atividades generativas:
- Em atividades de matematização, os ambientes de aprendizagem pretendem capacitar os alunos a aceder várias vezes a representações do conteúdo;
 - Em atividades generativas os ambientes de aprendizagem limitam o aluno no processo de criação, elaboração ou representação do conteúdo.
- 14 Aprendizagem cooperativa - este critério é dividido em não suportado e integral:
- Não suportado - não permite que os alunos trabalhem em grupos de 2 elementos ou mais;
 - Integral - permite o trabalho cooperativo de modo que os objetivos sejam partilhados pelos alunos, ou seja aluno é beneficiado tanto institucional quanto socialmente.

A figura 2.3 apresenta os catorze critérios pedagógicos de Reeves.

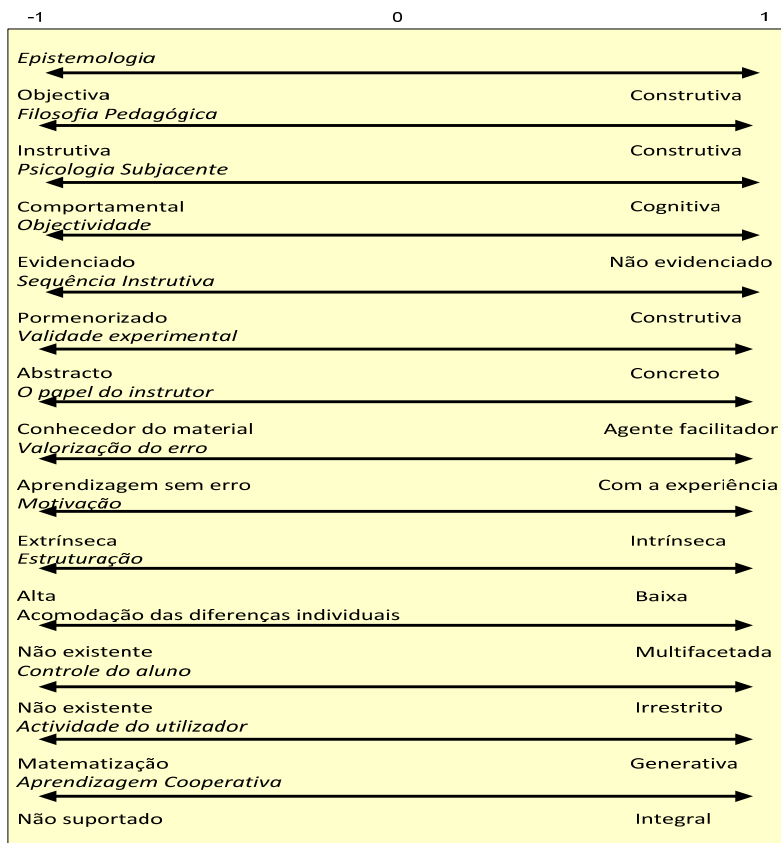


Figura 2.3 Critérios Pedagógicos de Reeves, adaptado (Campos, 1994)

Para Reeves os dez (10) critérios de interface não são suficientes para efetuar uma análise exaustiva de um *software* educacional, no entanto estes contribuem para analisar o ambiente gráfico, textual e auditivo, de interação com o utilizador. Estes critérios têm de se relacionar com os critérios pedagógicos. Apresentam também subdivisões, como se pode verificar:

- 1 O critério facilidade de uso - facilidade de utilização de um determinado *software*.
Tendo como subcritérios:
 - Difícil ou Fácil
- 2 O critério navegação - analisado na perspetiva:
 - Difícil ou Fácil.
- 3 O critério carga cognitiva - analisado como:
 - Não (não existe) ou Intuitiva.
- 4 O critério mapeamento - analisado como:
 - Nenhum (não existe) ou Poderoso.
- 5 O critério realce de ecrã - verifica se os princípios de Realce foram:
 - Violados ou Respeitados.

- 6 O critério compatibilidade espacial do conhecimento - verifica se a informação é:
 - Incompatível ou Compatível com o conhecimento do aluno.
- 7 O critério Apresentação da Informação apresenta-se de uma forma:
 - Confusa ou Clara.
- 8 O critério Integração de multimédia divide-se:
 - Não Coordenada ou Coordenada.
- 9 O critério Estética, está relacionado com a forma como a informação se apresenta, ou seja, se esta satisfaz em termos visuais o utilizar, sendo:
 - Desagradável ou Agradável.
- 10 O critério Funcionalidade Geral, avalia se o *software* é:
 - Não Funcional ou Altamente Funcional.

A figura 2.4 apresenta de uma forma sucinta os critérios de interface de Reeves.

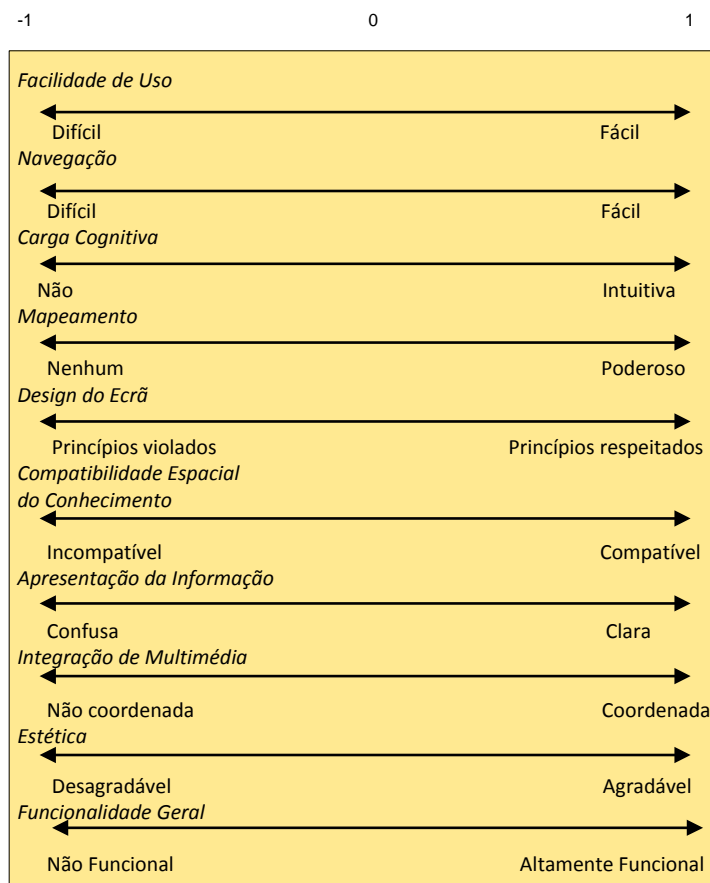


Figura 2.4 Critérios de Interface de Reeves

A figura 2.5 apresenta um exemplo da ligação entre dois critérios pedagógicos de Reeves.

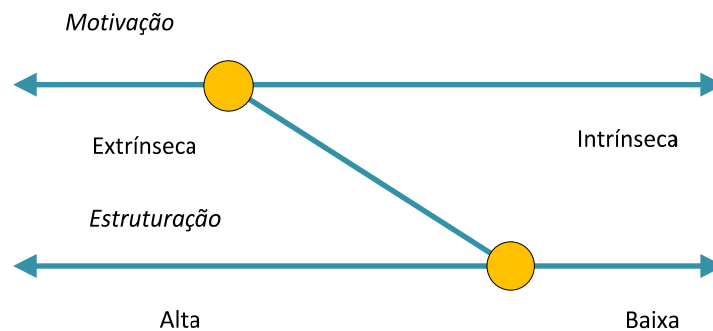


Figura 2.5 Representação da ligação de dois critérios da avaliação pedagógica de Reeves

2.2.2 Metodologia de Avaliação Pedagógica de Bloom

A taxonomia de Bloom referida por Sérgio Bertoldi (Bertoldi S. , 1999), também foi considerada muito importante para nós, visto que esta avaliação aponta para a parte pedagógica de um *software*.

De uma forma geral, esta taxonomia tem como finalidade classificar o que se pretende atingir e os objetivos educacionais, em contexto de sala de aula.

Bloom dividiu os objetivos educacionais em 3 partes:

- Cognitivas;
- Afetivas;
- Psicomotoras.

De salientar que, na área psicomotora Bloom não desenvolveu critérios ou varáveis que pudéssemos ter como referência. No entanto, a partir da sua ideia, tivemos em conta fatores como reflexos, rapidez de movimentos, perceção, controlo de dispositivos, etc.

A área cognitiva apresenta 6 níveis. Cada um destes níveis exige do aluno um nível de abstração elevado. Para que o aluno desenvolva a sua capacidade de raciocínio, estes níveis devem ser introduzidos nas lições e nos testes de avaliação. Pois, só assim o educador/professor pode analisar a evolução do educando/aluno. Passamos a descrever sucintamente os 6 níveis:

- Conhecimento – o aluno deve responder com exatidão a uma matéria que lhe tenha sido lecionada;
- Compreender – o aluno deve ser capaz de interpretar a matéria lecionada;
- Aplicar – o aluno deve ser capaz de aplicar a informação que lhe é fornecida;
- Analisar – o aluno deve analisar a informação fornecida e obter uma conclusão;
- Sintetizar – o aluno deve reunir toda a informação necessária, para apresentar algo novo, que reporta as suas ideias;
- Avaliação – o aluno deve confrontar dados, informações com um conjunto de critérios.

Assim, esta teoria reporta aquilo que os educadores/professores pretendem que os educandos/alunos saibam. A figura seguinte apresenta os seis níveis, que devem ser lidos de baixo para cima.

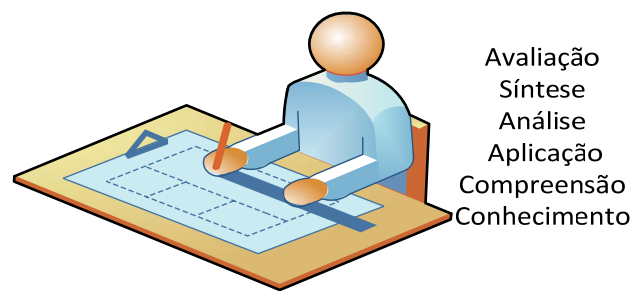


Figura 2.6 Os seis níveis da Taxonomia de Bloom

A tabela 2.2 descreve sucintamente os objetivos a atingir em cada um destes níveis, as capacidades que os alunos devem adquirir e os verbos que se devem aplicar em cada um dos níveis.

Níveis	Objetivos	Capacidades a Adquirir
CONHECIMENTO	Lembrar informações sobre: factos, datas, palavras, teorias, métodos, classificações, regras, critérios, etc.	Definir, descrever, distinguir, identificar, rotular, listar, memorizar, ordenar, reconhecer, reproduzir etc.
COMPREENSÃO	Entender a informação, captar o seu significado, utilizá-la em contextos diferentes	Classificar, converter, descrever, discutir, explicar, generalizar, identificar, inferir, interpretar, prever, reconhecer, redefinir, seleccionar, situar, traduzir, etc.
APLICAÇÃO	Aplicar o conhecimento em situações concretas	Aplicar, construir, demonstrar, empregar, esboçar, escolher, escrever, ilustrar, interpretar, operar, praticar, preparar, programar, resolver, usar, etc.
ANÁLISE	Identificar as partes e suas inter-relações	Analisar, calcular, comparar, distinguir, examinar, experimentar, testar, esquematizar, questionar, etc.
SÍNTESE	Combinar partes não organizadas para formar um todo	Compor, construir, criar, desenvolver, estruturar, formular, modificar, montar, organizar, planear, projetar, etc.
AVALIAÇÃO	Avaliar o valor do conhecimento	Avaliar, criticar, comparar, defender, detetar, escolher, estimar, explicar, julgar, seleccionar, etc.

Tabela 2.2 Taxonomia de Bloom, Níveis Cognitivos e seus objetivos fonte: (Telles, 2005)

A área afetiva da taxonomia de Bloom está relacionada com o comportamento, responsabilidade emoção, valores e atitudes da parte dos alunos. Seguindo o raciocínio anterior, foi desenvolvida uma tabela que nos indica os objetivos e as capacidades a adquirir pelos alunos.

Níveis	Objetivos	Capacidades a Adquirir
RECEPÇÃO	Dar-se conta de fatos, predisposição para ouvir, atenção seletiva	Dar nome, descrever, destacar, escolher, identificar, localizar, manter, perguntar, responder, seguir, selecionar, usar etc.
RESPOSTA	Envolver-se (participar) na aprendizagem, responder a estímulos, apresentar ideias, questionar ideias e conceitos, seguir regras.	Adaptar-se, ajudar, apresentar, desempenhar, discutir, escrever, estudar, falar, responder, selecionar, etc.
AVALIAÇÃO	Atribuir valores a fenómenos, objetos e comportamentos.	Aproximar, completar, convidar, demonstrar, diferenciar, dividir, explicar, iniciar, justificar propor etc.
ORGANIZAÇÃO (DE VALORES)	Atribuir prioridades a valores, resolver conflitos entre valores, criar um sistema de valores	Adaptar, alterar, combinar, comparar, completar, concordar, defender, explicar, formular, generalizar, identificar, integrar, inter-relacionar, modificar, ordenar, organizar, preparar, relacionar, sintetizar etc.
INTERNALIZAÇÃO	Adotar um sistema de valores, praticar esse sistema	Agir, cooperar, desempenhar, generalizar, influenciar, integrar, modificar, ouvir, propor, questionar, resolver, revisar, ser ético, verificar etc.

Tabela 2.3 Taxonomia de Bloom, Área Afetiva e seus objetivos, fonte: (Telles, 2005)

Com o passar dos anos, verificou-se que o ensino e a aprendizagem envolvem mais do que o simples raciocínio, pois temos de ter em conta o nível sócio cultural dos alunos e os factores psicológicos inerentes a eles próprios.

Perante tal, alguns estudiosos debruçaram-se sobre esta Taxonomia, nomeadamente Robert Marzano (Marzano & Kendall, Designing a New Taxonomy of Education Objectives., 2000) que nos exprime a ideia *“Sendo esta Taxonomia hierárquica, então a capacidade mais alta deve ser composta por todas as outras abaixo dela”*.

Segundo Robert Marzano (Marzano & Kendall, Designing a New Taxonomy of Education Objectives., 2000) como esta situação, não se verifica na Taxonomia de Bloom, é necessário retificá-la e assim apresenta-nos um novo conceito de Taxonomia que expõe duas dimensões, de acordo com a figura 2.7.

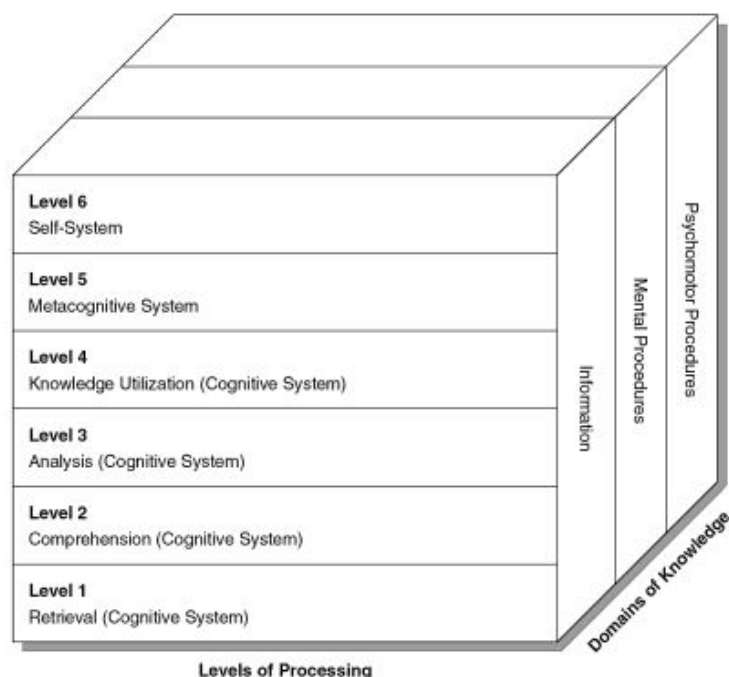


Figura 2.7 Nova Taxonomia, (Marzano & Kendall, Designing & Assessing Educational Objectives, Applying the New Taxonomy, 2008)

Esta nova taxonomia apresenta duas dimensões. A primeira do domínio do conhecimento, é composta por três áreas: informação, processos mentais e processos psicomotores. A segunda dimensão é representada pelos seis níveis de processamento e cada nível é composto por diversas variáveis operacionais, cujo significado é exposto nos Anexos C e C1.

A diferença mais significativa entre a Taxonomia de Bloom e a Nova Taxonomia consiste no facto de o conhecimento ser analisado numa outra perspetiva. Pois a Nova Taxonomia separa os diferentes tipos de conhecimento dos processos mentais que atuam sobre eles, enquanto a Taxonomia de Bloom inclui as várias formas de conhecimento com as diversas operações mentais que operam sobre o próprio conhecimento, o que confunde o objeto de uma ação com a ação em si.

A Nova Taxonomia evita esta situação ao requerer três domínios do conhecimento, que são trabalhados por três sistemas de pensamento e dos seus elementos, (Marzano & Kendall, Designing & Assessing Educational Objectives, Applying the New Taxonomy, 2008). A figura 2.8 retrata a situação.

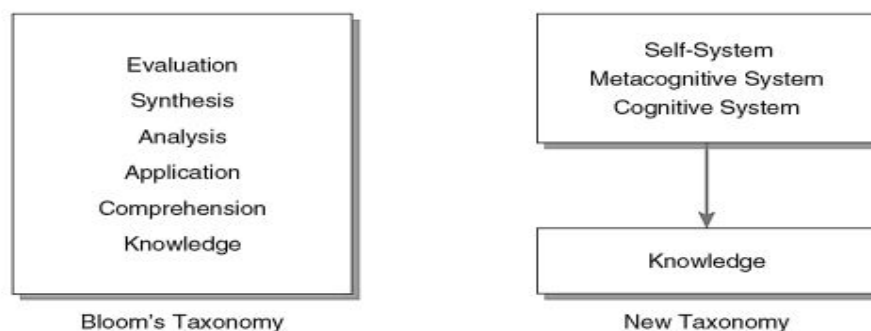


Figura 2.8 Comparação entre as duas Taxonomias, (Marzano & Kendall, Designing & Assessing Educational Objectives, Applying the New Taxonomy, 2008)

Marzano também chama a atenção para o facto de Bloom identificar tipos específicos de conhecimento dentro do próprio conhecimento que podem ser recordados ou simplesmente reconhecidos, o que torna a Taxonomia Bloom uma teia de aranha, pois todos os processos mentais atuam sobre o conhecimento. Através da Nova Taxonomia a situação é invertida, porque o auto conhecimento, os aspetos metacognitivos e os aspetos cognitivos atuam em conjunto para o conhecimento.

No entanto, “(...) na Nova Taxonomia estão embutidos em muitos componentes dos níveis 4, 5 e 6 da Taxonomia de Bloom”, (Marzano & Kendall, Designing & Assessing Educational Objectives, Applying the New Taxonomy, 2008).

No ano 2001 Lorin Anderson (Anderson & Krathmohl, 2001) argumenta “que actividades complexas do ensino requerem o uso de diversas capacidades cognitivas! Tal não se verifica na Taxonomia de Bloom”. Ao efetuar este estudo comparativo, Anderson apresenta-nos algumas alterações, de acordo com a tabela 2.4.

Taxonomia de Bloom	Nova Taxonomia por Anderson
Conhecimento	Lembrança de ou recordar de
Compreensão	Entender
Aplicação	Aplicar
Análise	Analisar
Síntese	Avaliar
Avaliação	Criar

Tabela 2.4 Estudo comparativo da Taxonomia de Bloom e da Nova Taxonomia de Anderson

Leslie Wilson retrata a situação referida anteriormente, através da figura 2.9.

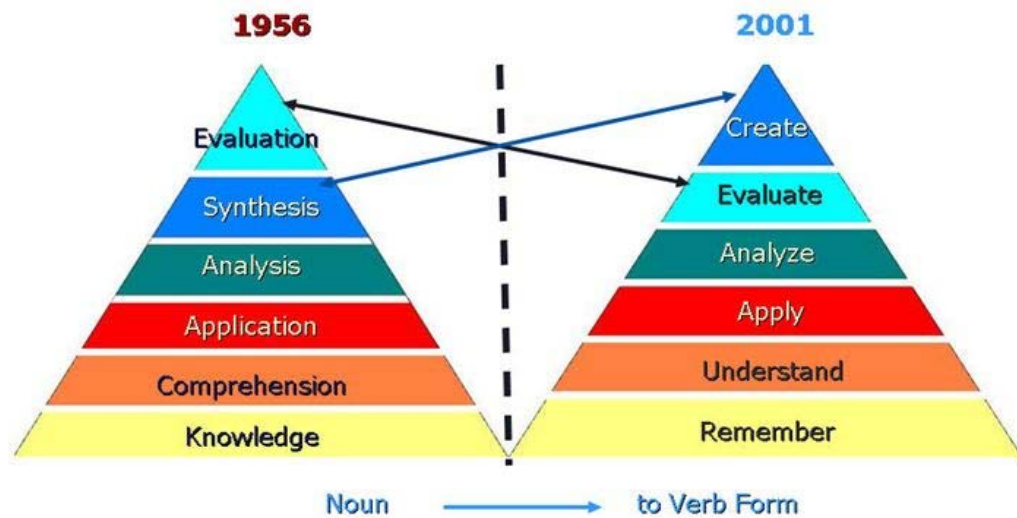


Figura 2.9 Comparação entre a Taxonomia de Bloom e de Anderson (Wilson, 2006)

O grande foco de alteração existe na troca dos últimos dois níveis do topo da pirâmide.

Como tal, podemos concluir que o aluno ao longo do seu percurso estudantil, assimila e retém um conjunto de informação que irá aplicar e desenvolver em novas situações.

Para Lorin Anderson (Anderson & Krathmohl, 2001) o aluno, no final da aquisição de conteúdos deve ser capaz de criar enquanto está a ser avaliado. Essa avaliação fica ao critério do professor, de acordo com os conteúdos lecionados.

Claro que os objetivos que os professores pretendem que os alunos atinjam, dependem do desempenho de cada aluno e do esforço por ele aplicado. Assim, segundo Corrêa (Corrêa, 2003), *“O esforço mental experimentado por um indivíduo é uma função complexa e pessoal, este depende da sua motivação e de outros factores singulares ao indivíduo”*.

De acordo com a figura 2.10, o esforço mental é considerado uma variável externa, que depende das exigências de uma determinada ação, à qual o aluno tem de responder de uma maneira eficaz. Essa resposta depende interação entre fatores emocionais, psicológicos e cognitivos de um aluno.

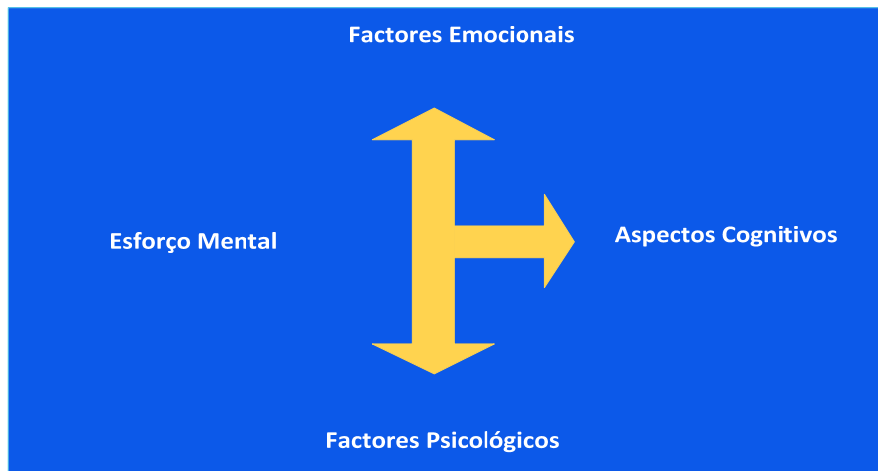


Figura 2.10 Esforço Mental - adaptado (Corrêa, 2003)

É de recordar que, para Alain Wisner (Wisner, 1994) todas as atividades de um ser humano incluem três aspetos: físico, cognitivo e psíquico. Cada um deles determina uma sobrecarga. No nosso caso debruçamo-nos sobre os aspetos cognitivos. Para Wisner estes dependem de três variáveis, sendo elas:

- Percepção - resultado do processamento do estímulo sensorial, originando um resultado, em que a experiência vivida, as relações, o contexto e a memória desempenham um função extremamente importante Itiro lida (lida, 2005), Michael Levine, Jeremy Shefner (Levine & Shefner, 2000) e June Griève (Griève J. , 2000);
- Memória – Função cognitiva responsável pela receção de informações, preparação e armazenamento destas, associando essa informação para uso posterior (Griève, 2006);
- Atenção e vigília – indicam como um indivíduo responde a um estímulo ou processo em seu meio. Nesta situação, estamos perante um processo cognitivo dinâmico, (Griève, 2006).

Assim e de acordo com estas três variáveis, direcionamo-nos para a Taxonomia de Bloom revista, visto que a evolução do aluno permite-lhe criar e desenvolver novos conceitos.

Quanto mais analisávamos estas metodologias, mais tínhamos a sensação de que ainda sabíamos muito pouco ou nada. Por isso, continuávamos a pesquisar, de modo a recolher o máximo de informação. Durante a pesquisa deparamo-nos com a Taxonomia SOLO, que consideramos muitíssimo interessante para este estudo.

2.2.3 Taxonomia Solo

Esta Taxonomia é denominada *Structure of Observing Learning Outcome* (SOLO) e foi desenvolvida por Biggs e Collis (Biggs & Collis, 1982). A Taxonomia SOLO baseia-se nos princípios de Piaget. Os seus autores concordam com as estruturas existentes nos estágios de Piaget, mas renomeiam-nos como modos de pensamento. Para eles, os modos são específicos para cada domínio de conhecimento; um modo não substitui outro, mas inter-relaciona-se com todos os outros já existentes.

Esta Taxonomia identifica níveis de formalização do pensamento, o que permite identificar níveis hierárquicos do conhecimento, que envolvem estruturas cognitivas cada vez mais complexas, estando estas relacionadas com os conteúdos assimilados, a partir de técnicas e ferramentas desenvolvidas para esse fim.

Como tal, permite avaliar a qualidade da aprendizagem dos alunos, e os níveis alcançados por eles. Para Amanda Amantes e Oto Borges (Amantes & Borges, s.d.) *“Os níveis são ascendentes e dizem respeito às relações estabelecidas entre os diversos elementos e o conteúdo apreendido”*.

Para John Biggs (Biggs J. , Bulletin of the Hong Kong Psychological Society, 1992) o conhecimento adquirido pelo aluno pode ser refletido numa aprendizagem profunda ou superficial. Quando a aprendizagem é profunda, envolve processos cognitivos de alto nível, existe um saber inerente sobre o conteúdo lecionado. Enquanto na aprendizagem superficial o aluno limita-se a reproduzir com exatidão o que lhe foi ensinado. No entanto, em qualquer destas aprendizagens o aluno pode mostrar no mesmo modo níveis de complexidade diferente.

A Taxonomia SOLO apresenta uma hierarquia, onde cada nível é construído a partir do anterior (Biggs J. , Teaching for Quality Learning at University, 2003). Ou seja, os níveis surgem em ordem crescente de complexidade estrutural, (Biggs & Collis, 1982), (B.Dahl, 2008). Sendo assim, a hierarquia da Taxonomia SOLO foi subdividida em cinco níveis, sendo eles: Nível Pré-estrutural (*Pre-Structural Level*), Nível Uni-estrutural (*Uni-Structural_Level*), Nível Multi-estrutural (*Multi-Structural Level*), Nível Relacional (*Relational Level*), Resumo Alargado (*Extend Abstract Level*), conforme se pode ver na tabela 2.5.

Hierarquia dos Cinco Níveis da Taxonomia SOLO	
Nível Pré-estrutural (<i>Pre-Structural Level</i>)	O aluno responde de uma forma inadequada, está distraído e confuso, SOLO1
Nível Uni-estrutural (<i>Uni-Structural_Level</i>)	O aluno só consegue equacionar um aspeto, mas efetua as ligações corretas, SOLO 2
Nível Multi- estrutural (<i>Multi-Structural Level</i>)	O aluno consegue lidar com vários aspetos, mas não consegue relacionar a informação, como descreve <i>“The student sees the many trees, but not the forest”</i> , SOLO 3
Nível Relacional (<i>Relational Level</i>)	O aluno consegue perceber a relação entre os vários aspetos apresentados, formando assim um todo, ou seja nesta altura o aluno vê como muitas árvores formam uma floresta, <i>“ The understanding forms a structure and now he does see how the many trees form a forest”</i> SOLO 4
Resumo Alargado (<i>Extend Abstract Level</i>)	O aluno tem um raciocínio bem estruturado e consegue visualizar toda a informação sob várias perspetivas e criar situações novas em diversas áreas, SOLO 5

Tabela 2.5 Hierarquia dos cinco níveis da Taxonomia Solo (Biggs J. , Teaching for Quality Learning at University, 2003)

Perante a descrição dos níveis da Taxonomia SOLO, o aluno cria competências perante a ascensão hierárquica do SOLO. Assim, podemos dizer que o aluno passa da aprendizagem superficial, para a aprendizagem profunda. Quando o aluno se encontra ao nível da aprendizagem superficial situa-se entre o SOLO 2 e 3. No momento em que passa para a aprendizagem profunda, pode situar-se entre o SOLO 2 e 5 ou no SOLO 4 e 5. O que implica uma progressão do aluno em termos quantitativos e qualitativos.

Em cada um destes níveis do SOLO, existem um conjunto de verbos proposto por John Biggs (John Biggs, 2007) pelo qual identificam esse SOLO, como se pode verificar no anexo D.

Podemos verbalizar que através da Taxonomia SOLO existe uma planificação detalhada do que se pretende em cada módulo de conteúdos, quais os objetivos que se aspira que os alunos atinjam e a sua evolução cognitiva.

2.2.4 Metodologias de Avaliação Pedagógica do Software Educativo

“O bom professor deverá utilizar livros e recursos de modo adequado tal como a sua própria perspicácia, simpatia e magnetismo”, (Thorndike, 1912).

Ao falarmos de metodologias de avaliação pedagógica do *software* educativo e colocamos a nós próprios algumas questões, que passamos a citar:

- 1ª Deve o professor participar nessa avaliação?
- 2ª Como é que os professores são capazes de avaliar com clareza, precisão e eficácia o *software* educativo?
- 3ª Devem utilizar grelhas de avaliação? Quais?

Para alguns, os professores devem participar na avaliação; Fino referencia a proposta de Michael Golebiewski *“Evaluatting Software for Educators”*, a qual apresenta as razões pelas quais os professores devem participar na avaliação do *software* educativo, *“As the amount and variety of educational software grows, there is a need for it to be assessed for the suitability for its intended purposes. Teachers need to be able to properly evaluate that a software program will meet their curriculum objectives, as well as, evaluate how software can impact the learning experience. The results from using computer assisted instruction in our classrooms and learning labs will only be as good as the software chosen to be available to our students (...) In essence, a teacher who is unable to evaluate a software program as “good” or “bad” is not going to be able to integrate the computer as a viable resource in the classroom”*. (Fino C. N., 2003)

Para Webber, Boff e Bono (Webber, Boff, & Bono, 2009) quem avalia *software* educativo deve ter *“experiência no ensino e no uso de tecnologias de informática na educação (...) Necessita-se de facto de critérios formais que aliados à experiência e percepção do avaliador permitam que uma análise correcta em termos de aplicabilidade adequação sejam realizadas”*.

Para Fernando Costa (Costa F. A., 1999), *“A participação dos professores na avaliação de produtos multimédia é considerada como um mecanismo de enriquecimento sucessivo dos próprios professores nomeadamente através da reflexão sobre o seu potencial em termos de exploração pedagógica”*.

Conforme Fernando da Costa (Costa F. A., 1999), o *software* multimédia educativo deve ser avaliado segundo o seu valor pedagógico. Desta forma O Projecto Pedactice (1998-2000) *“Educational Multimedia in Compulsory School: From Pedagogical Assessment to Product Assessment”*, teve como um dos objectivos validar critérios de avaliação do *software* educativo, de forma que se fosse possível retirar conclusões acerca do seu valor pedagógico. Assim, neste âmbito *“é oportuno que a avaliação específica de software multimédia e* (Costa F.

A., 1999) *educativo contemple não apenas o produto em si mesmo, mas se estenda à análise do produto em situação real de utilização em contexto de ensino/aprendizagem e à reflexão sobre o seu contributo em termos de efectividade na aprendizagem*”

Segundo Fernando da Costa (Costa F. A., 1999) foi construído um modelo de avaliação que tem como objetivo, apresentar objetos de avaliação. Este tem como referência uma avaliação que, embora globalizante em termos do objetivo último da “eficácia” e em termos de resultados de aprendizagem, permita refletir sobre aspetos pedagógicos e didáticos e forneça informação diferenciada relativamente às características e potencial de cada produto analisado, de acordo com a figura 2.11.

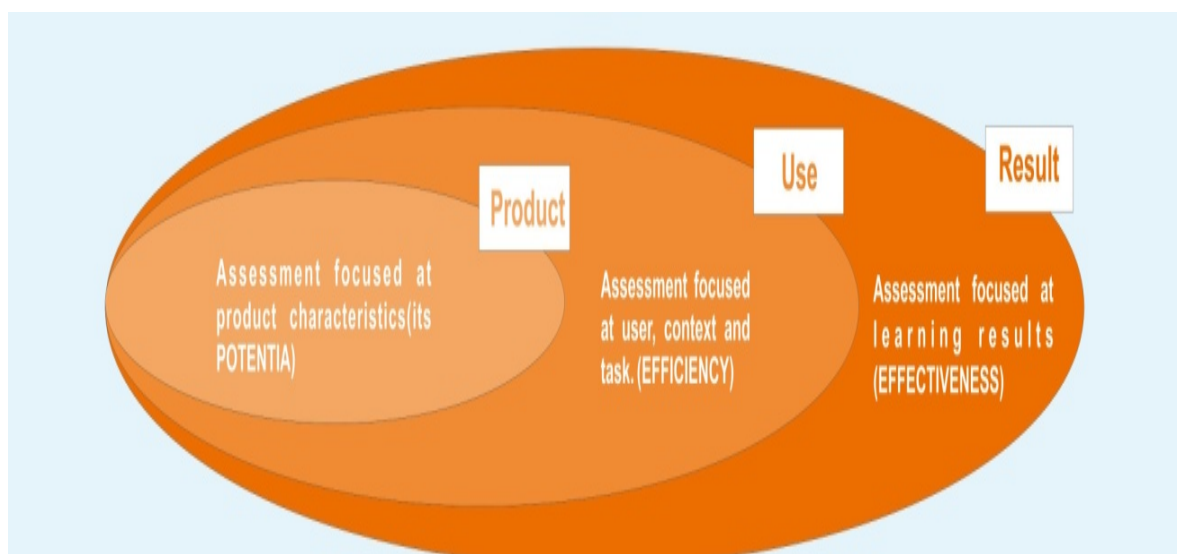


Figura 2.11 Objetivos a Avaliar (Costa F. A., 1999), adaptado (E.Rodrigues, J.Moreno, & D.Grilo, 2009)

Para Fernando Costa (Costa F. A., 1999), na Avaliação Centrada nas Características do Produto, é conveniente efetuar uma avaliação holística, que tem como principal meta de referência o contexto mais amplo de aprendizagem em que um determinado produto multimédia vai ser utilizado (Bevan N. , Measuring Usability as Quality of Use, 1995), (Gálvis, 1997). Assim, é possível focar cada produto em si mesmo e sobre ele manifestar sinais de valor independentemente da utilização propriamente dita ou da sua relevância em termos de aprendizagem.

Este tipo de análise fornece elementos esclarecedores sobre o produto numa perspetiva pedagógica que Fernando Costa (Costa F. A., 1999) designou como o “*potencial pedagógico do produto*”

Para Fernando Costa (Costa F. A., 1999) a avaliação centrada nos resultados da aprendizagem, depende da natureza da aprendizagem e dos requisitos da tarefa (objetivos didático), dos conhecimentos e capacidades cognitivas dos utilizadores e dos fatores contextuais e situacionais. O que lhe proporcionou desenvolver um quadro de referência para a avaliação a partir do esquema que se mostra na figura 2.12.



Figura 2.12 Elementos Estruturantes da Avaliação de Software (Costa F. A., 1999), adaptado (E.Rodrigues, J.Moreno, & D.Grilo, 2009)

Segundo Fernando Costa (Costa F. A., 1999) estes elementos são entendidos como:

- *“Questões – Conjunto de questões que permitem explicar cada componente e delimitar o respectivo conteúdo, nomeadamente, que tipo de comunicação é estabelecido entre a aplicação e o utilizador? Como se processa a interação entre a aplicação e o utilizador? “;*
- *“Critérios – Conjunto de formulações de carácter afirmativo que permitem a diferenciação em cada uma das categorias a avaliar e que permitirão a posterior emissão de juízos valorativos” (Costa F. A., 1999), nomeadamente a facilidade de aprendizagem que o software deve ter;*
- *“Indicadores – Conjunto de evidências observáveis que fornecem a informação necessária à operacionalização dos critérios” (Costa F. A., 1999), designadamente o grau de participação que um determinado software exige do utilizador, como a interatividade.*

Como resultado do seu estudo Costa apresentou quatro quadros de avaliação de *software*, que funcionam em conjunto (Costa F. A., 1999) . Denominou esses quadros (tabelas) de classificação da seguinte forma:

- O primeiro quadro (Quadro I) – direcionado para os requisitos técnicos (ver anexo E);
- O segundo quadro (Quadro II) – orientado para o conteúdo e aspetos pedagógicos da aplicação (ver anexo F);

- O terceiro quadro (Quadro III) – engloba a área técnica a nível de interatividade, interface gráfica e ferramentas de exploração (ver anexo G);
- O quarto (Quadro IV) – abrange aspetos dirigidos para a usabilidade da aplicação (ver anexo H).

Ao utilizar a informação sistematizada sobre cada um destes elementos (Costa F. A., 1999), obteve a informação adequada para formular uma opinião crítica e fundamentada sobre um determinado produto em análise.

Elsa Rodrigues ao observar um grupo de crianças e professores em contexto de sala de aula do ensino especial, no uso de um conjunto de *software* educativo desenvolveu uma ficha de avaliação para o mesmo, (Rodrigues E. , A Contribution for the Evaluation of Educational Interactive Software, 2002).

Esta ficha está dividida em três partes. A primeira parte corresponde à classificação do *software* educativo e educacional, sendo esta representada por uma grelha. Tal como se apresenta a tabela 2.6.

Domínio	Matemática	Escrita	Leitura	AVD	Estudo do Meio	Puzzle e Sequência
Tipo de SW						
Tutoriais						
Programas Práticos						
Jogos Educativos						
Investigação						
Simulação						
Programas de Resolução de Problemas						

Tabela 2.6 Descrição da grelha classificativa, (Rodrigues E. , A Contribution for the Evaluation of Educational Interactive Software, 2002)

A linha da identificação dos domínios, referencia a matemática (operações básicas, soma, subtração, multiplicação, divisão), a escrita (verificação do método de ensino utilizado), a

leitura (verificação do método de ensino está a utilizar), atividades da vida diária (AVD), estudo do meio (representação do meio físico), puzzle e sequência (raciocínio lógico e memorização).

A primeira coluna referencia os tutoriais, programas práticos, jogos, educativos e educacionais, investigação, simulação, programas de resolução de problemas.

A segunda parte da ficha - consiste na classificação dos produtos que servem de apoio para realizar tarefas, tais como: escrever, desenhar, armazenar informação, etc. De acordo com o seguinte *software* de ferramenta, representado na tabela 2.7.

<p>Processamento de Texto:</p> <p>Microsoft Word (disponível para PC) Claris(Disponível para Mac)</p> <p>Folhas de Cálculo:</p> <p>Excel (disponível para PC) Claris (disponível para Mac)</p> <p>Base de Dados:</p> <p>Access (disponível para PC) Claris (disponível para Mac)</p> <p>Edição Electrónica:</p> <p>Utilitários:</p> <p>PowerPoint - apresentações (disponível para PC) Software de Scanner Win Zip – compactador e descompactador (disponível para PC)</p>

Tabela 2.7 Software de Ferramentas, (Rodrigues E. , A Contribution for the Evaluation of Educational Interactive Software, 2002)

A terceira parte da ficha (ver apêndice A) corresponde à descrição detalhada do *software*. Essa descrição deve contemplar:

- A compensação - permitir premiar o esforço do utilizador;
- A interface - permitir o estabelecimento da relação entre o utilizador e a aplicação;
- A animação – permitir representações com a técnica de desenho animado;
- O controlo da apresentação do programa - permitir a alteração do grau de dificuldade da transmissão e manipulação da informação;

- A avaliação do aluno - permitir armazenar informação acerca da "pontuação" dos alunos que utilizam o produto;
- A capacidade de motivação – permitir a existência de mecanismos que motivem o aluno a continuar a utilizar o produto mesmo que as suas respostas não sejam, à partida, as mais corretas.

Desta forma, temos para cada produto a identificação do *software*, as características do hardware, aspetos pedagógicos, requisitos técnicos, características do interface, interatividade, e ferramentas de exploração descrição da aplicação.

2.3 Metodologias de Avaliação da Qualidade de Software

De salientar que este subcapítulo não tem como objetivo demonstrar a temática da qualidade de *software* aos modelos de qualidade em engenharia de software, mas sim de estabelecer a ligação entre qualidade e satisfação do utilizador.

Para que se possa avaliar o *software*, há que ter em conta todo o processo desde o seu início até ao seu término. Portanto, julgamos necessário definir o que é a qualidade de *software*, visto que foi de extrema importância encontrar fatores que contribuíssem para o desenvolvimento de *Software Interativo Adaptável Orientado para o Ensino*. Houve pois a necessidade de estudar determinados modelos de avaliação da qualidade de *software*, para que pudéssemos desenvolver o nosso modelo.

No dicionário de língua Portuguesa qualidade é definida como, propriedade ou condição natural de uma pessoa ou coisa que a distingue das outras; atributo; característica; predicado.

Segundo a *International Organization for Standardization (ISO)*, qualidade “é a totalidade das características de uma entidade que permitem satisfazer necessidades especificadas ou implícitas”.

Surgem vários autores a definir qualidade de *software*. Entre eles, Gilda de Campos (Campos, 1994) cita Michael Coleman e Tom Manns (Manns & Coleman, 1988) definem-na como um modelo sistemático de ações necessárias a fim de assegurar que o módulo ou projeto estão de acordo com os requisitos técnicos estabelecidos. No entanto, sugerem uma

definição mais alargada, a fornecida pelo AINSI/IEE (1981) “*Modelo planeado e sistemático de todas as acções necessárias para assegurar que o software opere de acordo com os requisitos técnicos*”.

Roger Pressman (Pressman, 1992) afirma, que “*o controle de qualidade de software é um conjunto complexo de factores que podem variar de acordo com as diferentes aplicações e de acordo com os utilizadores que o requisitam*”.

Contudo, a qualidade de *software*, abrange tanto a avaliação dos processos de desenvolvimento, como a avaliação do produto final. Assim, para que se possa avaliar a qualidade de *software* nestas fases, é necessário utilizar parâmetros objetivos que fundamentem juízos e forneçam argumentos para avaliações posteriores, e se utilizem metodologias e técnicas que tornem estes processos organizados e metódicos, de forma a reduzir custos e esforços aumentando assim a sua fiabilidade (Herbert & Price, 1995).

A qualidade do produto final de *software* pode e deve ser avaliada aquando da utilização de determinados modelos de qualidade, os quais são constituídos a partir de características a serem observadas e comensuradas através de métricas objetivas.

Para avaliar a qualidade de *software* alguns modelos foram desenvolvidos, nomeadamente o modelo de Boehm (Boehm, Brown, Kaspar, Lipow, Machleod, & Merritt, 1978), o modelo de MacCall (MacCall, Richards, & Walters, 1997), o modelo Ergonómico de Dominique Scapin e Christian Bastien (Scapin & Bastien, 1993), modelo ergonómico *Human Interface Computer* (HCI), o modelo *User Interface Design* (UID) e as Normas ISO, que iremos descrever nas próximas secções.

2.3.1 Descrição dos Modelos McCall e Boehm

Ambos os modelos têm em vista o produto final. Estes identificam atributos chave, os quais são designados por factores de qualidade, a exemplo, usabilidade, fiabilidade e eficiência. Mas existe a necessidade de detalhar estes factores de qualidade, para uma mensura mais correta. Assim, eles são decompostos em níveis mais baixos, os designados critérios de qualidade. Exemplo no modelo de Boehm, o fator fiabilidade é composto pelos critérios complemento, precisão e consistência, (Campos, 1994).

Por sua vez, medir os critérios de qualidade, implica outro nível de decomposição. Nestes caso os critérios são associados por um conjunto de subníveis de atributos, os quais são designados por métricas de qualidade, (Campos, 1994).

Modelo de MacCall

Este modelo é destinado à engenharia de *software*, com o objetivo de ser utilizado durante o processo de construção do sistema. O modelo identifica três áreas de trabalho de Software:

- Comportamento do produto – requer que o *software* seja aprendido facilmente, seja eficiente e que os resultados sejam aqueles que o utilizador espera.
- Revisão do produto – está relacionada com a correção dos erros e com a adaptação do sistema.
- Transição do produto – embora possa não ser relevante em todas as aplicações, a sua transição para sistemas distribuídos e o rápido aumento na troca de *hardware* fazem com que esta área tenha a sua importância.

A figura 2.13 representa o modelo de MacCall, com os seus critérios e subcritérios.

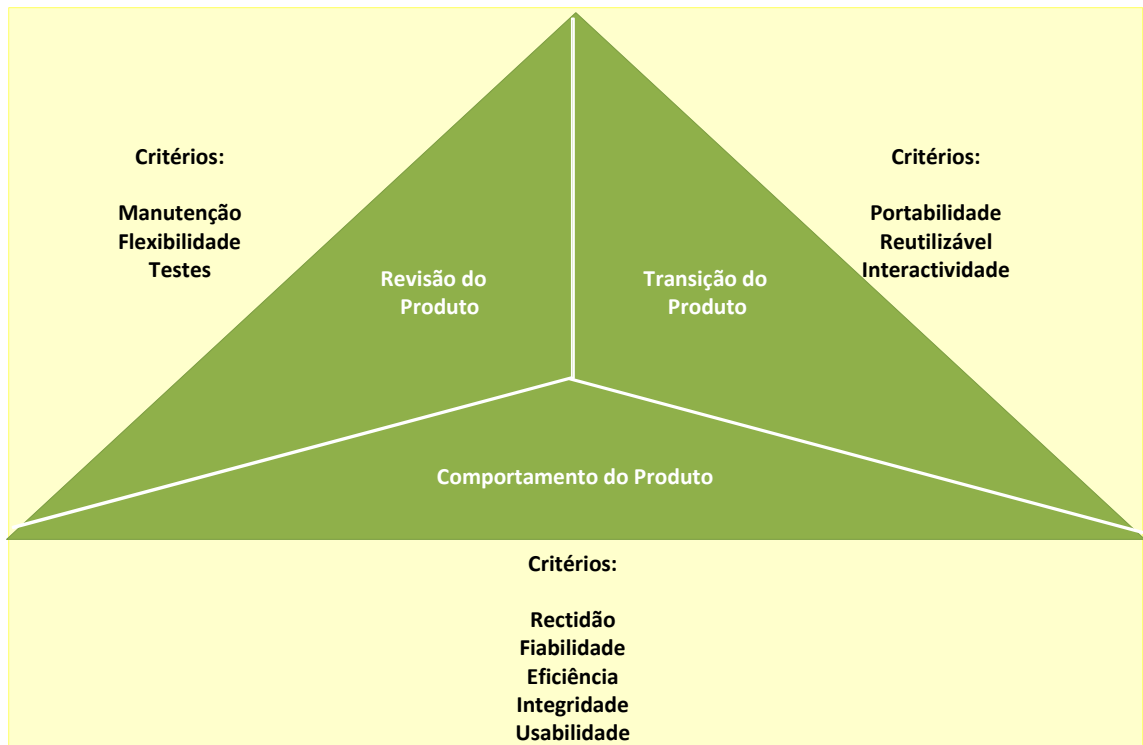


Figura 2.13 Modelo de Qualidade de Software de MacCall, adaptado (MacCall, Richards, & Walters, 1997)

Os critérios de qualidade definidos por MacCall, estão relacionados com a:

- Usabilidade - facilidade de utilização do *software*;
- Integridade - proteção do programa contra o acesso não autorizado;
- Eficiência - utilização de recursos, estando esta subdividida em:
 - Eficiência na execução;
 - Eficiência no armazenamento.
- Retidão - garantia que o programa atende a sua especificação;
- Fiabilidade - habilidade do programa para não ter falhas;
- Manutenção - esforço necessário para localizar e reparar uma falha do programa no seu ambiente operacional;
- Flexibilidade - facilidade de realizar modificações necessárias no ambiente operacional;
- Testes - facilidade de testar o programa para assegurar que ele não contenha erros e satisfaça a sua especificação;
- Portabilidade - esforço necessário para transferir um programa de um ambiente para o outro;

- Reutilizável - facilidade de reutilizar o *software* num contexto diferente;
- Interatividade - esforço necessário para juntar um sistema ao outro.

Neste modelo, a divisão dos vários critérios de qualidade de *software* em áreas tais como, Comportamento do Produto, Transição do Produto e a Revisão do Produto possibilita a concentração de esforços de forma direcionada e particularizada e enfatiza as características relacionadas com o produto final (Herbert & Price, 1995), referido por (Tait, 1998). Os critérios que se apresentam do lado esquerdo (da figura 2.13) não são mensuráveis, enquanto os critérios do lado direito são mensuráveis em relação à qualidade. MacCall propõe uma escala de avaliação entre zero (0) e dez (10).

Modelo de Boehm

O modelo de Boehm (Boehm, Brown, Kaspar, Lipow, Machleod, & Merritt, 1978) visa fornecer um conjunto de características de qualidade de *software* bem definidas e diferenciadas. Este modelo é hierárquico e os critérios de qualidade são subdivididos, com o objetivo de identificar as áreas a considerar, de forma semelhante ao modelo de MacCall.

O modelo de Boehm tem como objetivo agrupar características com pontos comuns, as quais não poderiam ser unidas em um único critério, devido à sua individualidade intrínseca. A partir do modelo, atribui-se pesos para cada uma das características, de acordo com os objetivos da medição de qualidade em determinada aplicação.

A unidade central identifica o produto final, pois ele é a chave para a avaliação, que, por sua vez, se subdivide em três características: Utilidade, Portabilidade, Manutenção. Estas agregam um conjunto de características, que consideramos as características de nível I. Por sua vez estas tornam a agregar outras características, a que designamos características de nível II.

Boehm avalia o *software* a partir deste “bolo” e atribui pesos às características. Estes são dados de acordo com a aplicação a ser avaliada, do objetivo desta e para quem é dirigida.

De referir que as características de nível II têm objetivos a alcançar. A tabela 2.8 apresenta as características, e os seus objetivos.

Características de nível II	Objetivos
Independência de dispositivos	Permite que o <i>software</i> possa ser executado em diferentes sistemas
Completo	Qualidade de estar “completo”. Os protótipos de baixa fidelidade não se conseguem avaliar
A precisão	As saídas são exatas. As indicações sobre o assunto são dúbias
A consistência	Qualidade de ser coerente
A eficiência de dispositivos	Capacidade de processamento pode apresentar desperdícios mínimos. Capacidade de processamento pode apresentar desperdícios máximos
A acessibilidade	O sistema é de fácil acesso.
A comunicabilidade	Facilidade de assimilação dos <i>inputs</i> e <i>outputs</i> e utilidade dos mesmos
A estrutura	Alude à organização da estrutura
A auto descrição	A nível dos protótipos, não se encontra desenvolvida
A concisão	Apointa para a quantidade de informação excessiva
A legibilidade	Facilidade de leitura do código.
Argumentativa	Facilidade de acomodar difusões nos dados e nas funções.

Tabela 2.8 Objetivos das características de nível II

A figura 2.14 apresenta o funcionamento do modelo de Boehm.

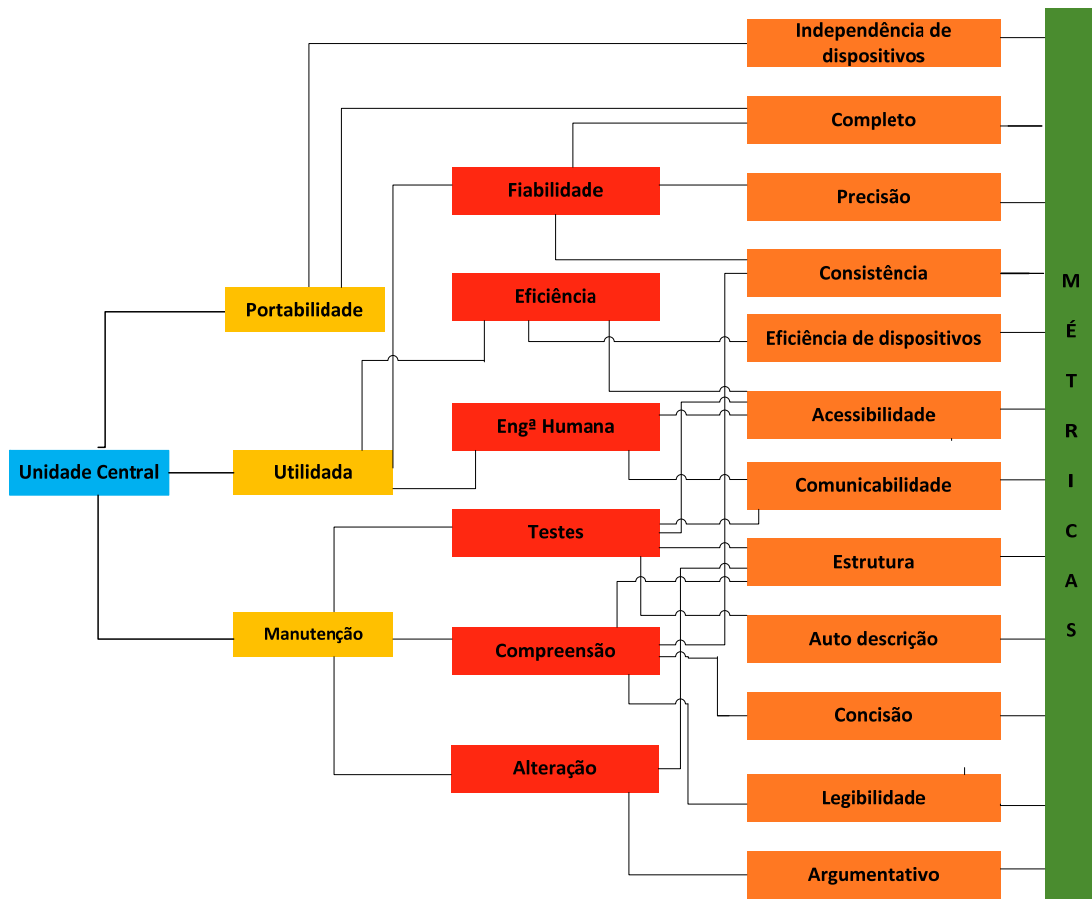


Figura 2.14 Modelo de Boehm (Boehm, Brown, Kaspar, Lipow, Machleod, & Merritt, 1978) de Qualidade de Software (Herbert & Price, 1995), adaptado

2.3.2 Modelos de Qualidade do Software Ergonómico, Princípios, Regras e Critérios

A origem da palavra Ergonomia, deriva do grego: ergon = Trabalho e nomos = Legislação, Normas.

Podemos questionar, “O que é o Trabalho?”, “O que são normas?”, “O que é Legislação”

Remontemos até à Bíblia Sagrada 3:4 Gn 3.17 “... da árvore de que te falei ordenei dizendo: Não comerás dela; maldita é a terra por tua causa; em fadiga comerás dela todos os dias da tua vida”. 3:4 Gn 3.19 “Do suor do teu rosto comerás o teu pão, até que tornes à terra, porque dela foste tomado; porquanto és pó, e ao pó tornarás” (Sagrada, 1963).

As ilações que retiramos de Gênesis(Gn) 3.17 e 3.19 estão associadas ao trabalho, e que este é algo penoso, doloroso.

A palavra "trabalho", deriva da palavra latina "*tripalium* " e o seu significado pode ser: instrumento de três pés, destinado a torturas, ou local onde se colocam os bois para serem ferrados, e ainda, instrumento constituído por três paus aguçados, por vezes munidos de pontas de ferro, (Santos J. B., 2000). No entanto, em qualquer um dos significados o sofrimento é notório.

Na Grécia Antiga a palavra "trabalho" apresentava dois significados: O primeiro significado decorre do grego "*ponos*", associa esforço e sofrimento; O segundo significado provém do grego "*ergon*", traduz beleza, obra de arte e criação, (Woleck, 2002) . Observamos que a associação do "trabalho" ao sofrimento fica para trás, e que este pode ser visto como algo belo, harmonioso e satisfatório.

Woleck refere que Karl Marx considerou o trabalho como a única fonte criadora da vida humana, " (...) *o trabalho revela o modo como o homem lida com a natureza, o processo de produção pelo qual ele sustenta a sua vida e, assim, põe a nu o modo de formação de suas relações sociais e das ideias que fluem destas*", (Woleck, 2002).

Iida indica que a Ergonomia é o "estudo da adaptação do trabalho ao homem"; o trabalho abrange não só as máquinas e equipamentos, mas todo o ambiente que envolve o homem e o seu trabalho (Iida, 2005).

Partindo deste princípio, a definição apresentada por AnaMaria Moraes (Moraes, s/data) "*Cumprer ressaltar que a singularidade da ergonomia está justamente na sua praxis, que integra o estudo das características físicas e psíquicas do homem, as avaliações tecnológicas do sistema produtivo, a análise da tarefa, com a apreciação, o diagnóstico, projectado, a avaliação e a implantação de sistemas homens-máquinas. O ergonomista, junto com engenheiros, arquitectos, desenhistas indústrias, analistas e programadores de sistema, organizadores do trabalho, propõe mudanças e inovações, sempre a partir de variáveis fisiológicas, psicológicas e cognitivas humanas e segundo critérios que privilegiam o ser humano*".

Para Alain Wisner (Wisner, 1994) e para Walter Cybis (Cybis, 1997) a Ergonomia é uma ciência que procura a adaptação do ambiente técnico e organizacional ao homem. Segundo eles *“a Ergonomia é o conjunto dos conhecimentos necessários para conceber ferramentas, máquinas, e dispositivos que possam ser utilizados com o máximo de conforto, segurança e eficiência”*.

No entanto, é sobejamente sabido que os aspetos laborais não são determinados unicamente por critérios ergonómicos. Estes podem ser alterados por influência de determinadas variáveis económicas, sociais e técnicas que determinam a organização do trabalho, a conceção de ferramentas, máquinas e a implantação de sistemas de produção, (Fialho & Santos, 1995), refere Luciano Gamez (Gamez, 1998).

Peixoto, ao analisar o conceito de ergonomia indicado por Itiro Iida (Iida, 2005), chama à atenção para o facto de a Ergonomia partir do conhecimento do homem para elaborar o projeto, ajustando-o à medida humana, Tei Hiratsuka (Hiratsuka, 1996).

Contudo, a Ergonomia só passou a ser verdadeiramente conhecida aquando da criação da *“Ergonomics Research Society”* composta por psicólogos, fisiologistas e engenheiros, interessados nos problemas da adaptação do trabalho ao homem Antoine Laville (Laville, 1997).

A *“Ergonomics Research Society”* divide a ergonomia em 3 domínios, sendo eles:

- Ergonomia Física - relaciona a carga física e psicológica com a reação do corpo humano;
- Ergonomia Cognitiva – alude os processos mentais, como atenção, percepção, cognição, controlo motor e recuperação de memória, e como estes interagem entre os seres humanos e outros sistemas. Neste sentido, incluímos a carga mental de trabalho, vigilância, tomada de decisão, desempenho de habilidades, interação homem computador e o treino;
- Ergonomia Organizacional - relacionada com a otimização dos sistemas sociotécnicos e que inclui a satisfação no trabalho, a teoria motivacional, o trabalho em equipa, trabalho à distância e ética.

Etienne Grandjean (Grandjean E. , 1998) identifica quatro estágios do desenvolvimento da ergonomia:

- Ergonomia de interface homem-máquina - diz respeito à ciência e tecnologia e na prática das questões físicas e de percepção aplicáveis a produtos, equipamentos e postos de trabalho;
- Ergonomia da interface homem-ambiente, ou ergonomia ambiental - trata de temas relacionados como o ser humano em seu ambiente;
- Ergonomia da interface organização-máquina - tenta perceber os aspetos relativos ao sistema produtivo como um todo;
- Ergonomia da interface homem-computador - foca essencialmente aspetos cognitivos.

Para Etienne Grandjean (Grandjean E. , 1998) “ *Ergonomia pode ser definida como a ciência da configuração de trabalho adaptado ao homem, sendo o seu objectivo, o desenvolvimento de bases científicas para a adequação das condições de trabalho às capacidades e realidades da pessoa que trabalha*”.

Segundo David Whitfield e Joe Langford (Whitfield & Langford, 2001) “*Ergonomics is the application of scientific information concerning humans to the design of objects, systems and environment for human use. Ergonomics comes into everything which involves people. Work systems, sports and leisure, health and safety should all embody ergonomics principles if well designed*”.

Como tal, a Ergonomia tem um papel crucial no desenvolvimento e avaliação de sistemas de informação e comunicação. Através da Ergonomia é possível compreender as interações entre o Homem e os componentes de um sistema de informação ou comunicação, tendo em vista a sua otimização, proporcionado conforto e segurança aos utilizadores e eficácia dos sistemas.

Para que os utilizadores se sintam confortáveis ao utilizarem um determinado sistema de informação, deve-se analisar a necessidade dos mesmos e entender como pode a tecnologia contribuir para alcançar as necessidades dos utilizadores, Heinrich Bauersfeld (Bauersfeld, 1998).

Existem um conjunto de técnicas para ajudar a compreender essas necessidades, de forma a ajudar os analistas e programadores de sistemas, a identificar, desenhar e desenvolver processos para o sucesso no *design* de interface adaptada ao utilizador. Estas técnicas fazem parte da Ergonomia de *Human Computer Interface (HCI)*, pois esta preocupa-se com a relação existente entre o utilizador face a um dispositivo interativo, procurando alcançar o equilíbrio ótimo entre o conforto, segurança e eficiência do utilizador de *software*.

Álvaro Gálvis (Gálvis, 1997) define que o interface é a zona de comunicação em que se realiza a interação entre o utilizador e o programa. Nele estão contidos todo o tipo de mensagens compreensíveis para os utilizadores, tais como, mensagens verbais, simples, sonoras e imagens. Por outro lado, existem as mensagens compreensíveis pelo próprio *software* sendo elas, mensagens, sinais elétricos, e outros. Para haver a troca de mensagens devem existir os dispositivos de entrada e saída e ainda as zonas de comunicação em cada dispositivo.

Para este autor, o *design* de interface depende em grande medida das características socioculturais e genéricas dos utilizadores. Como tal, na definição de uma interface, o *designer* deve analisar a terminologia, simbologia e particularidades do assunto de abrangência do sistema em referência, em função das características socioculturais dos utilizadores.

Human Computer Interface (HCI)

Os estudos da Ergonomia HCI têm estimulado o desenvolvimento de modelos ergonómicos de interação. Podemos mesmo dizer, que a ligação comportamental homem-computador envolve o estudo das tarefas dos utilizadores e dos fatores envolvidos no meio em que a interação ocorre (Eason, 1991) referido por Ping Zhang (Zhang, 2004) e Andrew Dillon (Dillon, 1993).

Segundo o autor Eason (Eason, 1991), tratar a interação homem-máquina como uma forma de conversação entre diferentes tipos de participantes, cada qual com as suas capacidades para armazenar, processar e transmitir, ajuda a compreender a formação dos processos cognitivos humanos. No entanto, o principal objetivo do homem na interação com os computadores, é de este poder executar tarefas em situações reais, que envolvam o seu meio ambiente, de forma a facilitar o seu dia a dia. O modelo do desempenho de

determinadas tarefas, baseado numa abordagem ergonómica clássica, homem-máquina-tarefa e meio ambiente, pode ser representado na figura 2.15. Podemos constatar que o núcleo central é o nível 1 onde a ligação para o nível 2 é efetuada pela tarefa. Por outro lado, não existe uma ligação direta entre o nível 2 e o nível 3, como se entre eles houvesse uma sobreposição.

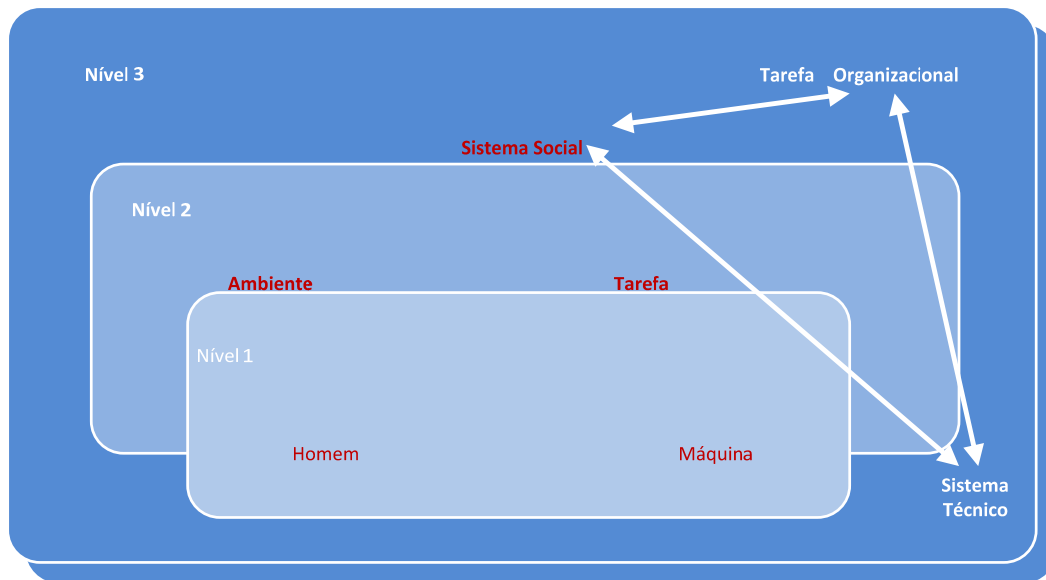


Figura 2.15 Abordagem Ergonómica das Tarefas, adaptado (Eason K. , 1991)

Devemos ter em conta que, independentemente do fim para que se destina todo e qualquer tipo de *software* desenvolvido, este é uma ferramenta usada pelo homem. Como tal, deve apresentar todas as características ergonómicas exigidas. Estas poderão ser, funcionalidade, representações mentais envolvidas, diálogos fáceis, apresentação clara, lógica de utilização coerente com a lógica do utilizador, linguagem natural da situação de trabalho, linguagem natural da situação educacional, entre outras (Jarufe, 2004).

O modelo *Human Computer Interface* (HCI) indica precisamente a situação por nós referida no parágrafo anterior. A figura 2.16 representa o relacionamento entre as características humanas (processos cognitivos, processos metacognitivos) e o computador.

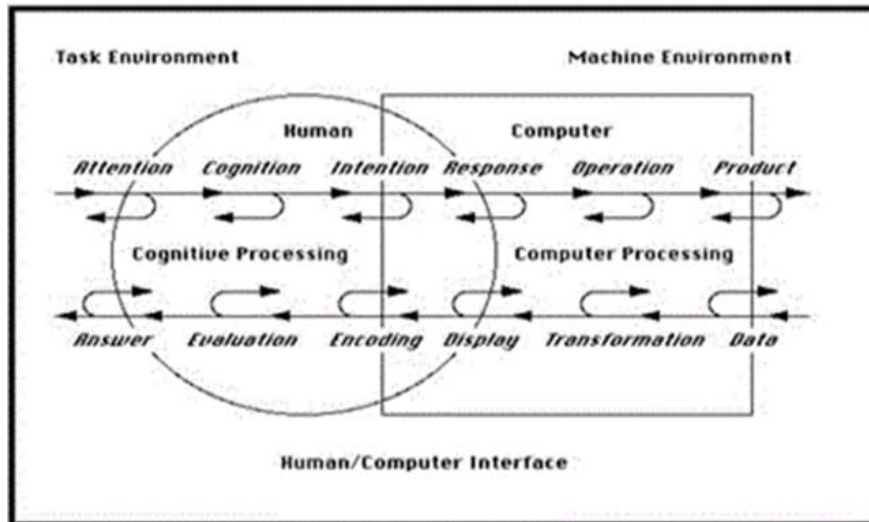


Figura 2.16 Modelo Human Computer Interface (HCI)

Este modelo deve relacionar-se com o modelo do processo *User Interface Design (UID)* (figura 2.26) e a ponte entre eles deve ser feita pelo especialista ergonómico e pela equipa de especialista em fatores humanos. O seu papel é fundamental no desenho do interface, que deve ser o mais acessível possível, para ir de encontro às necessidades humanas, e assim atingir a qualidade.

Maíria de Paula, Simone Barbosa e Carlos Lucena (Paula, Barbosa, & Lucena, 2005) acreditam que a comunicação entre a engenharia de *software* e modelo HCI é vital para o desenvolvimento de software ergonómico e interativo, transcrevem a sua visão através da figura 2.17. Para estas autoras, o papel dos “*HCI designers*” é de uma grandeza unívoca, pois, são eles que estabelecem a ligação entre os utilizadores da aplicação e os engenheiros informáticos.

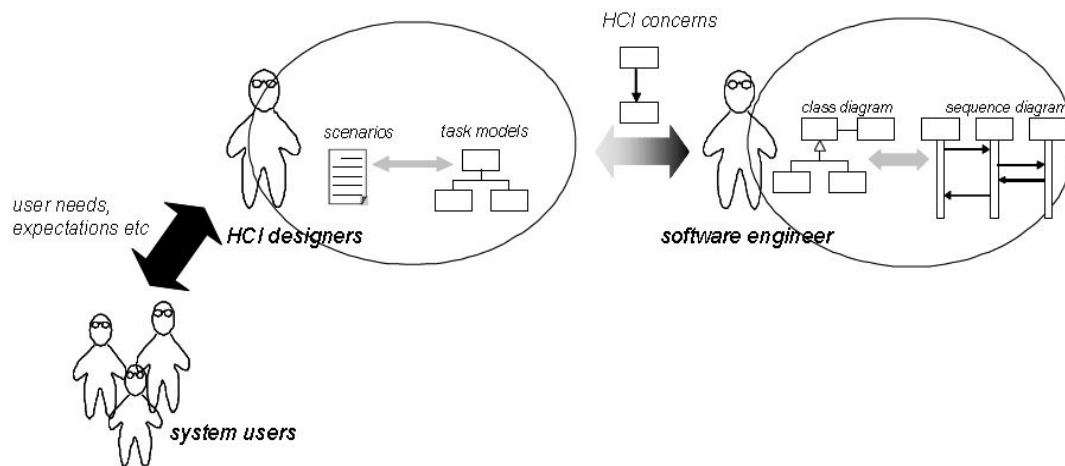


Figura 2.17 Envisage communication between HCI designers and software engineers, (Paula, Barbosa, & Lucena, 2005)

O trabalho destas autoras reflete a importância do modelo *A Modeling Language for Interaction as Conversation (MoLic)* com o modelo *Human Computer Interaction*.

O modelo MoLic é um modelo de interação criado na engenharia semiótica. “Ele representa todas as possíveis conversas interactivas que os utilizadores podem ter com o sistema, ou seja, todos os possíveis caminhos de interacção, incluindo caminhos alternativos para atingir o mesmo objectivo, e caminhos para a recuperação de erros ou falhas interactivas”, (Paula, Barbosa, & Lucena, *Conveying Human-Computer Interaction Concerns to Software Engineers through an Interaction Model*, 2005).

Todos os modelos apresentados, têm como objetivo final a qualidade. Esta qualidade reflete-se no *design* de interfaces interativos que está diretamente relacionada com a usabilidade e facilidade de uso dos sistemas.

Como tal, torna-se importante avaliar a qualidade em relação à usabilidade e facilidade de utilização dos sistemas, os quais estão diretamente ligados com a satisfação do utilizador. A satisfação do utilizador está interligada com a ergonomia, visto que, usabilidade e facilidade dos sistemas são características da ergonomia.

Portanto, não devemos esquecer que a Ergonomia se aplica em diversas vertentes. Independentemente da vertente o termo “usabilidade” está sempre presente, pois quanto

maior interatividade existir num sistema, maior são os níveis de eficácia, eficiência e satisfação do utilizador final, que se realiza através do *design* da interface.

As Oito Regras de Ouro de Shneiderman

Ben Shneiderman contribui com as oito regras de “ouro” de *design* de interfaces, (Shneiderman B. , 1998), as quais passamos a descrever:

- 1ª Manter a consistência ou seja a terminologia usada deve ser idêntica nos menus: nos ecrãs de ajuda, os *layouts*, as cores e o tipo de letra também devem ser consistentes;
- 2ª Reduzir o número de interoperabilidade e de aumentar o ritmo de interação dos utilizadores. Para tal, são necessárias abreviações, teclas de funções e instalações de macros. O tempo de resposta deve ser rápido e curto;
- 3ª Oferecer feedback informativo. Para determinadas ações o sistema deve dar uma resposta. Esta pode ser simples ou substancial, depende do tipo da ação envolvida;
- 4ª Desenhar diálogos com princípio, meio e fim, nos quais deve haver sempre um *feedback*;
- 5ª Oferecer prevenção e recuperação de erros, ou seja caso o utilizador comente algum erro o sistema deve ser capaz de o detetar e recuperar a informação;
- 6ª Permitir desfazer facilmente as operações. As ações cometidas pelos utilizadores devem ser reversíveis;
- 7ª Favorecer a sensação de controlo. Deve permitir a sensação de controlo do sistema, e que este responda às suas ações;
- 8ª Reduzir a carga na memória de curta-duração, através da conceção de ecrãs onde as ações são bem visíveis ou usando menus suspensos e ícones.

A oitava regra tem como base o estudo efetuado por George Miller, “*The Magical Number Seven, Plus or Minus Two: Some Limits on our Capacity for Processing Information*”, (Miller, 1956). O seu estudo prova que o sistema cognitivo do homem só consegue processar um número limitado de informações que variam entre cinco (5) a nove (9) elementos por vez, ou seja, consegue apreender, de maneira natural e satisfatória, de cinco a nove elementos de

informação por vez. Uma vez excedidos esses limites, o raciocínio e a aprendizagem ficam abaixo do desempenho esperado, sobrecarregando a estrutura cognitiva.

Segundo Miller a capacidade de memória em pessoas jovens é por volta de sete (7) elementos, os quais designou de “*Chunks*”, indiferentemente das características dos elementos, quer eles fossem dígitos, letras ou números.

Um estudo posterior realizado por Hulme, Roodenrys, Brown e Mercer revelou que a capacidade de memória depende da categoria dos *chunks*, isto é, a capacidade de memória é de sete elementos para dígitos, cerca de seis elementos para letras e à volta de cinco elementos para palavras. No entanto, na mesma categoria de *chunks* podem existir capacidades de memória diferentes, caso seja uma “palavra longa” a capacidade é menor do que para “palavras menores”, (Hulme, Roodenrys, Brown, & Mercer, 1995) .

Nelson Cowan releva que existem diversos fatores que podem influenciar a capacidade de memória de um indivíduo e como tal é difícil determinar a capacidade de memória de curto prazo pela quantidade de *chunks*, (Cowan, 2001) . Contudo, Cowan propõe que a capacidade de memória em adultos jovens é superior à das crianças e à dos adultos mais velhos.

Os 7 Princípios de Norman

Norman também contribui para o desenvolvimento da ergonomia, através dos aspetos cognitivos do desenho. Para tal, descreve, sete princípios que devemos ter em consideração, aquando da criação de *designer* de interface, os quais são conhecidos como “Os 7 princípios de Norman”, (Norman D. A., 1990) e que são:

- 1º Utilizar simultaneamente o conhecimento do mundo (e.g. balde do lixo para objetos desnecessários) e interno ao utilizador (e.g. teclas de atalho de outros programas);
- 2º Simplificar a estrutura das tarefas (e.g. assistente de instalação);
- 3º Tornar as coisas visíveis: estreitar os desníveis de Execução e Avaliação (e.g. agarrar num ficheiro e movê-lo para outra diretoria);
- 4º Efetuar as associações corretamente (e.g. nome das opções num menu);

- 5º Explorar o poder das limitações, tanto naturais como artificiais (e.g. sistema que oferece diferentes opções consoante o tipo de utilizador);
- 6º Desenhar para o erro (e.g. operações de desfazer);
- 7º Quando tudo falha, seguir o *standard*.

Os Oito Critérios de Scapin & Bastien

Entretanto as pesquisas efetuadas pelos ergonomistas franceses Dominique Scapin e Christian Bastien (Scapin & Bastien, 1993), foram desenvolvidas com o objetivo de auxiliar nos processos de avaliação de *software*, tendo como bases teóricas, a ergonomia aplicada à informática.

Estes ergonomistas, definem quais são as qualidades que devem ser atribuídas ao *software* durante o projeto, para que, durante a interação, satisfaçam o utilizador.

Assim, (Scapin & Bastien, 1993) definiram um conjunto de oito critérios, sendo eles: condução, carga de trabalho, controle explícito, adaptabilidade, gestão de erros, consistência, significado dos códigos e compatibilidade. Esta avaliação também é conhecida como avaliação heurística. Passamos a descrever os objetivos de cada um dos critérios.

- A *condução* define-se como a rapidez do sistema, na legibilidade das informações, no *feedback* imediato das ações do utilizador e no agrupamento e distinção entre itens no ecrã. Esse último subcritério refere-se tanto aos formatos - agrupamento e distinção por formato, como à localização - agrupamento e distinção por localização dos itens;
- A *carga de trabalho* define-se na brevidade das apresentações – concisão nas entradas – ações mínimas e na densidade informacional dos ecrãs como um todo;
- O *controlo explícito* define-se no carácter explícito das ações do utilizador – ações explícitas e no controle que ele tem sobre os processamentos – controle do utilizador;
- A *adaptabilidade* refere-se tanto às possibilidades de personalização do sistema que são oferecidas ao utilizador – flexibilidade como ao facto da estrutura do sistema estar adaptada a utilizadores de diferentes níveis de experiências – consideração da experiência do utilizador;
- A *gestão de erros* refere-se tanto aos dispositivos de prevenção que possam ser definidos nas interfaces – proteção de erros, como à qualidade das mensagens de erro

fornecidas e as condições oferecidas para que o utilizador recupere a normalidade do sistema ou da tarefa – correção dos erros;

- *A consistência* refere-se à homogeneidade e coerência das decisões de projeto quanto às apresentações e diálogos.

2.3.3 Normas Ergonómicas

Passemos a analisar o significado da palavra “Norma,” segundo o Instituto Português da Qualidade (IPQ), *“uma norma é um documento estabelecido por consenso e aprovado por um organismo reconhecido, que fornece regras, linhas directrizes ou características, para actividades ou seus resultados, garantindo um nível de ordem óptimo num dado contexto”,* (Desconhecido, Universidade de Aveiro, sem data).

“Qualquer norma é considerada uma referência idónea do mercado a que se destina, sendo por isso usada em processos: de legislação, de acreditação, de certificação, de metrologia, de informação técnica, e até por vezes nas relações comerciais Cliente – Fornecedor” (Desconhecido, Instituto Português da Qualidade, S/data).

No dicionário de língua Portuguesa Infopédia, norma é uma regra de procedimento, um princípio, um preceito ou um modelo, (Editora, 2003).

As normas estão associadas à legislação e esta por sua vez à lei. Assim, a legislação é conhecida como: coleção de leis de um país; direito de legislar ou um conjunto de preceitos legais que regulam certa matéria. A lei pode ser entendida como: - prescrição do poder legislativo cujo cumprimento visa a organização da sociedade; - qualquer norma de conduta, geralmente jurídica; - preceito emanado de autoridade soberana; - obrigação; - regra; -norma, (Editora, 2003).

As normas para a avaliação de qualidade de *software* são criadas na Organização Internacional de Normalização (ISO) e na *International Electrotechnical Commission* (IEC). O título de uma norma ISO e/ou IEC indica a sua origem (ver anexo I). No entanto, as normas ISO e/ou IEC têm significados, princípios e recomendações, para que possamos seleccionar determinada norma para um determinado objetivo. As normas são agrupadas de acordo com as suas características e abrangências, o que está de acordo com o anexo J.

Existem normas internacionais que relacionam o *software* com o modelo HCI e a usabilidade, visto que nestas relações está inerente a qualidade do *software*. Há inúmeras expectativas em relação à qualidade do *software* e mais ainda em relação às normas de desenvolvimento, pois estas normas contribuem para uma melhor facilidade de utilização do *software*.

Para que se possa gerir a qualidade de *software*, é importante avaliar tanto a qualidade do produto como a qualidade do processo de desenvolvimento. Assim sendo, “é necessário um modelo que permita a avaliação da qualidade de *software*”, (ISO/IEC 9126/1991), (International standards for HCI and usability, 2003) .

Esta norma foi alterada, originando um novo padrão, designado por “Parte 4”, (Usabilidade, 2001). Factualmente esta norma assegura que a usabilidade desempenha duas funções. A primeira função deve apresentar em detalhe toda a atividade do desenho de *software*. A segunda função deve atender todas as necessidades do utilizador.

A Norma ISO/IEC 9126-1 relaciona a fase de projeto e a fase de implementação, de modo a identificar as possíveis alterações a serem efetuadas, para assim se obter um *software* de qualidade (International standards for HCI and usability, 2003).

A qualidade como objetivo a alcançar, apresentava um conjunto de características que se tem de verificar num *software*. A tabela 2.9 descreve essas características.

A parte 4 da norma ISO/IEC D TR 9126-4 (International standards for HCI and usability, 2003) está relacionada com a qualidade do produto e as suas métricas. Esta norma propõe mensurar a eficácia, produtividade, satisfação e a segurança. Esse resultado pode ser apresentado através de um relatório específico de usabilidade, de acordo o *National Institute of Standards and Technology* (NIST).

Características	Itens da característica	Pergunta chave para a característica
Funcionalidade (Satisfaz as necessidades?)	Adequação, Precisão, Interoperabilidade, Conformidade, Segurança de Acesso	Propõe-se a fazer o que é apropriado? Faz o que foi proposto de forma correta? Interage com os sistemas específicos? Está de acordo com as Normas, leis, etc.? Evita acesso não autorizado aos dados?
Fiabilidade (É imune a falhas?)	Maturidade, Tolerância a erros Recuperabilidade	Com que frequência apresenta erros? Ocorrendo erros, como reage? É capaz de recuperar dados em caso de erro?
Usabilidade (É fácil de usar?)	Inteligível Apreensível Operacional	É fácil entender o conceito e a aplicação? É fácil de aprender a usar? É fácil de operar e controlar?
Eficiência (É rápido e conciso?)	Tempo Recursos	O tempo de resposta e execução é adequado? Quantos recursos usam? Durante quanto tempo?
Manutenção (É fácil de modificar?)	Analisável Modificável Estável Testável	É fácil de encontrar uma falha, quando ocorre? É fácil de modificar e adaptar? Há riscos quando se fazem alterações? É fácil testar quando se fazem alterações?
Portabilidade (É fácil usar em outro ambiente?)	Adaptável, Facilidade de Instalação, Conformidade, Facilidade de Substituição	É fácil de adaptar a outros ambientes? É fácil instalar em outros ambientes? Está conforme os padrões de portabilidade? Fácil de substituir por outro ambiente?

Tabela 2.9 Descrição da norma ISO 9126, adaptado (International standards for HCI and usability, 2003)

A norma ISO/IEC D TR 9126-4 passou a apresentar três abordagens diferentes para a qualidade de um produto (Bevan N. , Quality in Use: Meeting User Needs for Quality, 1999).

No entanto todas as abordagens se inter-relacionam, sendo elas:

- A qualidade interna está relacionada com as propriedades estáticas do código;
- A qualidade externa é analisada pelas propriedades dinâmicas do código quando executado, ou seja, pelo tempo de resposta;
- A qualidade no uso é vista sob a perspectiva do utilizador, ou seja, se o *software* atende às necessidades do utilizador.

A figura 2.18 representa a situação descrita por Nigel Bevan (Bevan N. , Quality in Use: Meeting User Needs for Quality, 1999)

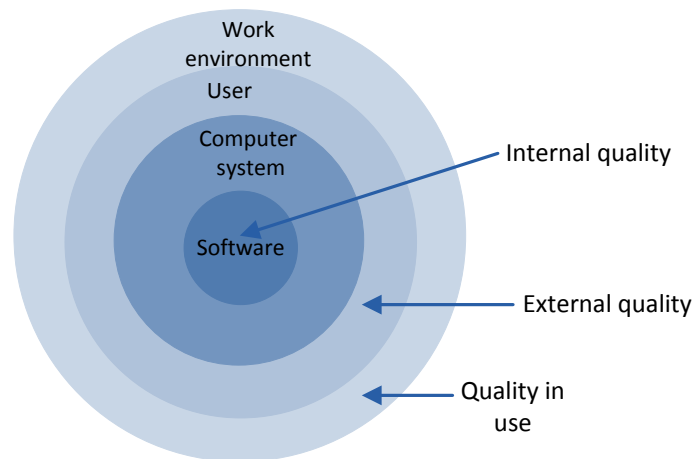


Figura 2.18 Relação entre as três abordagens de *software*, fonte (Bevan N. , *Quality in Use: Meeting User Needs for Quality*, 1999)

Segundo o mesmo autor, para que o *software* atenda às necessidades específicas dos utilizadores, os atributos da qualidade interna são um pré-requisito, para alcançar o comportamento externo exigido. Este por sua vez é também um pré-requisito para se alcançar a qualidade em uso, de acordo com a figura 2.19.

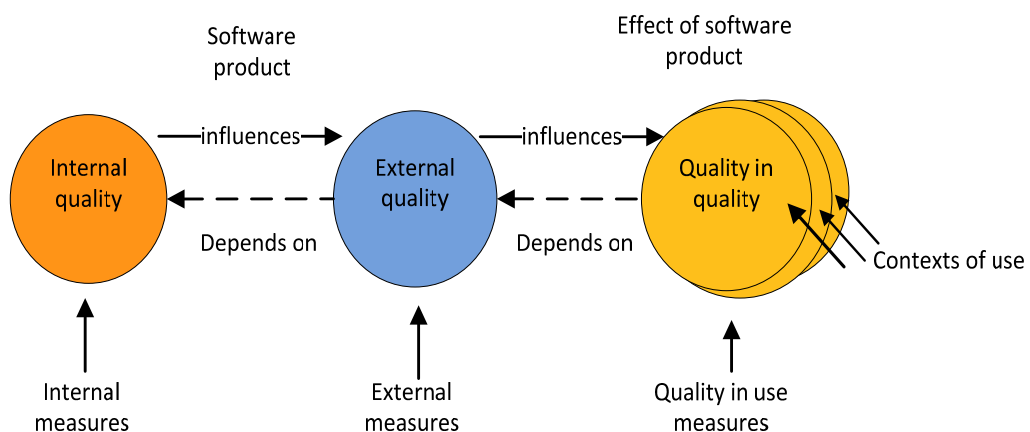


Figura 2.19 Abordagens para a qualidade de produtos de *software*, fonte (Bevan N. , *Quality in Use: Meeting User Needs for Quality*, 1999)

A norma ISO 9241- *Ergonomics requirement for office work with visual display terminals*, trata das recomendações ergonômicas para o trabalho de escritório informatizado. Relaciona-se com a norma anterior, na medida que trata com maior profundidade os aspetos relacionados à usabilidade. Apresenta um conjunto de recomendações ergonômicas no *design* de sistemas, destinadas a orientar a concepção de produtos e promover a usabilidade dos sistemas, (Bevan, Claridge, M.Maguire, & M.Athousaki, 2002).

Para esta norma, “*eficácia, é a capacidade que os sistemas concedem a diferentes utilizadores, para que estes atinjam os seus objectivos em número e qualidade*”, a “*eficiência, é a quantidade de recursos que os sistemas solicitam aos utilizadores para a obtenção dos seus objectivos*”, a “*satisfação, é o prazer que os sistemas proporcionam quando os utilizadores atingem os seus objectivos*”. No nosso estudo esta norma teve toda a nossa atenção, pois um dos nossos objetivos principais vai de encontro a estas variáveis.

De referir que existem outras normas que nos podem apoiar no desenvolvimento de interação de interface com o utilizador, tais como: A Norma ISO 14915 e IEC 61997, a Norma ISO/IEC 11581, a norma ISO/IEC 10741.

De acordo a organização *Interface Systems International Design* (I.S.I.D., 1999) as normas de usabilidade podem-se agrupar da seguinte forma:

- Qualidade de uso (eficácia, eficiência e a satisfação do utilizador);
- Qualidade do produto que está relacionado com o interface e a interação com o utilizador;
- Qualidade do processo que relaciona o processo utilizado para desenvolver o produto;
- A capacidade organizacional que reflete a capacidade de uma organização para aplicar o *design* centrado no utilizador.

A figura 2.20 apresenta as fases correspondentes ao processo qualidade de desenvolvimento de *software*.

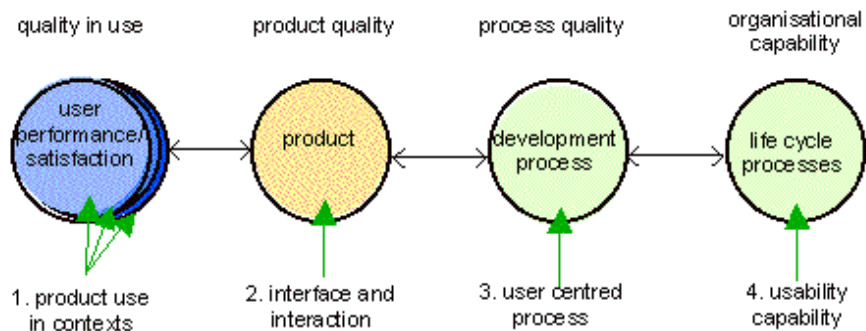


Figura 2.20 Fases do processo de qualidade de desenvolvimento de *software*, fonte (ISID, 1999)

2.4 Metodologias de Desenvolvimento de Software

Os sucessos e os insucessos do desenvolvimento de *software*, devem-se a um conjunto de ocorrências, em cuja gênese raramente se encontram somente problemas técnicos.

Devemos ter em conta que, independentemente para que fim se destina todo e qualquer tipo de *software* desenvolvido, este é uma ferramenta usada pelo homem. Como tal, deve apresentar todas as características ergonómicas exigidas. Estas poderão ser: funcionalidade, representações mentais envolvidas, diálogos fáceis, apresentação clara, lógica de utilização coerente com a lógica do utilizador, linguagem natural da situação de trabalho, linguagem natural da situação educacional, entre outras (Jarufe, 2004).

No entanto, nós iremos abordar somente as questões técnicas, pois são aquelas que estão direcionadas para os nossos objetivos, anteriormente descritos. As questões técnicas são tratadas na engenharia de *software*, que engloba a análise de requisitos, tais como desenhar, codificar, testar, etc.

A engenharia de *software* engloba três fases, estas por sua vez englobam várias atividades, independentemente do domínio aplicacional, da dimensão e da complexidade do projeto, (Miguel, 2006) . Assim, podemos concluir que o desenvolvimento de *software* tem um ciclo de vida, denominado “Ciclo de Vida de Software”. De acordo com esse ciclo, apresentamos a descrição das três fases e das suas atividades.

A fase de definição é direcionada a identificar a informação a ser processada, as funções, os desempenhos pretendidos, o comportamento do sistema, definição de interfaces, restrições e critérios de validações. Como tal, Fernanda Pedro e Elsa Rodrigues (Pedro & Rodrigues, 2009) propõem que esta fase englobe as seguintes atividades:

- Atividade 1 – Levantamento dos Requisitos Funcionais e Não Funcionais
 - Identificar utilizadores chave;
 - Identificar problemas atuais;
 - Estabelecer objetivos para o novo sistema;
 - Determinar possíveis automatizações do sistema;
 - Detalhar o resto do ciclo de vida do projeto.

- Atividade 2 – Análise de Sistemas

- Desenvolver a análise recorrendo às ferramentas de modelação.

➤ Atividade 3 – Projeto

- Especificação de interfaces;
- Apresentação dos protótipos de baixa fidelidade.

A fase de desenvolvimento, traduz o modo como os dados estão estruturados, a implementação das funções dos interfaces, ou seja o desenho é traduzido numa linguagem de programação. As atividades que compõem esta fase de acordo com Fernanda Pedro e Elsa Rodrigues (Pedro & Rodrigues, 2009) são:

➤ Atividade 4 – Implementação

- Codificação;
- Protótipos de alta-fidelidade;
- Integração de módulos (programação e implementação.)

➤ Atividade 5 – Geração de testes

- Criação de casos de testes

A fase de manutenção, deve corrigir os erros e efetuar as adaptações necessárias ao *software*. Segundo Roger Pressman (Pressman R. , 2000) durante esta fase é possível encontrar quatro tipos de alterações, são elas:

- Correções – pedido de alterações ao software da parte do utilizador ou do cliente;
- Adaptações – adaptações do *software* a alterações provenientes de variáveis externas, exemplo, regras do negócio alteradas;
- Melhorias – melhorias pretendidas pelo cliente/utilizador, ou seja, aquisição de funções adicionais, que não estavam contempladas nos requisitos funcionais iniciais;
- Prevenção – está associada à reengenharia de *software*, que permite efetuar alterações aos programas para que eles possam ser adaptados e melhorados.

Esta fase contempla as seguintes atividades, de acordo com Fernanda Pedro e Elsa Rodrigues (Pedro & Rodrigues, 2009) :

➤ Atividade 6 – Controlo de Qualidade

- Teste final à qualidade do sistema.

- Atividade 7 – Procedimentos
 - Descrição dos procedimentos associados ao sistema
- Atividade 8 – Conversão de dados
 - Extração;
 - Limpeza e conversão (*cleansing*) de dados de sistemas anteriores para os atuais.
- Atividade 9 – Arranque (*go live*)
 - Instalação e entrada em produtivo do novo sistema.

O desenvolvimento de *software* implica a existência de processos de desenvolvimento. Um processo é determinado por um conjunto de atividades sequenciais de forma a executar uma tarefa.

Segundo Francisco Duarte (Duarte, 2002) “(...) *não existem processos de desenvolvimento de software mais ou menos correctos, mas sim processos de desenvolvimento mais ou menos adequados à complexidade do projecto pedido, ao tipo de equipa que o pratica, e ao tipo de utilização pretendido para o produto de software resultante (...)*”.

Assim, no momento de escolha do modelo de processo a usar, também se deve ter em conta as seguintes abordagens genéricas citadas por Francisco Duarte (Duarte, 2002):

- “*O desenvolvimento incremental: acarreta uma série de incrementos de funcionalidades em situações onde se verifique restrições de pessoas e orçamentos, sistemas de alto risco, requisitos funcionais mal especificados*”;
- “*O desenvolvimento orientado ao custo ou ao plano de prazos: as capacidades requeridas para um sistema devem ter prioridades e o desenvolvimento das capacidades devem ser apresentados por ordem decrescente de importância*”.

Nos subpontos seguintes descrever-se-ão resumidamente os principais modelos do processo de desenvolvimento.

2.4.1 Modelo em Cascata/Queda de Água/WaterFall

Este modelo de desenvolvimento de *software* é constituído por um conjunto de fases executadas sequencial e sistematicamente, de acordo com a figura 2.21. Uma fase só começa quando a anterior estiver concluída. Não permite modificações nos requisitos, o que dificulta o papel do analista, visto que é muito difícil capturar todos os requisitos de uma só vez. É excessivamente sincronizado, ou seja o cliente/utilizador recebe a versão final do *software* muito tarde. Esta situação origina que mais de metade dos custos sejam imputados na manutenção.

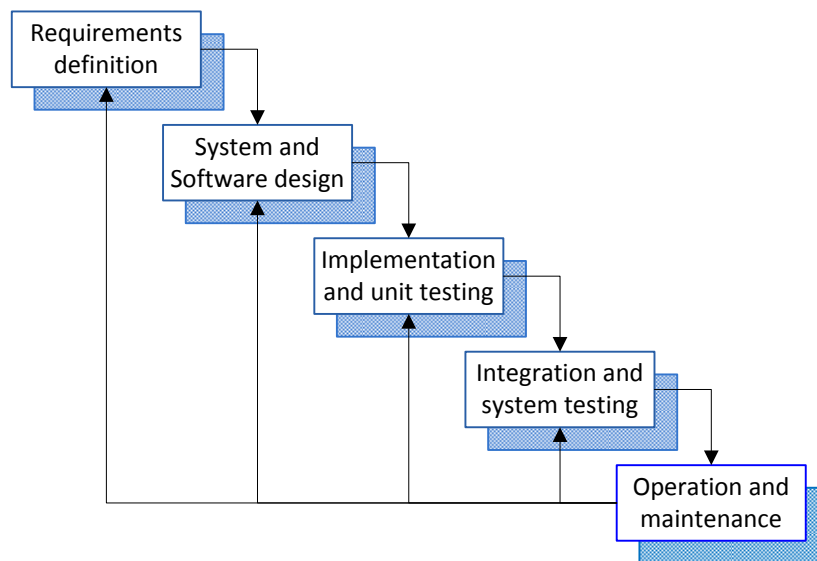


Figura 2.21 Modelo em Cascata/Queda de Água/WaterFall, fonte (Sommerville I. , 2001)

Entretanto obriga a reconfiguração do projeto dinamicamente. Como tal, é um risco muito grande, porque não se pode garantir que todas as fases vão correr bem, (e.g. pode-se ter dificuldades na deteção de erros a nível do código).

2.4.2 Modelo de Prototipagem

Este modelo surge, quando a incerteza se instala num projeto, pelo facto do cliente/utilizador não ter identificado todos os requisitos ou por ventura a equipa de desenvolvimento não está segura sob a aplicabilidade de um dado algoritmo, Roger Pressman (R.S. Pressman, 1997).

Assim, cria-se um protótipo que se apresenta ao cliente e, posteriormente, são inseridas as suas suscitações que estarão no próximo protótipo. Este ciclo repete-se até se obter o protótipo que satisfaça os requisitos do cliente. Logo, o projeto é apresentado ao cliente/utilizador ao longo do processo. Esta situação permite que os requisitos essenciais sejam identificados em conjunto, de acordo com a figura 2.22.

Gerir as reuniões, entre a equipa que desenvolve o *software* e o cliente/utilizador, torna-se uma tarefa árdua para que não haja mal-entendidos. O cliente/utilizador deve ser informado de que, neste modelo, não são considerados aspetos como a manutenção ou a qualidade do *software*. Esta situação deve ser devidamente acutelada, para não originar conflitos e dissabores entre todos os elementos.

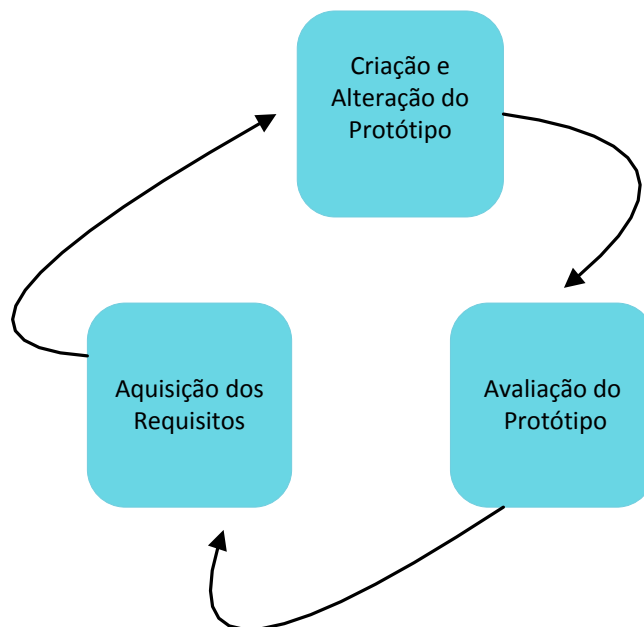


Figura 2.22 Modelo de Prototipagem

2.4.3 Modelo em V (Quality Assurance e Quality Control)

O modelo em V preconiza que os testes se devem iniciar na especificação dos requisitos, sendo estes paralelos e integrados ao processo de desenvolvimento de *software*. A figura 2.23 seguinte representa o Modelo V.

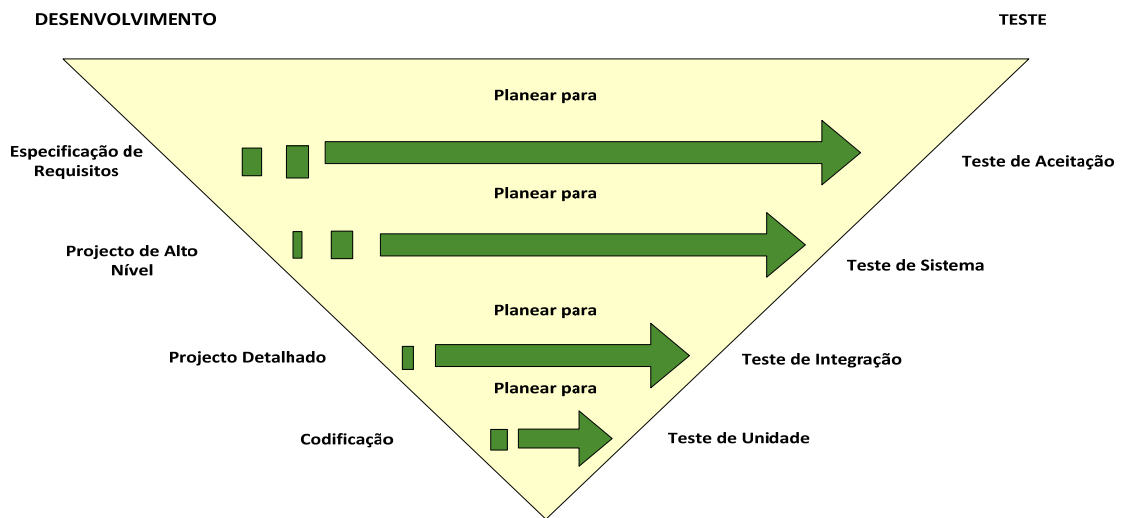


Figura 2.23 Modelo em V descreve o paralelismo entre as atividades de desenvolvimento e testes ao software, fonte (Neto A. C., s/data), cita (Craig & Jaskiel, 2002)

Para (Tamashiro, 2010) “(...) este modelo minimiza os custos da não qualidade do software, segundo a regra 10 de Myers quanto mais cedo um defeito for encontrado, menor será o custo para a sua correcção”, ver figura 2.24.

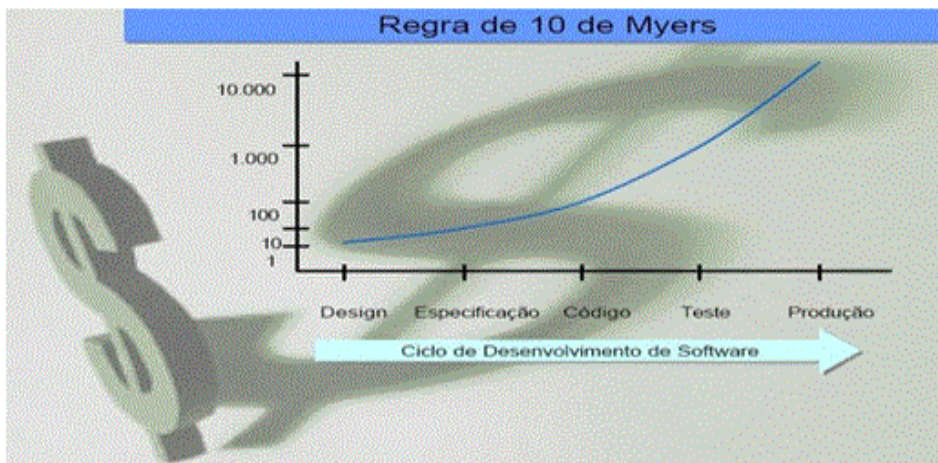


Figura 2.24 Regra de 10 de Myers, fonte (Tamashiro, 2010)

O modelo em V revela-se apropriado quando não há nenhuma incerteza do que se pretende que o sistema seja ou faça.

2.4.4 O Modelo Mercado e o Modelo Híbrido

O Método Híbrido, proposto por Figueiredo (Silva & Figueiredo, 1994) para o desenvolvimento de *software* educacional, é inspirado nos métodos de Análise e Projeto Estruturados de Eduard Yourdon (Yourdon, Modern Structured Analysis, 1989), no Modelo do Mercado de Cross (Cross, 1990) e Minnken, Stenseth e Vavik (Minnken, Stenseth, & VaVik, 1988), nos princípios da Prototipificação Estruturada (Booch, 1991). Articula componentes das concepções pedagógicas e informáticas. Sendo considerados os métodos de concepção informática os que são virados para a programação por oposição aos métodos que são dirigidos para a situação de ensino/aprendizagem, o que estabelece a ligação entre as fases da concepção pedagógica e da implementação.

O Método Híbrido compreende 3 fases cíclicas:

- “Uma fase de concepção pedagógica, sustentada por uma adaptação do Modelo do Mercado e uma versão do conceito da Tabela de Responsabilidade associado a este modelo.
- Uma fase intermédia, que abraça técnicas da Análise e Projeto Estruturados, como sejam os Diagramas de Fluxo de Dados, os Diagramas de Entidade-Relacionamento, os Diagramas de Estrutura, as Tabelas de Decisão, e os Dicionários de Dados, associando-lhe novas abordagens, tais como, Dicionário de Contexto, a construção de um Protótipo, a elaboração de uma proposta de Utilização e a discussão de uma Versão Preliminar.
- Uma fase de implementação, desenvolvida de forma engrenada nas duas fases anteriores.

Trata-se de um método interativo na sequência segundo a qual se cumprem as fases, e interativo no apelo que faz à colaboração dos diversos intervenientes no processo. Os passos a seguir para o Método Híbrido estão representados na figura 2.25.

O Método Híbrido permite alterações na ordem proposta, e a não realização de um ou mais passos propostos, assim como a introdução de outros passos caso o *software* em concepção o necessite.

1. Ideia
2. Título
3. Objectivos
4. Campo de Aplicações
5. Metáforas e Vivências
6. Tabelas de Responsabilidade
7. Tabelas de Responsabilidades Individual
8. Tabelas de Responsabilidades Orientadas
9. Diagramas de Mercado
10. Ecrã Chave
11. Ecrãs Secundários
12. Descrição de Ecrãs
13. Diagrama de fluxo de Dados
14. Diagramas de Entidade Relacionamento
15. Dicionário de Dados
16. Tabelas de Decisão
17. Árvore de Decisão
18. Caixa de Diálogos
19. Dicionário de Conceitos no Contexto
20. Ficha de Utilização
21. Protótipos
22. Preliminares
23. Versão Beta
24. Software

Figura 2.25 Modelo Híbrido, fonte (Silva & Figueiredo, 1994)

Este método é considerado dinâmico, pois permite a circulação de informações entre os vários intervenientes, engloba preocupações pedagógicas e informáticas, utiliza técnicas e ferramentas de engenharia de *software*, e considera as situações de campo, isto é verte a realidade entre o processo ensino/aprendizagem e o software interativo educacional ou software interativo educativo, criando o elo de ligação entre a conceção e a implementação.

2.4.5 Processo de Desenvolvimento User Interface Design (UID)

Como refere Thiago Amorim (Amorim, 2009) citando Luciano Gamez (Gamez, 1998) “A concepção de software adapta às características de funcionamento do homem atende a dois aspectos. Sendo o primeiro, a adaptação às características psicológicas gerais do ser humano, levando-se em conta as mais desfavoráveis situações de trabalho. Neste sentido, procura-se uma adequada apresentação das informações no ecrã, evitando dificuldades na percepção visual ou na compreensão da lógica de utilização. Da mesma forma, procura-se limitar as solicitações da memória de curto termo e respeitar os estereótipos mais corriqueiros. O

segundo é a adaptação do sistema de apresentação da informação e de diálogo à dinâmica das acções do utilizador. Este aspecto não pode ser abordado apenas a partir de uma consideração de dados e recomendações disponíveis na literatura. Ao contrário, trata-se de, para cada situação de trabalho, a ser concebida, procurar meios de prever as características respeitantes à estrutura da acção futura dos utilizadores”.

O segundo aspeto, referido no parágrafo anterior, está diretamente relacionado com a interação homem-máquina, visando este auxiliar o utilizador para atingir uma maior eficiência no desempenho das suas tarefas e das suas realizações interpessoais. É necessário identificar as informações necessárias para essa interação, quais as formas apropriadas e os métodos de análise e classificação das mesmas.

O processo *User Interface Design* (UID) está direcionado para a citação referida por Amorim. A figura 2.26 representa todo o processo de desenvolvimento de qualquer *software* ergonómico, pois tem em conta o utilizador final.

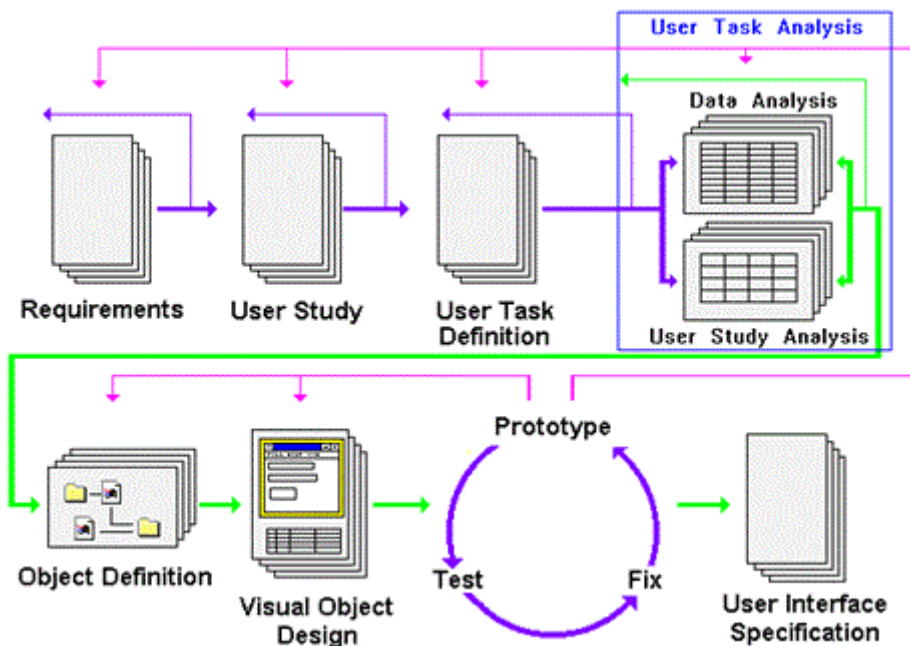


Figura 2.26 Processo UID, 1991-1999 Interface Systems international Inc Last Modified: March 20, 1999, (ISID, 1999) , referido por (Rodrigues & Moreno, 2006)

De acordo com a figura 2.26, as zonas “Requirements” e “User Study” dizem respeito aos requisitos não funcionais e funcionais, não esquecendo estes as características psicológicas da

população em questão. A inter-relação destas zonas, origina a zona “*User Task Definition*”, o que corresponde ao primeiro aspeto referido por Thiago Amorim (Amorim, 2009).

Toda a informação contida na zona “*User Task Definition*” é analisada posteriormente na perspetiva de Análise de Sistemas de Informação (ASI), o que corresponde à zona “*User Task Analysis*”; esta por sua vez é composta pela zona de “*Data Analysis*” e “*User Study Analysis*”, onde se aplicam as metodologias de ASI. Esta zona corresponde ao segundo aspeto referido por Thiago Amorim (Amorim, 2009).

Referimos que o segundo aspeto citado por Thiago Amorim (Amorim, 2009) pode e deve ser dividido em quatro fases. A primeira fase está descrita no parágrafo anterior. A segunda fase corresponde às zonas “*Object Definition*” onde se define a linguagem de programação a utilizar e o porquê. Enquanto a zona “*Visual Object Design*” corresponde à criação dos protótipos e à sua sequência.

A terceira fase corresponde ao ciclo “*Prototype*”, “*Test*” e “*Fix*”, onde são efetuados todos os testes e alterações necessárias ao *software* em causa. Esta fase termina aquando os requisitos funcionais e não funcionais estejam totalmente abrangidos.

Finalmente, surge a quarta fase, que define as tarefas dos utilizadores, tendo esta como objetivo verificar qual o utilizador que vai realizar determinada tarefa na aplicação, e assim verificar quais os requisitos para tal ato.

Para tal, equipas de especialistas em fatores humanos devem ser integradas nas equipas de análise e desenvolvimento de *software* interativo para prestar auxílio na análise de tarefas e nas características dos utilizadores.

2.4.6 Modelos de Desenvolvimento Evolutivos

Os modelos referidos anteriormente, pertencem à classe dos modelos de desenvolvimento clássico, que não contemplavam a natureza evolutiva e iterativa do desenvolvimento de *software*. Os modelos seguintes consideram estas questões.

Modelo em Espiral

O modelo em Espiral de Barry Boehm (Boehm B. , 2000) combina características do modelo de queda de água e do modelo de prototipagem. Introduce o “Planeamento” e o novo conceito a “Incerteza”. É considerado um meta-modelo, porque é um produtor de modelos de processos, como se pode ver na figura 2.27.

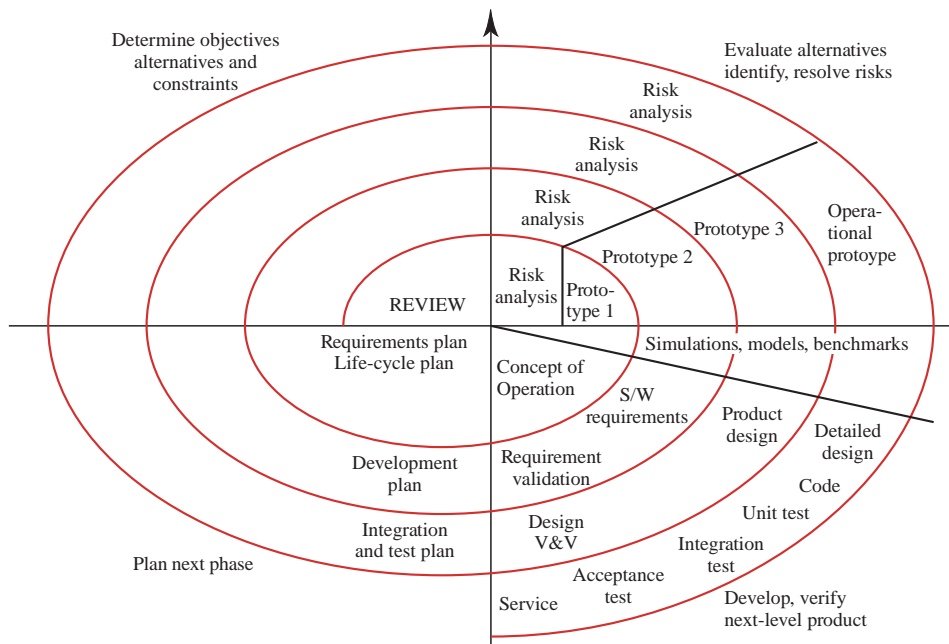


Figura 2.27 Modelo em Espiral, fonte (Sommerville I. , 2001)

Neste modelo, o processo começa no interior da espiral e vai crescendo no sentido dos ponteiros do relógio para o exterior. Cada uma das voltas da espiral corresponde a uma fase, estas estão divididas em quatro sectores, de acordo com a figura 2.27.

No sector “*Determine objectives alternatives and constraints*” são definidos os objetivos para cada fase do processo. É efetuado um plano minucioso de gestão e projetados planos alternativos, caso seja necessário.

O sector “*Evaluate alternatives identify, risks*” efetua uma análise dos riscos, com o objetivo de minimizá-los. O custo de um ciclo é determinado pela dimensão do problema nesse ciclo. Ou seja, a dimensão do raio mede os custos ocorridos até ao momento.

No sector “*Develop, verify, next-level product*” é selecionado o modelo de desenvolvimento de *software*, de acordo com a análise de riscos.

O sector “*Plan next phase*” efetua as revisões do projeto e verifica a necessidade de um novo ciclo. Caso seja necessário, é efetuada uma planificação para o próximo ciclo.

Para uma apreciação correta dos riscos e segundo Francisco Duarte (Duarte, 2002) deve-se ter em conta os seguintes pontos de análise:

- Objetivos - intentos da análise;
- Restrições – fatores que confinam as possibilidades;
- Riscos - incertezas possíveis para as alternativas identificadas;
- Resolução de riscos – estratégias adquiridas para a redução dos riscos identificados;
- Resultados - efeitos obtidos após a resolução dos riscos;
- Planos – planificação para a próxima fase da análise;
- Compromissos - Decisões tomadas pela gerência, sobre como continuar.

Observemos que todo este processo é orientado para os riscos em decréscimo do produto.

O risco é calculado pela (probabilidade de existir) X (gravidade dos efeitos). Estes componentes são avaliados qualitativamente mediante os critérios “muito baixo, médio, alto, muito alto”.

Todos riscos identificados devem ser analisados, segundo: o identificador; a causa; a probabilidade de ocorrência; os efeitos e a gravidade; a classificação e as ações de contingência.

Este modelo é mais versátil perante as mudanças e situações problemáticas são descobertas mais cedo.

Segundo Roger Pressman (Pressman R. , 2000) “*o modelo em espiral é de aplicação complexa, são necessários alguns anos de prática até se poder aplicar o modelo com eficácia*”.

Modelo de Entrega Incremental

O modelo de entrega incremental cruza características do modelo em V e do modelo de prototipagem iterativa.

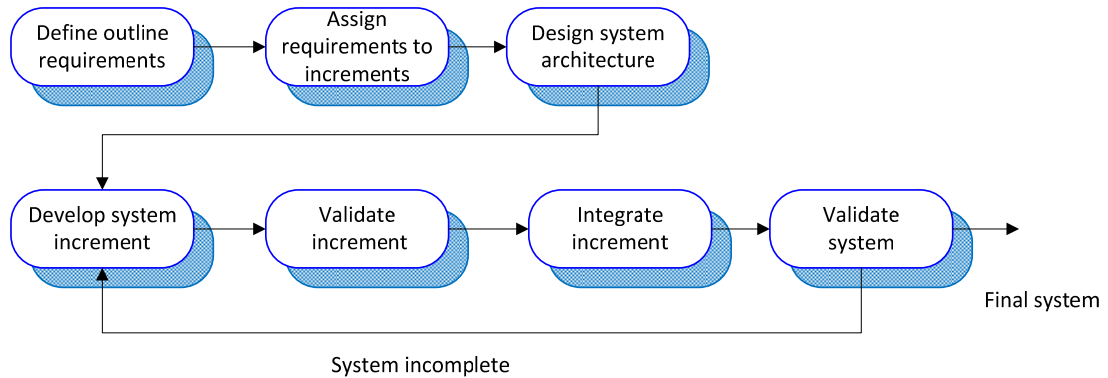


Figura 2.28 Etapas do Modelo de Entrega Incremental, (Sommerville I. , 2001)

O modelo em causa tem a vantagem de permitir que o cliente identifique quais as funcionalidades mais importantes que serão desenvolvidas e entregues pela ordem acordada (incrementos) e colocadas em produção.

Segundo António Miguel (Miguel, 2006) o sucesso deste modelo depende da arquitetura criada pelo engenheiro de sistemas. Se a arquitetura criada, visionar a funcionalidade final para o sistema, e garantir que a adição do próximo incremento de funcionalidade não origina uma reengenharia de sistemas ao nível da arquitetura, o sucesso é garantido.

Este modelo é útil quando não existe uma especificação dos requisitos e a dimensão do sistema a desenvolver é pequena. Normalmente, o primeiro incremento é desenvolvido sob os requisitos básicos, visto que os restantes requisitos ainda estão por definir. Após o desenvolvimento do primeiro incremento é promovida uma reunião como o cliente/utilizador, este avalia o incremento apresentado e define-se o que se pretende para o próximo incremento. Este processo repete-se até que o sistema esteja completo. É de reparar que após cada reunião entre a equipa de desenvolvimento e o cliente/utilizador torna-se a refazer a análise de sistemas de informação (figura 2.28), que engloba as áreas de especificação e de desenho.

É óbvio que ao refazer a análise torna-se extremamente difícil saber o tempo que o projeto vai demorar, assim como o número de iterações necessárias.

Por outro lado, através deste modelo o cliente/utilizador tem um papel crucial no desenvolvimento de uma aplicação. Pois ele acompanha todo o processo passo a passo e a sua opinião é válida durante todo o processo.

A figura 2.29 representa a sequência de passos necessários para o desenvolvimento dos incrementos (funcionalidades) pretendidos para um determinado sistema.

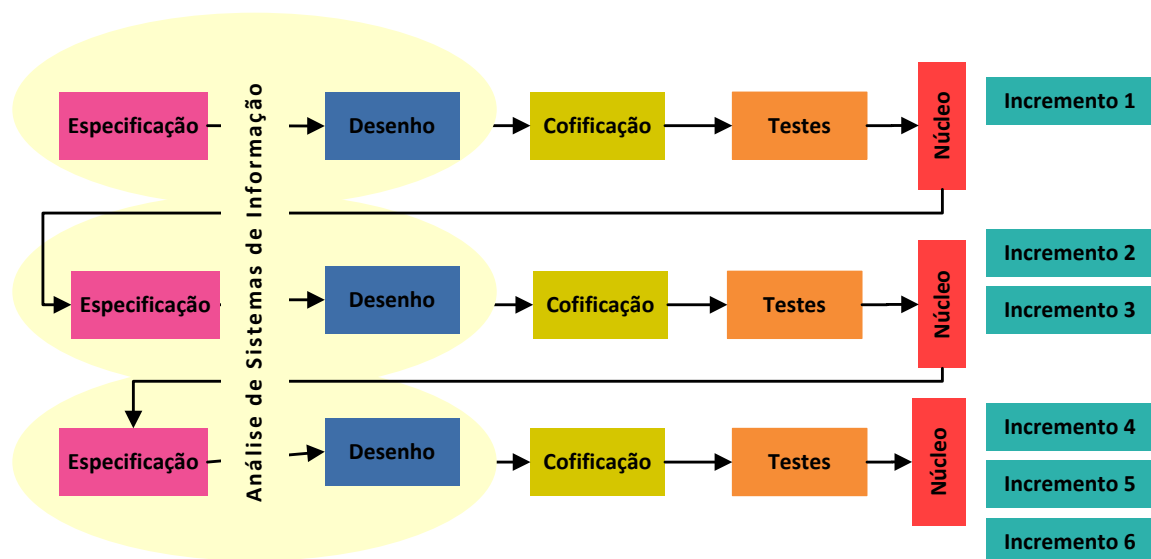


Figura 2.29 Modelo de Entrega Incremental, adaptado de ((Miguel, 2006), (Desconhecido, engenharia de software))

2.4.7 Metodologias de Desenvolvimento Ágil

Questionemo-nos acerca da agilidade. O que é agilidade?

Segundo Steven Goldman *“Agilidade é um termo dinâmico, utilizado dentro de um contexto, agressivamente relacionado com a aceitação de mudanças e sempre orientado para o crescimento (...)”*, cita Otávio Santos (Santos O. A., s/data).

Segundo Roger Pressman (Pressman R. S., 2005) os princípios das metodologias de desenvolvimento ágil focam:

- Indivíduos e as iterações;

- Participação do cliente em vez de negociação de contratos;
- Funcionamento do *software* em vez dos documentos abrangentes;
- Motivação da equipa de desenvolvimento e dos clientes;
- Desenvolvimento sustentável;
- Simplicidade tendo em vista a excelência;
- Equipas de desenvolvimento auto-organizadas e efetivas;
- Resposta a modificações em vez de seguir um plano,

Nesta metodologia, o cliente tem um papel ativo, iteração a iteração, pois é ele o elemento decisivo na definição dos novos requisitos. Esta prática caminha no sentido oposto das metodologias tradicionais, onde tudo deve ser planificado e acordado no início do projeto. De realçar que esta metodologia realça as pessoas e não os processos.

No entanto, como nada é conjeturável e os projetos não são exceções, na maioria das vezes, sofrem mutações. As alterações num projeto devem-se a vários fatores, tais como nível dos requisitos, alterações na equipa, etc. Assim, perante estas mudanças, os sistemas devem ser rápidos a agir, de uma forma simples e eficiente. Para tal, tem de haver uma maior adaptabilidade e flexibilidade às alterações independentemente do tipo de alterações.

Referimos que esta metodologia assenta nas pessoas e não nos processos. Assim sendo a equipa de desenvolvimento tem toda a importância e deve ter características especiais, Roger Pressman (Pressman R. S., 2005). As particularidades desta equipa devem ser as seguintes, segundo Mário Tomás (Tomás, 2009):

- Competência – A equipa deve ter conhecimento das metodologias tradicionais como das metodologias ágeis e conhecimentos específicos relacionados com o *software*.
- Foco Comum – Todos os elementos da equipa devem ter diferentes competências e conhecimentos, mas todos devem ter o mesmo objetivo, que é entregar uma funcionalidade (um incremento) ao cliente a funcionar e no prazo acordado;
- Colaboração – O chefe de projeto, os restantes elementos da equipa e o cliente têm de colaborar uns com os outros de forma a analisar e avaliar a informação eficiente e eficazmente;

- Capacidade de tomada de decisão – A equipa deve ter autonomia para tomar decisões a nível técnico e de projeto;
- Habilidade de resolver problemas imprecisos – A equipa deve ser capaz de resolver qualquer tipo de problema e não se esquecer que já resolveu aquele enigma, mesmo que no futuro não tenha significado;
- Respeito e confiança mútua – A equipa tem de funcionar como um todo, e deve-se tornar “*tão fortemente aglutinada que o todo é maior que a soma das partes*”;
- Auto-organização – A equipa ágil organiza-se para o trabalho ser feito, organiza o processo para melhor acomodar seu ambiente local e o cronograma de trabalho que conseguir melhor entrega do incremento de *software*.

De seguida, iremos apresentar os processos mais conhecidos que sustentam esta metodologia.

Desenvolvimento Adaptativo de Software (ASD)

A metodologia de desenvolvimento adaptativo de *software* (*Adaptive Software Development - ASD*), proposta por Jim Highsmith, baseia-se na colaboração de pessoas e na auto-organização da equipa cita Roger Pressman (Pressman R. , 2005), de acordo com a figura 2.30. Tem como ações a especulação, a colaboração e a aprendizagem.

Na especulação inicia-se a análise de sistemas, focando os requisitos iniciais, as restrições do projeto e as premissas do cliente.

A ação colaboração está relacionada com o fator motivacional. Tem como objetivo o trabalho em grupo e como tal todos os elementos da equipa devem direcionar esforços na obtenção dos requisitos.

A aprendizagem é vista sob três aspetos relacionados com o produto final. O primeiro aspeto foca os grupos de trabalho ao analisar o feedback sobre o produto. O segundo aspeto tem como objetivo as revisões técnicas formais e está direcionado para a qualidade do produto. Por último, a conclusão que analisa o próprio desempenho.

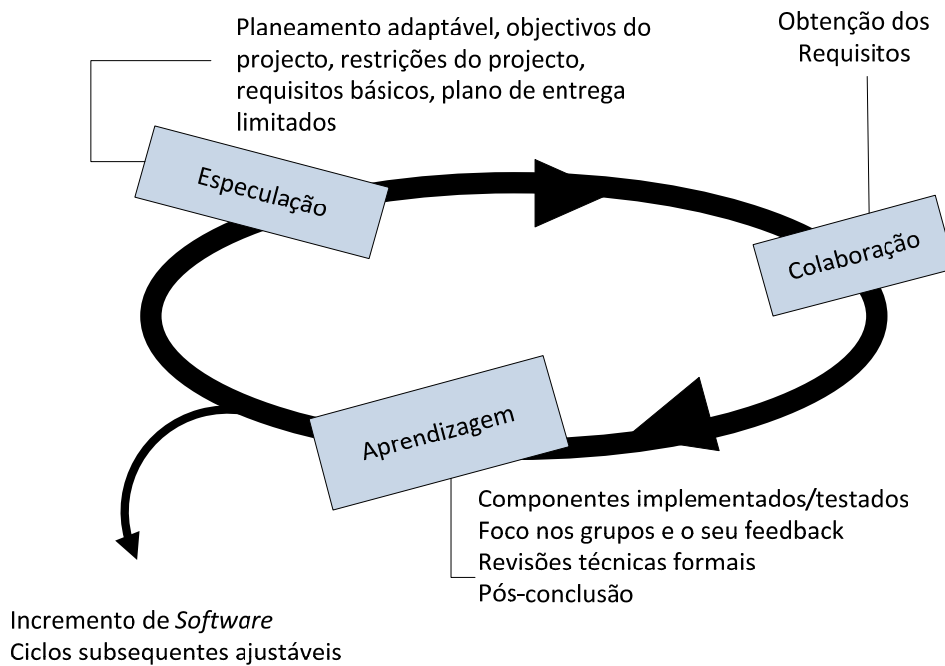


Figura 2.30 Desenvolvimento de Software Adaptativo, (Pressman R. , 2005)

Extreme Programming (XP)

A *Extreme Programming* utiliza uma abordagem orientada a objetos como paradigma de desenvolvimento. O seu processo é composto pelas atividades planeamento, desenho, codificação e teste, estas são repetidas iteração a iteração Roger Pressman (Pressman R. S., 2005), de acordo com a figura 2.31. A filosofia inerente a cada uma destas atividades é a seguinte:

- Planeamento – Começa com a criação dos objetivos (histórias) pelo cliente, que descreve as características e funcionalidades pretendidas para o *software* a desenvolver. A equipa analisa a lista dos objetivos, sob duas perspetivas: primeira - caso o objetivo precise de mais três semanas de desenvolvimento, é pedido ao cliente que o divida em objetivos mais pequenos; a segunda perspetiva é atribuída um custo ao (s) objetivo (s), de acordo com o número de semanas. A decisão da ordem de implementação e da data de entrega de cada um dos objetivos é definida pela equipa e pelo cliente, iteração a iteração. Após a entrega da primeira fase do projeto, deve-se calcular a sua velocidade de desenvolvimento, designada como *project velocity*. A velocidade do projeto

consiste em calcular o número de objetivos implementados durante esse tempo, pois pode haver a necessidade de se efetuarem ajustes no cronograma.

- Desenho – Baseia-se no princípio “*Keep it simple – KIS*”. Deve-se apostar numa estrutura simples em vez de uma estrutura complexa. O programador *developer* deve seguir as linhas orientadoras dos métodos de engenharia de *software*, e não desenvolver funcionalidades extras com designs próprios, que não são reconhecidos pelo cliente ou pelos utilizadores. A equipa de desenvolvimento deve usar o mecanismo *Class-Responsibility-Collaborator (CRC) cards*, protótipos para a resolução de objetivos problemáticos. O XP instiga o refazimento (*refactoring*), este é um processo que permite alterar e aperfeiçoar o sistema de *software* interno, sem que se altere o comportamento externo.
- Codificação – O XP aconselha que a codificação seja efetuada em pares, pois promove a rapidez e a qualidade. Devem criar-se scripts de testes unitários para cada objetivo. Ou seja, existe uma submissão do código ao teste unitário.
- Testes – Os testes unitários são conservados ao longo de todas as iterações, posteriormente são englobados para serem testados periodicamente (intervalo de horas, no final do dia, no final da semana).

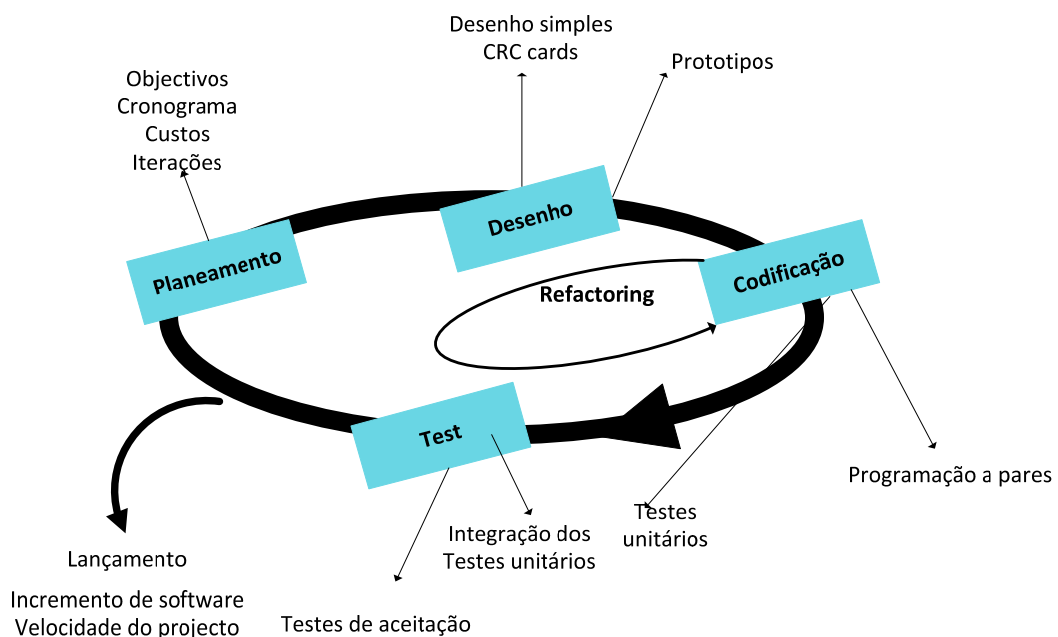


Figura 2.31 *Extreme Programming Process*, fonte (Pressman R. , 2005)

Conclusões

Com este capítulo, pretendeu-se mostrar a importância e a complexidade envolvida no processo de desenvolvimento de *software* educativo. Este processo envolve o cumprimento de várias etapas no processo organizacional, a sistematização de critérios de aprendizagem e critérios de qualidade, o seguimento das recomendações ergonómicas, nas quais estão implícitas as recomendações para a qualidade do *software*.

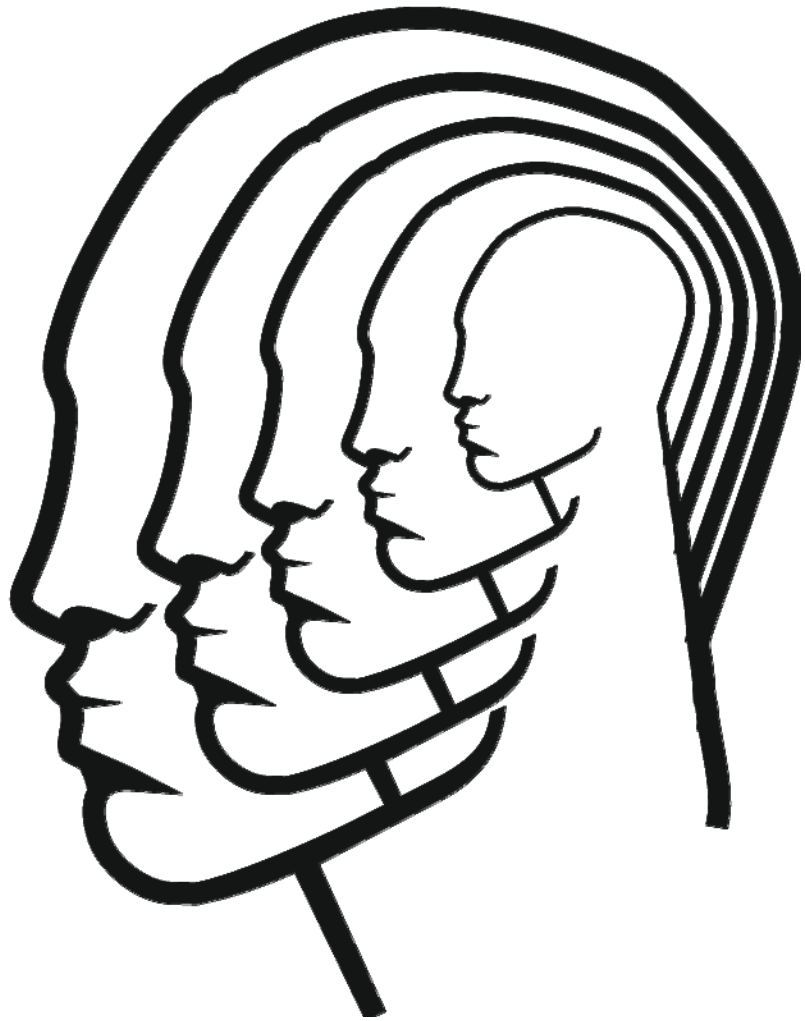
Por outro lado, ao analisarmos os processos de ensino aprendizagem, deparamo-nos com um conjunto de teorias que demonstram a evolução cognitiva dos indivíduos e qual o papel do transmissor de conhecimento nessa evolução. É de realçar que os estilos de aprendizagem não são o mesmo que os estilos cognitivos e tão pouco o mesmo que as inteligências múltiplas. Trata-se sim de teorias e conceitos que se interrelacionam.

Para nós, os estilos cognitivos são representados como maneiras típicas de perceber, recordar, pensar e resolver. Enquanto os estilos de aprendizagem são caracterizados pela forma pessoal de processar a informação, os sentimentos e comportamentos demonstrados na assimilação da informação, que se cruzam com os estilos cognitivos.

Esta parte do estudo foi de uma mais-valia em termos pessoais (conhecimento), visto que muitas destas teorias eram completamente desconhecidas para nós. Porém ajudaram-nos de forma muito significativa a entender e perceber melhor os passos que teríamos de efetuar para atingir os objetivos inicialmente propostos.

Após analisarmos esta panóplia de metodologias e teorias podemos concluir que avaliar a qualidade em produtos de *software* relacionados com o ensino, implica entender o processo ensino aprendizagem, para que se possa verificar a relação qualidade e satisfação do utilizador (aluno, professor).

Estudar e analisar as metodologias de desenvolvimento de *software*, metodologias ergonómicas e normas permitiu-nos conhecer e opinar sobre os pontos fracos e fortes de cada uma dessas metodologias.



Capítulo 3

Proposta do Novo Modelo

“Se eu tivesse oito horas para derrubar uma árvore, passaria seis horas afiando o meu machado “ Abraham Lincoln, refere (Molinari, s/data).

Desenvolver um novo modelo requer um estudo prévio a outros modelos, já que só assim é possível justificar as incapacidades dos modelos existentes face a determinados objetivos.

Sendo a população alvo deste estudo tão específica e o tema extremamente complexo, foi analisado em minúcia todo o Capítulo 2. Convém referir que, neste estudo, o grande desafio consistiu em relacionar essas metodologias e modelos, no processo ensino/aprendizagem “de forma interativa, ergonómica e adaptável” usando as tecnologias de informação e comunicação (TIC). Para que isso fosse possível, decidimos dividir essa análise em dois grupos: Grupo - Professor; Grupo - Aluno. Desta forma, foi possível verificar as necessidades específicas de cada um dos grupos e o que existia em comum entre eles.

Perante a análise de requisitos (a descrever no capítulo 4) concluiu-se:

- Os objetivos dos grupos eram diferentes, sendo eles:
 - Para o grupo professores – ensinar de forma interativa e diferenciada;
 - Para o grupo aluno – aprender.
- O que existia em comum nos grupos:
 - As TIC.

O resultado dessa análise originou um novo modelo de desenvolvimento de *software* educativo, denominado “ Modelo Interativo Adaptável Orientado para o Ensino”.

Antes de descrever todo o método inerente ao modelo, deve-se referir quais as metodologias e modelos de estudo que foram consideradas.

3.1 Metodologias Adotadas

“Abre também a outra folha para fazer entrar um pouco mais de luz”. (Arruda, 1998)

Para que o novo modelo apontasse para a “excelência” foram selecionadas algumas das metodologias e modelos analisados. Optamos por selecionar as metodologias que se ajustavam aos nossos objetivos. Porém, as metodologias de aprendizagem foram de difícil seleção. Selecionar entre teorias comportamentais e teorias cognitivas, quando aparentemente em todas elas, os processos cognitivos e metacognitivos estão presentes, foi uma tarefa difícil e exigente.

3.1.1 Teorias de Aprendizagem

“A educação do homem começa no momento do seu nascimento; antes de falar, antes de entender, já se instrui” de Jean Jacques (Rousseau, Émile ou De L' Éducation, 1966).

Para este estudo, as teorias do desenvolvimento cognitivo, teorias de desenvolvimento comportamental, teoria de inteligências múltiplas, a memória e os modelos mentais revelaram-se muito importantes. Como tal, o modelo desenvolvido contempla estas teorias e modelos.

É de referir, que consideramos este conjunto de teorias e modelos como um conjunto das teorias de aprendizagem, visto que contribuem para que o indivíduo evolua intelectualmente e socialmente. Ao adquirir novos conhecimentos desenvolve novas proficiências, o que origina novos comportamentos sociais. Para que possamos esclarecer os motivos da nossa escolha, apresentamos a tabela 3.1.

Teorias e Modelos de Aprendizagem	
Jean Piaget	A população do estudo situa-se na faixa etária dos 6 anos aos 10 anos, existindo algumas situações até aos 14 anos. A aprendizagem é diferente em todas as idades, o que aponta para Jean Piaget
Sócio Construtivismo de Vygotsky (Vygotsky, 2007)	Os estudos do sócio construtivismo de Vygotsky (Vygotsky, 2007) indicam que o aluno ao contactar com pessoas mais experientes (neste caso o professor) vê as suas capacidades cognitivas modificadas o que suscita novos conhecimentos. Situação também verificada pelo professor, o que origina novos exercícios para o aluno
Teoria Cognitiva	De recordar que a motivação está relacionada com a teoria cognitiva, logo, motivação está relacionada com a satisfação, a atenção, a ansiedade, o reforço e o feedback. Como tal, todo o processo educativo deve motivar o aluno para que ele possa aprender
Teoria Behaviour de Watson, Skinner e de Bandura, (Bandura, 1977)	Ao considerar a motivação como um processo e referir este como um conjunto de “caminhos” para obter um resultado, caminha-se no sentido da teoria behaviour de Watson, Skinner e de Bandura, (Bandura, 1977)
Teoria Comportamental (Weiner, 1990)	No entanto, Bernard Weiner, defensor da teoria comportamental (Weiner, 1990) realça que motivação pode ser dividida em intrínseca (realização pessoal) e extrínseca (realização através de recompensas)
	De salientar que tanto a motivação intrínseca como a motivação extrínseca, são contempladas no modelo proposto, através da interatividade como o aluno/professor e o sistema computacional;
Teoria de Inteligências Múltiplas	Howard Gardner (Gardner, 1995) ao evocar a teoria de inteligências múltiplas e ao considerar a inteligência como um “potencial biopsicológico” indica que qualquer uma das inteligências pode ser trabalhada e conseqüentemente melhorada, e esta depende de aluno para aluno. Portanto o professor deve ter em conta a aprendizagem diferenciada de forma a poder estimular as várias inteligências aluno a aluno (Antunes C., 2005)
A Memória	Independentemente da teoria considerada, o papel da memória no processo aprendizagem é relevante. Os resultados da aprendizagem são retidos na memória, este processo é efetuado por etapas, inicia-se na memória a curto prazo – a informação é armazenada por um período de alguns segundos, parte desta informação é passada para a memória a longo prazo. A memória a longo prazo permite conservar a informação adquirida durante dias, meses, anos ou durante toda a vida
	O mágico número 7 proposto por Miller (Miller, 1956) prova que o sistema cognitivo do homem só consegue processar um número limitado de informações que variam entre cinco (5) a nove (9) elementos por vez. Como tal, o conhecimento a transmitir tem de ser faseado, para posterior reconhecimento como “um todo”
Modelos Mentais	Os modelos mentais permitem analisar a mente humana e perceber como esta interage com os sistemas informáticos

Tabela 3.1 Teorias e Modelos de Aprendizagem

Portanto, o modelo desenvolvido, ao pretender “conectar” o sistema humano com os sistemas computacionais, tem em conta todos estes pontos, para decifrar e avaliar o código da mente.

De acordo Carlos Fuentes (Fuentes, 2003) “ *A memória é o género que se atreve a dizer o seu próprio nome. A bibliografia diz-nos: ‘És o que foste.’. O romance diz-nos: ‘És o que imaginas.’ A confissão diz-nos: ‘És o que fizeste.’ Mas biografia, confissão ou romance requerem memória, pois a memória, diz Shakespeare, é a guardiã da mente. Uma guardiã, diria eu, que se radica no presente para olhar com uma face o passado e com a outra futuro*”.

3.1.2 Metodologias de Avaliação de Software Educativo

“A avaliação diferencia os alunos desde aqueles que nada fazem até aqueles tudo sabem”, (Alcino, 1998)

Separar o ensino da avaliação é impraticável. São duas áreas que se moldam na procura da excelência, e nesta, a clareza dos objetivos tem de estar perfeitamente estabelecida. Perante tal, o modelo desenvolvido contempla algumas das metodologias referidas no capítulo 2 que passamos a apresentar.

A Taxonomia de Bloom revista por Roberto Marzano (Marzano & Kendall, 2000)

esenta três grandes áreas: cognitiva, afetiva e psicomotora, que se inserem nas características do Grupo – Aluno (descritas no Capítulo 1). A área afetiva está relacionada com a relação entre professor – aluno, em termos afetivos, interesses e atitudes ou valores. Em termos computacionais, a área afetiva é estudada na perspetiva da relação *software* - aluno e *software* – professor, tendo como características principais a interatividade, a usabilidade e a motivação.

De salientar, que tanto a Taxonomia de Bloom como a Taxonomia de Bloom Revista por Roberto Marzano (Marzano & Kendall, *Designing a New Taxonomy of Education Objectives*., 2000) não desenvolvem critérios na área psicomotora. No entanto, neste estudo, a área psicomotora é incluída através dos seguintes critérios: destreza de movimentos, percepção e

deficiência de visão. De acordo com estes critérios, houve a necessidade de recorrer ao uso de periféricos especiais, tais como:

- Computador de pressão variável – em substituição do rato normal;
- Teclado especial (aumentativo) – em substituição do teclado normal.

Convém recordar que cada categoria taxionómica representa o que o aluno aprende e não aquilo que ele já sabe, independentemente do local e da forma como o aluno adquiriu esse conhecimento.

Metodologia Pedagógica de Reeves

Thomas Reeves ao propor duas abordagens de avaliação de *Software* totalmente diferentes, fê-lo analisando separadamente os aspetos pedagógicos e de interface. Apesar de Reeves abordar de forma mais exaustiva os aspetos pedagógicos, estas duas avaliações complementam-se entre si.

Os aspetos pedagógicos propostos por Reeves abrangem a área da aprendizagem e a forma como os conteúdos didáticos se apresentam ao aluno. Como tal, houve a necessidade de ter esta metodologia em consideração e relacioná-la com a Taxonomia de Bloom.

Os aspetos de interface propostos contribuem para que se possa avaliar o ambiente textual, auditivo e gráfico de interação com o aluno. No entanto, esta análise não engloba de forma detalhada a interatividade e a usabilidade, que, no estudo em questão, foram englobadas nos modelos de qualidade de *software* ergonómico.

De referir que Reeves levanta algumas questões relacionadas com os aspetos pedagógicos, de aprendizagem e de interface. Segundo Reeves (Reeves, *Dimensões Pedagógicas Eficazes de Sistemas Interactivos de Aprendizagem*, 1994), *“deve-se diminuir a carga cognitiva apresentada nos ecrãs para permitir a atenção nos conteúdos de aprendizagem (...) a cor não deve ser uma variável importante, parece ser motivo de distração, mas pode ter efeitos secundários na atracção da atenção”*

As questões oriundas de Reeves foram foco de análise neste estudo e serão descritas no capítulo 4.

De acordo com este estudo, os critérios pedagógicos de Reeves eram de máxima importância, pois em conjunto com a Taxonomia de Bloom, iriam analisar a avaliar a evolução do aluno numa perspetiva mais correta, ou seja, mais próxima da realidade.

Modelo de Avaliação de Fernando da Costa, (Costa, 1999)

O Modelo de avaliação de Fernando da Costa, apresenta as seguintes características:

- Permite refletir sobre aspetos pedagógicos e didáticos e fornece informação diferenciada relativamente às características e potencial do produto;
- É composto por quatro quadros, cada um deles com objetivos bem definidos e diferenciados, de acordo com a tabela 3.2.

Quadro	Objetivos
QUADRO I	Está direcionado para os requisitos técnicos do <i>software</i> (ver anexo E)
QUADRO II	Está direcionado para os aspetos pedagógicos e de conteúdos (ver anexo F)
QUADRO III	Está direcionado para a interatividade entre o aluno e o <i>software</i> (ver anexo G)
QUADRO IV	Está direcionado para a usabilidade do produto (ver anexo H)

Tabela 3.2 Objetivos dos Quadros de Costa

- Este modelo integra uma nova noção “requisitos técnicos do *software*”, que permite saber em que sistema operativo o software pode ser executado, o equipamento requerido, informações técnicas e informações sobre o processo de instalação;
- Através destes quadros (Costa F. A., 1999), obteve a informação adequada para enunciar uma opinião crítica e fundamentada sobre um determinado *software* em análise.

Ficha de Avaliação de Software de Elsa Rodrigues, (Rodrigues, 2002)

A ficha de avaliação de *software* de Elsa Rodrigues (Rodrigues E. S., 2002), acrescenta algo de novo em relação ao modelo de Costa, de acordo com as tabelas 2.6, tabela 2.7 e o inquérito Avaliar (Ficha de Avaliação de Software Educacional Parte III) (ver capítulo 2).

Sendo a ficha de avaliação composta por três partes, cada parte contribui para a avaliação do *software*, da seguinte forma:

- É composta por critérios mensuráveis e não mensuráveis. Os critérios não mensuráveis foram avaliados pelo número de vezes que houve a mesma resposta;
- Os critérios de avaliações mensuráveis apresentam uma escala de avaliação compreendida entre zero (0) e cinco (5) de acordo com o descrito no capítulo 1.

3.1.3 Modelos de Qualidade

“A qualidade nunca se obtém por acaso, ela é sempre o resultado do esforço inteligente”,
John Ruskin (Ruskin, 2007)

Modelos McCall e Boehm

O modelo McCall faz a ponte entre os utilizadores e os analistas de sistemas e concilia o ponto de vista dos utilizadores com a prioridade dos analistas de sistemas. O modelo identifica três áreas de trabalho a avaliar no desenvolvimento de *software*, sendo elas:

- Comportamento do produto – requer que o *software* seja aprendido facilmente, seja eficiente e que os resultados sejam aqueles que o utilizador espera.
- Revisão do produto – está relacionada com a correção dos erros e com a adaptação do sistema.
- Transição do produto – embora possa não ser relevante em todas as aplicações, a sua transição para sistemas distribuídos e o rápido aumento na troca de *hardware* fazem com que esta área tenha a sua importância.

Para o estudo em causa, foi tido em conta a área do comportamento do produto, visto que os seus critérios (Usabilidade, Integridade, Eficiência, Retidão, Fiabilidade) são identificados nos objetivos do estudo.

O modelo Boehm é um modelo hierárquico. Baseia-se num conjunto de critérios, em número superior ao modelo de McCall, tais como: usabilidade, clareza, eficiência, a confiança, acessibilidade, concisão, legibilidade, comunicabilidade, teste, compreensão, alteração, etc. (ver capítulo 2)

Estes modelos possuem algumas características em comum, ambos se baseiam no ponto de vista do utilizador. No entanto, McCall distingue reutilizável de flexibilidade, enquanto Boehm indica modificável. Como tal, nenhum dos modelos pode ser considerado universal.

Contudo, como ambos os modelos têm o utilizador como ponto comum e os seus critérios estão direcionados para os critérios ergonómicos e para os objetivos do estudo, ambos os modelos foram adotados.

Modelos de Qualidade do Software Ergonómico

Os modelos ergonómicos adotados neste estudo foram os seguintes: *Human Computer Interface* e *Human Interface Designer*. O modelo *Human Computer Interface*, apresentado na figura 2.15, relaciona os processos cognitivos e os processos metacognitivos e o computador através da avaliação do utilizador. A evolução do utilizador é verificada através da sua avaliação perante o uso de um determinado *software*; este por sua vez deve-se adaptar às necessidades específicas do utilizador.

Logo, o *software* deve apresentar determinadas características ergonómicas. Para tal, nós temos em conta:

- As Oito Regras de Ouro de Shneiderman (desenvolvido no capítulo 2), (Shneiderman B. , 1998)
- Os 7 Princípios de Norman (desenvolvido no capítulo 2), (Norman D. A., 1990);
- Os Oito Critérios de Scapin & Bastien (desenvolvido no capítulo 2), (Scapin & Bastien, 1993);
- O modelo MoLic (Paula, barbosa, & Lucena, Conveying Human-Computer Interaction Concerns to Software Engineers through an Interaction Model, 2005), (desenvolvido no capítulo 2);
- As normas ergonómicas, para desenvolvimento de *software* (desenvolvido no capítulo 2), (International standards for HCI and usability, 2003):
 - A norma ISO/IEC 9126-1 - relaciona a fase do projeto e a fase de implementação
 - A norma ISO/IEC D TR 9126-4 - está relacionada com a qualidade do produto e as suas métricas

- A norma ISO 9241
- A norma ISO 14915 e IEC 61997
- A norma ISO/IEC 11581
- A norma ISO/IEC 10741

Todos estes princípios, modelos, regras e normas foram adotados visto que o novo modelo promove a qualidade, a interatividade, a usabilidade e uma aprendizagem diferenciada. Como tal, o modelo tem de permitir:

- A adaptabilidade para com o utilizador, de acordo com as necessidades do próprio (neste caso o aluno);
- A usabilidade tem de estar presente, ou seja deve ser de fácil uso;
- A interatividade tem de ser inerente ao próprio modelo.

3.1.4 Metodologias de Desenvolvimento de Software

“O nascimento de uma nova teoria rompe com a tradição da prática científica e introduz uma nova, o que se leva a cabo com regras diferentes e dentro de um universo de razões também diferentes” Thomas kuhn (kuhn, 1996).

Modelo Híbrido

O Modelo de Híbrido de António Dias de Figueiredo e Cristina Silva (Figueiredo & Silva, 1994) é dinâmico, espelha preocupações pedagógicas e informáticas, utiliza modelos e ferramentas da área da Engenharia de Software e de Análise de Sistemas de Informação, considera as situações de campo e representa a ponte entre a conceção e a implementação de um sistema informático.

Este modelo é composto por três fases:

- A fase de conceção pedagógica;
- A fase de engenharia de *software* e análise de sistemas;
- A fase de implementação

Segundo estes autores “ (...) trata-se de um método interativo na sequência segundo a qual se cumprem as fases, e interativo no apelo que faz à colaboração dos diversos intervenientes no processo”.

As características apresentadas pelo modelo Híbrido e os objetivos do nosso estudo apontam no mesmo sentido, assim foi considerado como um dos modelos a adotar. No entanto, nem todos os passos proposto no modelo foram considerados e a ordem proposta foi alterada, a apresentar no ponto 3.2.

Processo de Desenvolvimento UID e o Modelo HCI

O processo UID considera o utilizador como um elemento omnipresente durante todo o processo de desenvolvimento, desta forma alia-se com o modelo HCI que tem como objetivo compreender e simbolizar a interação entre o computador e o ser humano e como é que o conhecimento é difundido entre eles. Esta ligação é estabelecida através de determinadas zonas do processo UID (Análise de Requisitos, Análise ao Utilizador, Definição das Tarefas do Utilizador, Análise de Sistemas, Definição dos Objetos, Objetos Visuais de Desenho), conforme a figura 3.1.

De salientar que por si só, o nome do processo e o nome do modelo evocam a ergonomia, que evoca o homem e o trabalho, e que o trabalho se deve adaptar às características do homem. Toda esta situação caminha no sentido da opinião de Alain Wisner (Wisner, 1994) e Walter Cybis (Cybis, 1997) que a “*Ergonomia é uma ciência que procura a adaptação do ambiente técnico e organizacional ao homem*”.

Sendo um dos objetivos deste estudo a procura da adaptação do *software* ao conjunto aluno/professor no processo de aprendizagem, obviamente que o processo UID e o modelo HCI foram inseridos no modelo proposto. Para tal, houve a necessidade de os relacionar.

O relacionamento entre o processo UID e o modelo HCI foi efetuado de acordo com a figura 3.1, que representa as diferentes atividades existentes durante o processo UID e as zonas correspondentes aos processos cognitivos e aos processos computacionais do modelo HCI, ou seja, relaciona as características humanas e as características tecnológicas.

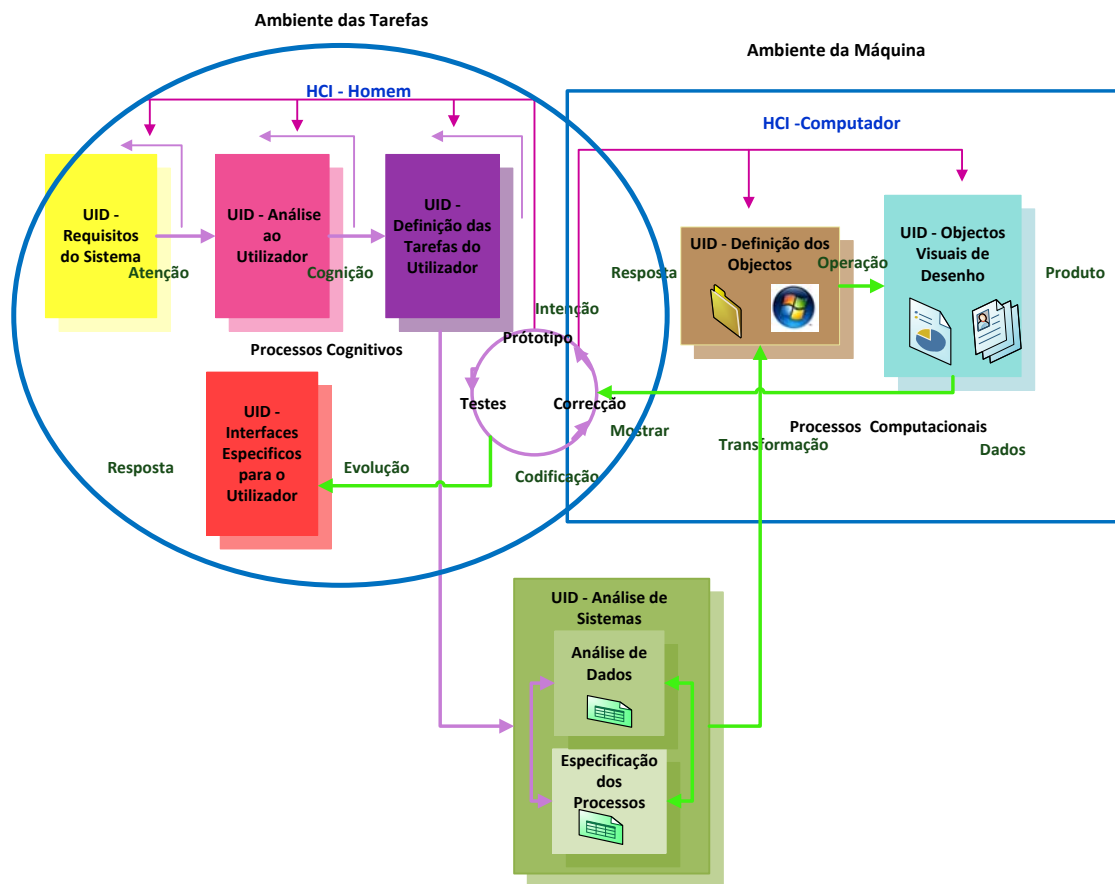


Figura 3.1 Conjugação do Processo UID e do Modelo HCI

A figura 3.1 apresenta diversas cores, com o objetivo de demonstrar que os dados sofrem alterações ao longo do percurso de desenvolvimento. O seu significado é descrito através da seguinte legenda:

- Setas de cor roxa - Recolha dos Dados e seus objetivos;
- Setas de cor verde - Tratamento dos dados e sua transformação;
- Seta de cor rosa - Dados a serem alterados;
- Círculo azul – representa o Modelo Interfaces Homem;
- Retângulo azul – representa o Modelo Interfaces Computador;
- Os processos UID apresentam diversas cores, de forma a ser mais fácil o seu reconhecimento de acordo com as suas mutações.

A conjugação do Modelo HCI com o Processo UID tem em consideração as diferentes capacidades apresentadas por cada ser humano (percepção, aprendizagem, memória) e como estas se comportam face ao uso dos sistemas informáticos. Nesta conjugação, os aspetos emocionais (sensações ou emoções despertadas) estão presentes através da zona Intenção, que se situa na zona de intersecção do modelo HCI – Homem e HCI – Máquina. Esta situação é analisada na perspetiva da interatividade, o que está de acordo com a visão de inteligência múltipla de Gardner que vai muito para além do fator quociente de inteligência como fator único e imutável.

De acordo, com o modelo HCI (figura 2.15) e com a figura 3.1 determinadas zonas para além de promoverem o quociente de inteligência podem promover o quociente emocional, o que aponta para Daniel Goleman (Goleman, 2002) *“Motivarmo-nos a nós mesmos (...) mobilizar as emoções ao serviço de um objectivo é essencial para concentrar a atenção, para a automotivação, para a competência e para a criatividade. O autocontrolo emocional – adiar a recompensa e dominar a impulsividade – está subjacente a todo o tipo de realizações. E ser capaz de entrar em estado de fluidez permite desempenhos de grande qualidade em todas as áreas”*.

Segundo Daniel Goleman (Goleman, 2002) *“ (...) Os psicólogos usam um ‘palavrão’ metacognição para significarem a consciência das próprias emoções (...) ”*, assim confirma-se que o Modelo HCI na zona Intenção, analisa as sensações ou emoções vividas pelos utilizadores perante a interatividade do *software*.

Contudo as sensações ou as emoções só podem ser analisadas no campo da observação, como tal não são mensuráveis, o que aponta para Elsa Rodrigues (Rodrigues E. S., 2002) e Fernando Albuquerque da Costa (Costa F. A., 1999).

Daniel Goleman (Goleman, 2002) considera que os cientistas cognitivistas se deixaram seduzir pelos sistemas computacionais *“ (...) como modelo operativo da mente e esqueceram-se que a rede cerebral está mergulhada num confuso e pulsante banho de neuroquímicos, sem qualquer relação com o esterilizado e ordenado mundo de silicone de onde saiu a metáfora para a mente”*.

Como se pode verificar através da figura 3.1 a concatenação do processo UID e do modelo HCI tem em conta metodologias de avaliação de *software* educativo, metodologias de análise e

metodologias de desenvolvimento de *software*. Ao relacionar a frase anterior “ (...) *a rede cerebral (...) ordenado mundo de silicone*” com a figura 3.1, verificamos que esta conjugação é extremamente complexa.

No entanto, não é impossível! Visto que, ao permitir avaliar o utilizador face à sua própria evolução, implica da parte deste um melhor desempenho. Por sua vez, o *software* tem de permitir maior interação, o que está de acordo com Alan Dix, Janet Finlay, Gregory Abowd e Russell Beale (Dix, Finlay, D.Abowd, & Beale, 2004) pois inclui a análise e conceção do processo de interação entre os utilizadores o *software* e o âmbito em que estes interagem.

É de referir que antes da zona Intenção, está a zona Cognição, que se refere aos processos pelos quais adquirimos conhecimentos através do raciocínio, obtenção de habilidades e desenvolvimento ou criação de ideias. É de recordar que a psicologia cognitiva é de extrema importância como elemento chave para o principal objetivo do modelo HIC que é estudar, analisar e antever o comportamento humano fase à interação homem-computador.

Porém, a zona Cognição é despertada pela zona Atenção, pois os nossos sentidos auditivos e visuais são motivados pelas imagens, cores e sons. No entanto, quando surgem demasiadas imagens, cores e sons, ou seja, excesso de informação, cria a desmotivação no utilizador perante o *software*.

Perante tal, deve-se evitar o excesso de informação, que segundo o psicólogo William James “*Todos conhecem o que é a atenção, é a tomada de posse da mente de uma forma clara e vívida para uma de entre diversas opções simultâneas de caminhos de pensamento. Requer desvencilhar-se de algumas coisas para conseguir lidar efectivamente com outras*”, (James, 1890), assim, deve-se usar o processo seletivo de atenção, ou seja, escolhermos prestar atenção a alguns estímulos em detrimento de outros.

Como tal, no desenvolvimento de “nosso *software*” analisamos quais os estímulos que chamam a “atenção” do utilizador, sem haver sobrecarga para o próprio, independentemente dos estímulos serem auditivos ou visuais. Esta situação é verificada através do paralelismo com processo computacional.

Na área do processo computacional, podemos verificar que o computador responde perante um conjunto de operações e estas, por sua vez, originam um produto - os dados, que

são transformados e apresentados através de ecrãs. De referir, que, perante as suas respostas se verifica a evolução do utilizador durante o processo “UID – Interfaces Específicos para o Utilizador”.

Para além dos factos apresentados, pode-se verificar que tanto o Modelo HCI como o Processo UDI têm como base a ergonomia, que evoca a adaptação do trabalho ao homem. Como tal, foi obrigatório adotar a conjugação do Modelo HCI e o Processo UID, conforme foi apresentado na figura 3.1, já que, um dos objetivos deste estudo foi a criação de um modelo que se adaptasse ao utilizador, independentemente das suas características interpessoais.

Assim, foi necessário distinguir os processos UID que fazem parte da área HCI – Homem e da área HCI – Computador. É de salientar que o processo UID - Análise de Sistemas não está incluído em nenhuma das áreas anteriores, porém, após a execução deste processo os dados são dirigidos para a zona HCI – Computador.

Após o desenvolvimento do produto são efetuados os testes, através dos protótipos, nos quais a HCI – Homem, volta a demarcar-se pela importância auferida na deteção de erros e de possíveis pressupostos não desenvolvidos. Esta situação origina a criação de interfaces específicas para o utilizador, de acordo com os objetivos pretendidos.

Caso haja, alterações, estas seguem dois sentidos (setas cor de rosa) HCI – Homem e HCI – Máquina. Se o sentido da seta for de “Testes” para “HCI – Homem” volta-se ao ponto de partida e percorre-se novamente a toda a figura 3.1. Porém, se o sentido for de “Testes” para “HCI – Máquina” só os processos computacionais são alterados.

Modelo em Cascata

Apesar de o Modelo em Cascata ser o modelo de desenvolvimento mais antigo é o mais utilizado. Apresenta uma abordagem sistemática e sequencial ao desenvolvimento do *software*. De salientar, que o modelo tem tido algumas alterações ao longo dos anos, de forma a prever a possibilidade de a partir de qualquer atividade do ciclo poder regressar a uma atividade anterior, para contemplar as alterações/problemas que tenham surgido durante o seu ciclo de vida.

Apesar de o modelo apresentar alguns problemas, tais como: a dificuldade do cliente em especificar os requisitos; disponibilidade do cliente aceder apenas à versão final do produto;

(entre outros), fornece um padrão no qual se encaixam métodos para engenharia de requisitos, análise de sistemas, implementação, testes e manutenção.

Ao observar a figura 3.1 pode-se verificar que se está a utilizar o conceito do modelo em cascata, não de uma forma linear, visto que, determinadas atividades estão a ser desenvolvidas paralelamente tais como: Análise de Requisitos, Análise ao Utilizador, Definição das Tarefas do Utilizador e posteriormente a Especificação de Interfaces de acordo com o utilizador, em oposição ao conceito do modelo em que uma atividade só se inicia quando termina a anterior.

Metodologia de Desenvolvimento Adaptativo de Software (ASD)

Esta metodologia faz parte das metodologias ágeis, apresentando como princípio fundamental as pessoas; o seu papel é dinâmico, empenhado, participativo em todo o desenvolvimento do *software*, Roger Pressman (Pressman R. S., 2005) .

De salguardar, que nestas metodologias o cliente faz parte da equipa, portanto ele é o elemento chave na definição dos requisitos e a sua participação é ativa.

Outro dos princípios das metodologias ágeis é a qualidade, obtida através da aprendizagem que está relacionada com o produto final. Por sua vez, a aprendizagem foca os grupos de trabalho e o seu *feedback* perante os componentes implementados e testados, ou seja, a própria equipa deve aprender/evoluir durante todo o ciclo de vida do projeto.

Como tal, as revisões técnicas e formais fazem parte da aprendizagem, estas podem ser efetuadas através de testes. Estes devem decorrer durante todo o processo de análise e de desenvolvimento de um projeto, princípio por nós optado.

Assim sendo, outra das características da aprendizagem é a conclusão, onde se analisa o próprio desempenho dos grupos de trabalho.

Esta metodologia funciona em ciclo e permite que ciclos subsequentes sejam ajustáveis de acordo com o incremento, situação por nós vivida.

Como esta metodologia foca a qualidade, permite inserir os modelos de qualidade, as normas de qualidade, os 7 Princípios de Norman, as Oito Regras de Ouro de Shneiderman e os Oito Critérios de Scapin & Bastien.

Os objetivos e a população deste estudo permitiram selecionar algumas das características desta metodologia, visto estarem direcionadas para a adaptabilidade e consequentemente para ergonomia.

Normas ISO

As normas por nós optadas foram as que apontam para a qualidade do produto: ISO/IEC 9126-1: Engenharia de Software Qualidade do Produto, Parte I: Qualidade do Modelo; ISO/IEC 9126-4: Engenharia de Software – Qualidade do Produto, Parte-4: Métricas de Qualidade.

3.2 O Novo Modelo Interativo Adaptável

“A mente que se abre a uma nova ideia jamais voltará ao seu tamanho original”, (Einstein, s/data)

Após todo o estudo desta panóplia de metodologias, tivemos que integrá-las, e extrair o que consideramos uma mais-valia em cada uma delas. Assim foi possível criar o Modelo de Desenvolvimento de *Software* Interativo e Adaptável. Este modelo é o reflexo de uma boa gestão entre a qualidade do produto e a qualidade do processo de desenvolvimento. De acordo com a norma ISO/IEC 9126 “é necessário um modelo que permita a avaliação da qualidade de *software*”, (Usabilidade, 2001).

Perante tal facto, este modelo responde a essa necessidade, pois permite que o *software* seja adaptável a qualquer utilizador. O modelo em causa é designado por “Modelo Interativo Adaptável” (figura 3.2) e incorpora o que foi descrito no ponto 3.1.

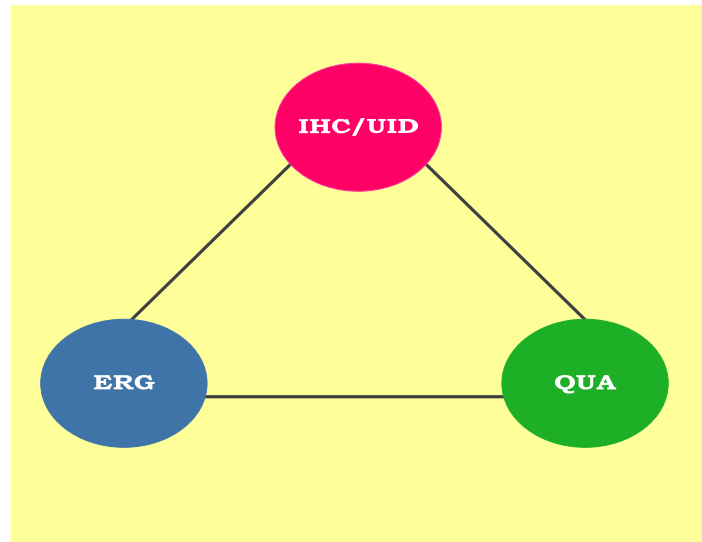


Figura 3.2 Modelo Interativo Adaptável (MIA)

O modelo apresenta um triângulo, com uma sigla em cada vértice, que por sua vez representam o conjunto de modelos e metodologias consideradas. A relação entre os vértices tem de ser de perfeita sincronia, para que o resultado final seja o referido no parágrafo anterior.

As diferentes cores representadas na simbologia do modelo - indicam os modelos e as metodologias consideradas em cada um dos vértices. Por sua vez, os vértices simbolizam “variáveis”, estas são denominadas como as “três variáveis chave” (IHC/UID,QUA,ERG) que devem estar em perfeita harmonia, ou seja, de forma equilibrada. Caso, haja uma alteração numa das variáveis, as restantes devem ajustar-se a essa alteração, para que voltem a equilibrar-se. Só perante o equilíbrio destas variáveis o objetivo do estudo é alcançado, originando o nosso modelo de desenvolvimento de *software*.

Para que possamos verificar a importância do equilíbrio no modelo, suponhamos que existe um conjunto de subvariáveis que vão ser alteradas, estas por sua vez, vão ter influência no comportamento do vértice IHC/UID. Por outro lado, estas subvariáveis vão ter influência no vértice ERG, o qual reage, originando uma resposta para com o vértice IHC/UID, através de subvariáveis. Em nenhum destes momentos, houve uma interação como o vértice QUA. Portanto, não se verificou se as alterações existentes mantêm os níveis de qualidade exigidos, o que levou a um desequilíbrio do modelo, como se pode verificar através da figura 3.3.

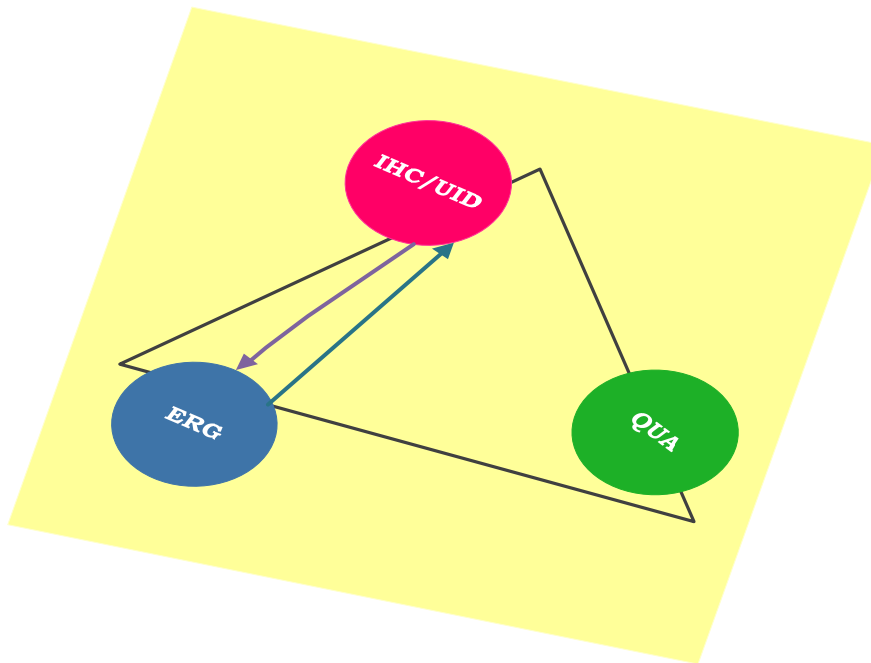


Figura 3.3 Modelo Interativo Adaptável em Desequilíbrio

Para que tal não aconteça, todos os vértices têm de obrigatoriamente interagir entre eles. Portanto, cada uma destas variáveis é composta por subvariáveis que se inter-relacionam com as subvariáveis dos outros vértices, através das arestas. Como tal, existe um cruzamento entre todas as variáveis e subvariáveis.

Para que se perceba de forma correta o modelo, começamos por identificar o que significa cada vértice, através da tabela 4.3.

Vértices	Significado
O vértice IHC/UID	Representa o Modelo de Interface Homem Computador e o Processo Desenho de Interfaces para o Utilizador, que por sua vez incorporam os modelos de desenvolvimento de <i>software</i>
O vértice QUAL	Representa a Qualidade de Software e engloba as metodologias de avaliação de <i>software</i> educativo, as metodologias de qualidade analisadas e adaptadas ao projeto, assim como as normas de qualidade para o desenvolvimento de <i>software</i>
O vértice ERG	Representa As Oito Regras de Ouro de Shneiderman, os 7 Princípios de Norman, os Oito Critérios de Scapin & Bastien, o modelo MoLic, as normas ergonómicas para desenvolvimento de <i>software</i> e os próprios IHC e o UID

Tabela 3.2 Significado dos Vértices no Modelo Interativo Adaptável

O modelo em causa tem por base o Método Interativo Adaptável Orientado para o Ensino (IAOE). Este método incorpora os conteúdos referidos nos vértices e também as teorias de aprendizagem selecionadas e referidas no ponto 3.1 e no Capítulo 2, representado pela sigla SIAOE. A figura 3.4 representa o Modelo Interativo Adaptável Orientado para o Ensino.

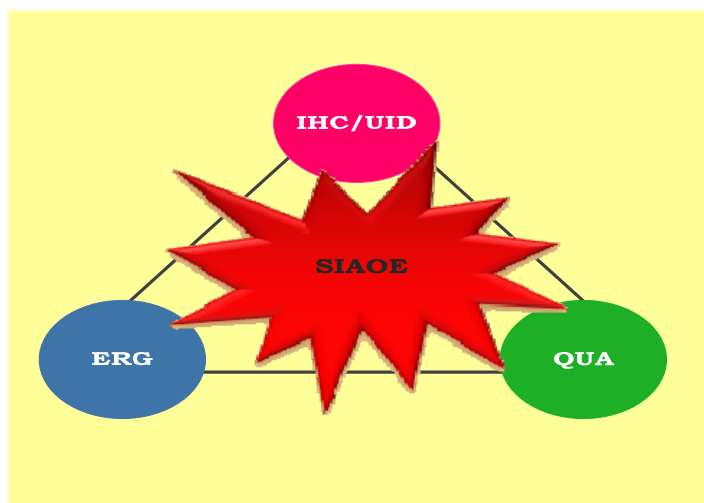


Figura 3.4 Modelo Interativo Adaptável Orientado para o Ensino

Para provar que o Modelo Interativo Adaptável Orientado para o Ensino funciona, foi necessário desenvolver o método que comprovasse o modelo, que designamos como Método Interativo Adaptável Orientado para o Ensino.

3.2.1 Descrição do Método Interativo Adaptável Orientado para o Ensino (MIAOE)

O Método Interativo Adaptável Orientado para o Ensino (MIAOE) tem um significado específico, sendo ele:

- Método – Ordem/Processo racional pedagógico na educação;
- Interativo – Interação entre o aluno/professor e o sistema informático;
- Adaptável – Capacidade de se moldar às necessidades individuais do aluno;
- Orientado Para – Capacidade de indicar a linha a seguir para alcançar o objetivo;
- Ensino - Capacidade de instruir.

O método é desenvolvido seguindo determinados passos, sequenciais ou não. Permite incorporar todas as fases de análise e desenvolvimento de um *software* de uma forma interativa,

possibilitando que as ideias e os processos sejam verificados testados e avaliados constantemente.

A nomenclatura do método desenvolvido é representada por retângulos, elipses, pelo seu relacionamento com o Processo UID e o Modelo HCI e pelas metodologias de desenvolvimento de *software*.

- Os retângulos representam as interações externas;
- As elipses representam os processos;
- As setas indicam o seu relacionamento as suas cores estão de acordo com as da figura 3.1.

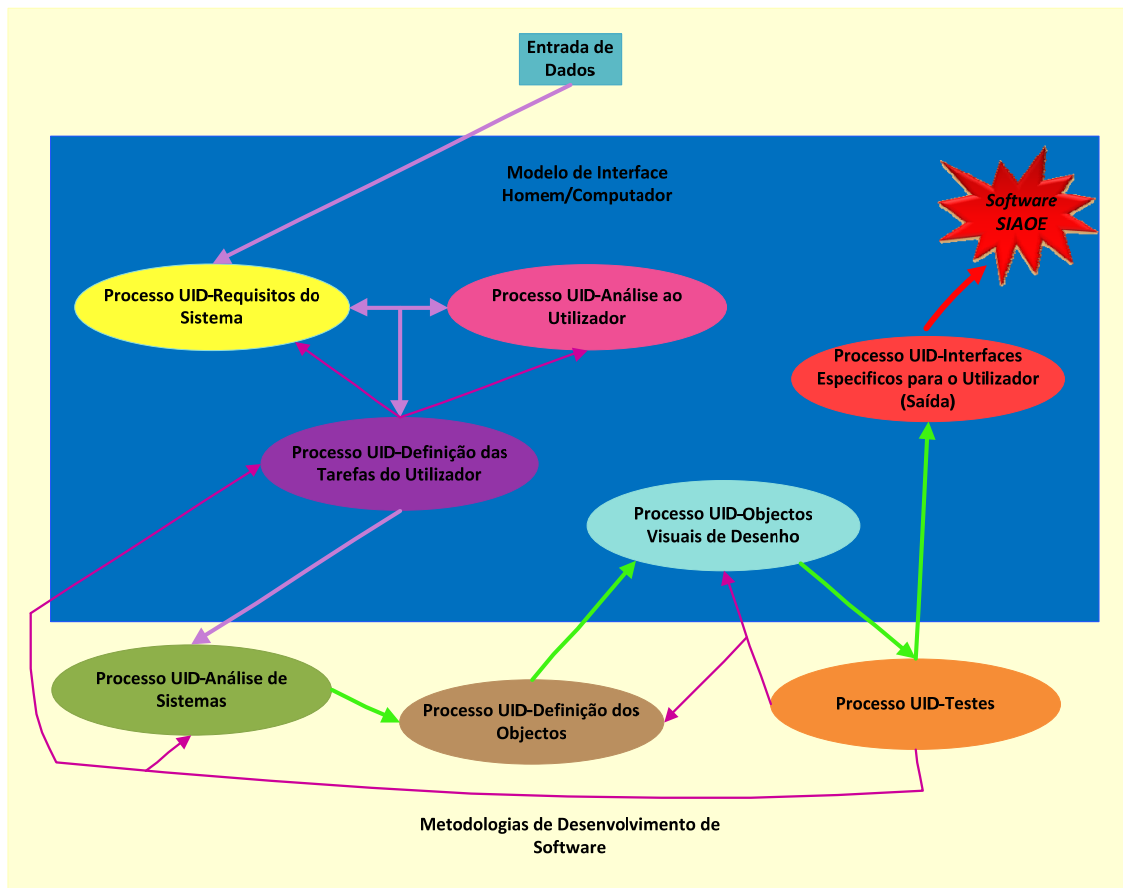


Figura 3.5 Método Interativo Adaptável Orientado para o Ensino

O “retângulo Dados” corresponde aos objetivos gerais do projeto, desde a sua identificação às perspetivas que se tem e a área que se pretende analisar.

O processo “UID-Análise ao Utilizador” indica o público-alvo a estudar e os objetivos pedagógicos. Enquanto o processo “UID -Análise de Requisitos” foca os requisitos funcionais e não funcionais. Estes dois processos têm de estar obrigatoriamente relacionadas um com o outro, vistos que os requisitos funcionais e não funcionais dependem do estudo em análise.

O processo representado pela elipse “UID-Definição Final das Tarefas do Utilizador” mostra a conjugação final do conteúdo das elipses anteriores, que reflete todas as tarefas que irão ser exigidas ao utilizador.

O processo “UID-Análise de Sistemas” indica as etapas a seguir no processo de análise, tais como: Análise de Dados; Diagrama de Fluxos de Dados; Modelo Concetual; Dicionário de Dados; Especificação de Processos (pode incluir Árvores de Decisão/Pseudocódigo/ Linguagem Estruturada, Casos de Uso).

Enquanto o processo “UID-Definição dos Objetos” refere a linguagem de programação a utilizar, o processo “UID-Objetos Visuais de Desenho” abrange a área dos protótipos de baixa e alta-fidelidade, descrição dos ecrãs, interação com o utilizador, modelos de qualidade do *software*, regras ergonómicas e as normas ISO.

O retângulo Modelo de Interface Homem/Computador está inter-relacionado com os processos” UID-Análise ao Utilizador”, “UID-Análise de Requisitos”, “UID-Definição Final das tarefas do Utilizador”, “UID-Objetos Visuais de Desenho”, visto que os aspetos cognitivos do ser humano, a interatividade entre o *software* e o utilizador têm de estar sempre presentes.

O processo “Testes”, indica a obrigatoriedade de se efetuarem testes ao *software*. Estes devem ser realizados em qualquer momento e com os utilizadores. Após a fase de testes existe a possibilidade de se efetuarem alterações em determinados processos. Essas alterações são apresentadas pelas setas com a cor rosa, no sentido do processo “Testes” para:

- O processo” Análise de Sistemas”;
- O processo “UID- Definição das Tarefas do Utilizador”, o que provoca alterações/verificações nos processos:
 - “UID-Análise de Requisitos e UID- Análise ao Utilizador”;
 - O processo ”UID-Objetos Visuais de Desenho”;
- O processo “UID-Definição dos objetos” .

Após se ultrapassar a fase de testes surge o processo “Interfaces Específicas para o Utilizador” e culmina com o *Software* Interativo Adaptável Orientado para o Ensino.

De salientar que alguns processos, podem ser divididos em subprocessos de acordo com o estudo em causa. Assim, descreve-se em maior detalhe os subprocessos. A figura 3.5 exibe todos os subprocessos intervenientes no desenvolvimento do próprio *software*.

Para melhor compreensão do método, descreve-se todas as suas etapas, as interacções externas e os seus componentes, assim como as processos e os subprocessos. A figura 3.6 representa em detalhe o Método Interativo Adaptável Orientado para o Ensino (MIAOE).



Figura 3.6 Pormenorização do Método Interativo Adaptável Orientado para o Ensino

A tabela 3.4 descreve as etapas Método Interativo Adaptável Orientado para o Ensino, como se pode verificar.

Em relação à interação externa entrada de dados, os primeiros três pontos são referidos modelo Híbrido de António Dias Figueiredo e Paula Silva (Figueiredo & Silva, 1994). Os restantes pontos (perspetivas, área) definem com maior detalhe o estudo em causa.

Descrição do Método IAOE	
A Interação Externa Entrada de Dados é composta por:	<ul style="list-style-type: none"> Ideia - planejar algo (um estudo, uma análise) Título - nome ou designação do estudo Objetivos - propósitos a alcançar no estudo Perspetivas - ter como prováveis determinados resultados referentes ao estudo Área - âmbito do estudo
Processo UID-Análise ao Utilizador	<ul style="list-style-type: none"> analisa o utilizador, os subprocessos correspondem às teorias de aprendizagem e às metodologias de avaliação pedagógica. A seleção das metodologias de avaliação depende do estudo em causa
Processo UID-Requisitos do Sistemas	<ul style="list-style-type: none"> define requisitos do utilizador para o sistema informático a desenvolver. O subprocesso , descreve os requisitos funcionais e os requisitos não-funcionais: Requisitos funcionais descrevem - serviços e as funções do sistema Requisitos não funcionais descrevem - constrangimentos em que o sistema deve operar (sistemas operativos, constrangimentos temporais) ou de desenvolvimento (línguas específicas)
Processo UID-Definição das tarefas do Utilizador	<ul style="list-style-type: none"> Reverte as tarefas do utilizador face ao Processo UID-Análise ao utilizador, ao Processo UID-Requisitos do Sistema e ao Processo Testes
Process UID-Análise de Sistemas	<ul style="list-style-type: none"> Procede do Processo UID-Definição das Tarefas do Utilizador, divide-se em dois subprocessos Análise de Dados - evoca a análise de sistemas, através da modelação de sistemas , representação abstrata do sistema cujos requisitos foram analisados e especificados Pormenorização da Análise de Dados - evoca a especificação de processos , que deve ser expressa de uma forma que possa ser entendida por todos os intervenientes na fase de análise
Processo UID-Definição dos Objetos	<ul style="list-style-type: none"> O analista de sistemas elege a linguagem de programação a utilizar face aos objetivos do estudo
Processo UID-Objetos Visuais de Desenho	<ul style="list-style-type: none"> divide-se em três subprocessos: O primeiro subprocesso engloba as metodologias de avaliação da qualidade de <i>software</i> (os modelos de McCall e Boehm) e as normas ergonómicas de desenvolvimento de <i>software</i> O segundo subprocesso engloba os protótipos de baixa fidelidade, descrição dos ecrãs e interação com o utilizador, através dos modelos de qualidade de <i>software</i> ergonómico, os 7 Princípios de Norman, as Oito Regras de Shneiderman e os Oito Critérios de Scapin & Bastien O terceiro subprocesso engloba os protótipos de alta-fidelidade
Modelo de Interface Homem/Computador	<ul style="list-style-type: none"> Estabelece ligações entre os processos UID-Análise ao Utilizador, UID-Requisitos do Sistemas, UID-Definição das tarefas do Utilizador, UID-Objetos Visuais de Desenho e UID- Interfaces Específicos para o Utilizador Estas ligações são estabelecidas o número de vezes necessário até se alcançar a desejada "excelência" de qualidade, donde $qualidade = qualidade^{n+1}$
Processo UID-Testes	<ul style="list-style-type: none"> O objetivo de se efectuarem testes ao sistema desenvolvido é de encontrar falhas/limitações no mesmo. Os testes devem ser efetuados durante o desenvolvimento do <i>software</i>; os elementos a participarem nos testes devem ser os utilizadores e a equipa de desenvolvimento do <i>software</i>. Os Testes devem contemplar os seguintes princípios: Planear o tipo de teste Planear em detalhe as atividades Definir o procedimento de testes Definir os resultados esperados Avaliar os resultados obtidos Melhorar o <i>software</i>
Teorias de	<ul style="list-style-type: none"> Englobam todas as teorias de aprendizagem referidas anteriormente.

Aprendizagem	
--------------	--

Tabela 3.3 Descrição do Método Interativo Adaptável Orientado para o Ensino

Conclusões

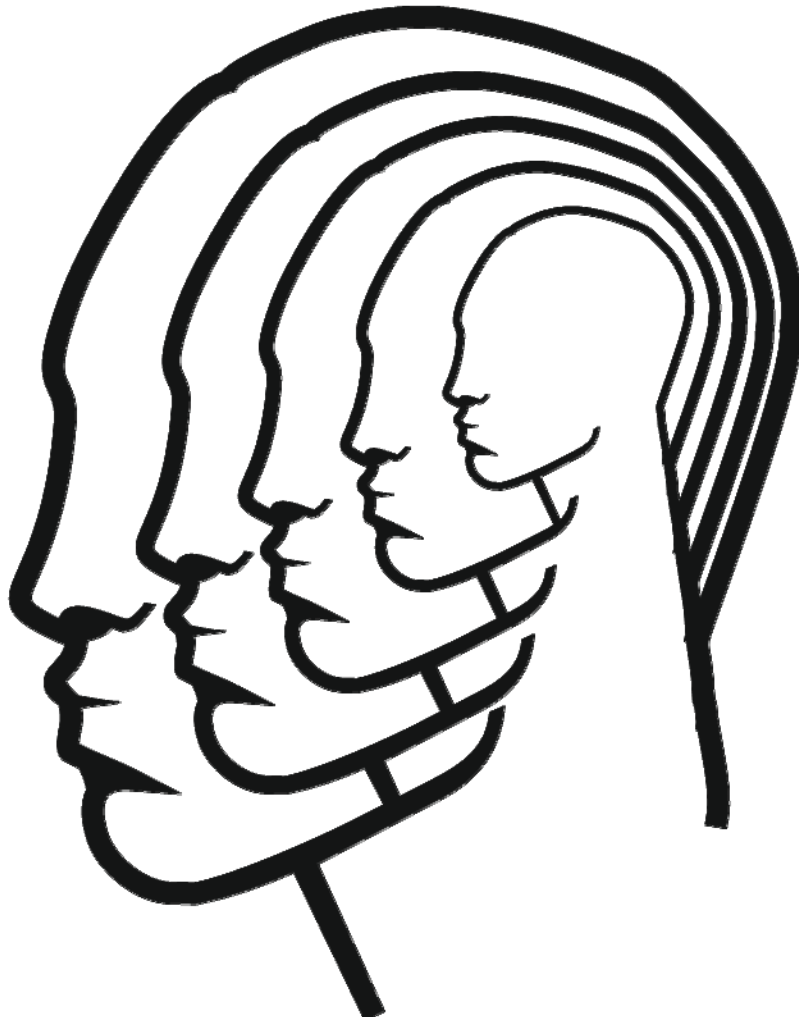
Desenvolver este modelo e o método permitiu adquirir um conjunto de conhecimentos inexistentes para além de proporcionar uma inter-relação entre nós e a população de estudo.

Este modelo de desenvolvimento de *software* surge pela necessidade de dar resposta às situações analisadas no projeto anterior, referidas no capítulo 1. Desta forma, foi possível verificar que os modelos existentes de desenvolvimento de *software*, não permitiam analisar a evolução cognitiva do utilizador/aluno.

Através deste modelo e do seu método é possível sistematizar um conjunto de atividades relacionadas com o processo de análise ergonómica das tarefas. Como tal, o método proposto permite desenvolver o Modelo Interativo Adaptável no processo ensino aprendizagem. Para tal, este método permite que o modelo se adapte ao utilizador qualquer que seja ele, tendo em conta as suas necessidades específicas. O qual, permite constar a capacidade evolutiva do aluno à medida que este aprende conceitos mais complexos.

O “Modelo Interativo Adaptável” e o método denominado “Método Interativo Adaptável Orientado para o Ensino” pretendem conciliar as contribuições pedagógicas que tanto o cognitivismo, sócio construtivismo, como o comportamentalismo, as teorias Behaviour, os modelos mentais, teoria das inteligências múltiplas, os estágios de Jean Piaget e o papel da memória oferecem, na ligação com os processos computacionais, incluindo estes os modelos de qualidade, modelos ergonómicos.

Devo salientar que não existe nenhum modelo ou método que responda de forma universal à panóplia de novas situações que surgem no dia a dia. Pois o modelo a utilizar no desenvolvimento de *software* depende dos objetivos e do ambiente onde decorre.



Análise Sistemático do Contexto

“(...) não podemos resolver os problemas de hoje com a mesma mentalidade com que foram criados”, de Einstein, referido (Regojo, 2011)

A abordagem sistémica visa conjugar conceitos de diversas áreas científicas a respeito de um determinado estudo de pesquisa. Esta abordagem parte do princípio que um objeto de estudo possui diversas dimensões e aspetos que podem ser estudadas por diversas ciências e que os conceitos e os princípios de diferentes ciências podem ser utilizados no estudo e compreensão de determinado fenómeno por determinada ciência, ou seja, qualquer descoberta que seja efetuada numa área de uma ciência pode ser utilizada por outra área de outra ciência. Portanto, estamos perante uma teoria interdisciplinar.

De acordo com a teoria interdisciplinar e como foi referido nos capítulos anteriores o nosso estudo está inserido nessa teoria e num determinado “contexto”. Porém, existem várias definições para o termo “contexto”, as quais dependem da sua área de aplicação. Como o “contexto” desempenha um papel relevante nas áreas da Psicologia Cognitiva, Linguística, Interatividade e Usabilidade, etc., foi por nós adotado, visto que o nosso estudo engloba todas estas áreas. O que nos permite utilizar este termo no “ contexto ensino”.

Segundo Teresa Rafael (Rafael, 2010) o processo de desenvolvimento de sistemas interativos foca cinco elementos chaves: a informação, a tecnologia, a interação, o utilizador e o contexto. É de realçar que (Rafael, 2010) não foca a adaptabilidade, no processo de desenvolvimento de sistemas interativos. Mas, adaptabilidade para o nosso projeto é de

extrema importância. Como tal, houve a necessidade de inserir o conceito da adaptabilidade no termo contexto, no processo desenvolvimento de sistemas interativos e provar a viabilidade do nosso método e metodologia criado no contexto do ensino.

Portanto, tivemos de inter-relacionar todos estes elementos no contexto do ensino, como se ilustra na figura 4.1 Como no nosso estudo o foco principal é o utilizador, decidimos que ele é o elemento central do núcleo que se inter-relaciona com os restantes elementos, representados nas diferentes camadas. Para nós, o contexto do ensino abraça e é abraçado pelos restantes elementos. Visto que a inter-relação entre estes elementos ocorre em ambos os sentidos.

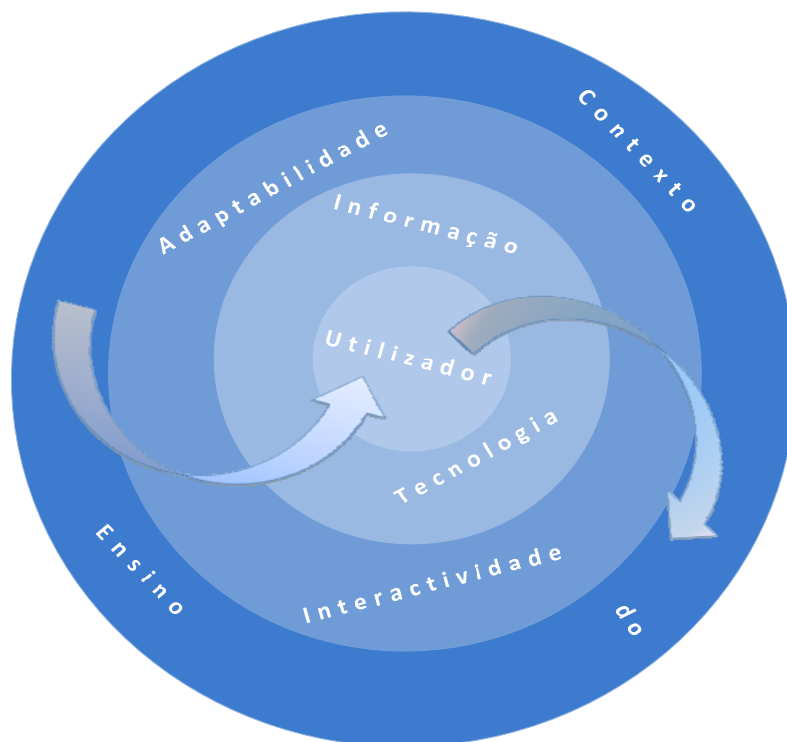


Figura 4.1 Interligação de Contextos

O conceito de contexto ao abranger a ciência da informação e a arquitetura da informação, toma dois papéis distintos. A ciência da informação tenta compreender o utilizador antes, durante e depois da sua interação com o sistema, enquanto, que a arquitetura da informação deve conhecer os seus utilizadores, o que os motiva e os seus comportamentos fase a determinadas situações, (Reis, 2007).

É de referir, que a motivação do utilizador em interagir com o sistema aponta para o contexto de uso. Assim, inter-relacionando a arquitetura da informação e a ciência da informação com o contexto de uso, entende-se que também se projetam quatro contextos:

contexto tecnológico, simbólico, social e cultural, menciona (Sousa, 2008) ao referir Frederick Van Amstel.

De acordo com Amstel e referido por (Sousa, 2008) existe um modelo de definição de contexto de uso que envolve as quatro contextos mencionados no parágrafo anterior, como se apresenta na figura 4.2.

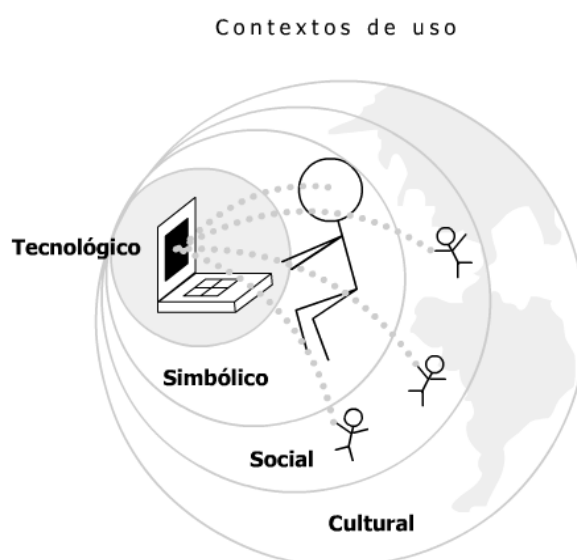


Figura 4.2 Modelo da definição de contexto de uso, retirado de (Sousa, 2008)

No que respeita a este modelo, tanto o contexto social como o contexto cultural não foram por nós utilizados, já o mesmo não aconteceu com o contexto simbólico e o contexto tecnológico. O contexto simbólico refere a linguagem que o interface usa para com o utilizador, ou seja, a linguagem de interação existente entre estes dois elementos. O contexto tecnológico de um produto está relacionado com as características da implementação desse produto, ou seja os atributos do sistema.

Perante tal, a análise sistemática do contexto desenvolvida por nós, visa desenvolver um sistema que prove a viabilidade do Modelo Interativo Adaptável.

Aliando o “contexto do ensino” ao nosso modelo “ Modelo Interativo Adaptável Orientado para o Ensino”, passamos a descrever a análise sistemático do contexto por nós efetuada. Esta análise visa efetuar uma abordagem clara e detalhada do sistema. Para tal, foi primeiro necessário realizar o levantamento da informação através da análise de requisitos, simultaneamente indicar o que o sistema deve fazer e de que dados necessita para satisfazer esses requisitos. Ou seja, definimos um sistema num ambiente ideal completamente

independente de restrições tecnológicas. Simultaneamente apresentámos a estrutura dos processos, dos dados e a abordagem dos controlos integrando os processos e os dados. Desta forma, começamos por identificar os eventos externos, as funções do sistema que respondem a esses eventos, ao eventos gerados internamente e as próprias funções, para que o modelo a implementar represente o ambiente real, agora já completamente dependente de restrições tecnológicas.

4.1 Análise de Requisitos

“ (...) um milagre tinha acontecido – o milagre de transformar aquilo que se faz naquilo que se crê.”, Paulo Coelho (Coelho, 1990)

Nesta primeira fase, ao querer testar a interação homem-máquina, houve que identificar as informações necessárias para essa interação, na recolha e análise de dados, as quais são identificadas como requisitos. Para tal, houve que seguir um conjunto de conceitos, práticas e técnicas de forma a garantir que os requisitos do sistema são consistentes, credíveis e completos. Vejamos alguns desses conceitos.

A IEEE (Association, 1990), define requisitos do sistema como:

- 1) Uma condição ou capacidade necessária a um utilizador, de forma a resolver um problema ou alcançar um objetivo;
- 2) Uma condição que deve ser cumprida de forma a satisfazer o utilizador (independentemente do tipo de utilizador);
- 3) Documento que ilustre as condições definidas nos pontos 1 e 2.

De acordo com Sommerville e Sawyer (Sommerville & Sawyer, 1997), requisitos do sistema são descrições de como o sistema se deve comportar, ou atributos de um sistema, podendo também ser restrições no processo de desenvolvimento de um sistema que podem incluir informações sobre o domínio da aplicação.

Segundo Dahlstedt (Dahlstedt, 2003) os requisitos tem três elementos que o caracterizam: a origem, o motivo e o objeto da sua realização, como se apresenta na **Figura 4.3**.

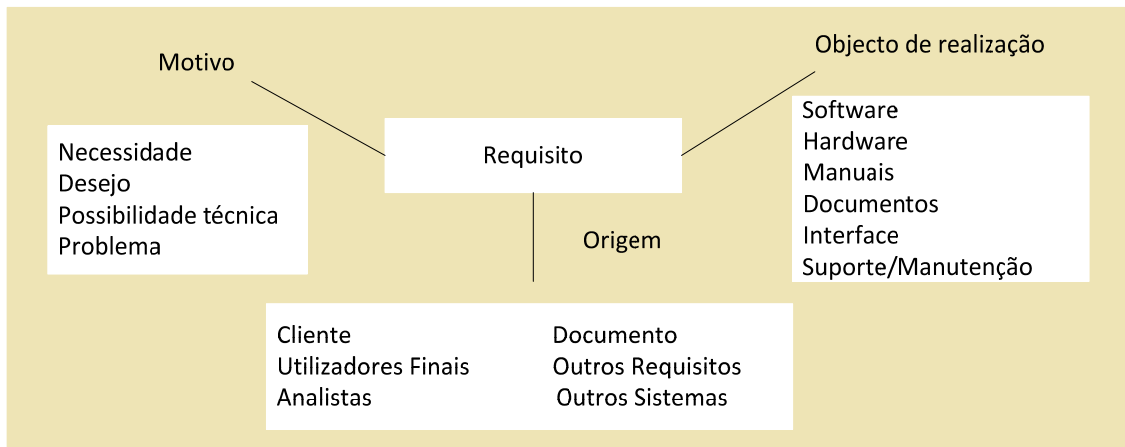


Figura 4.3 Elementos de um requisito, adaptado de (Dahlstedt, 2003)

Segundo o mesmo autor a origem de um requisito deve-se ao interesse que uma pessoa demonstra perante o desenvolvimento de um sistema, ou por requisitos especificados anteriormente ou gerados de interações com outros sistemas, onde os utilizadores finais e os analistas tem um papel predominante. O motivo pelo qual um requisito existe é pela necessidade de satisfazer os interessados num sistema e pela possibilidade técnica de resolver problemas existentes num sistema ou na relação com outros sistemas. Para o mesmo autor, o objeto de realização de um requisito pode ser um ou mais objetos de realização, normalmente Software, porque pode implicar uma alteração nos restantes objetos de realização.

É de realçar que todas as descrições de requisitos de sistema levaram a que os requisitos se dividissem em requisitos funcionais e requisitos não funcionais, de acordo com a figura 4.4.

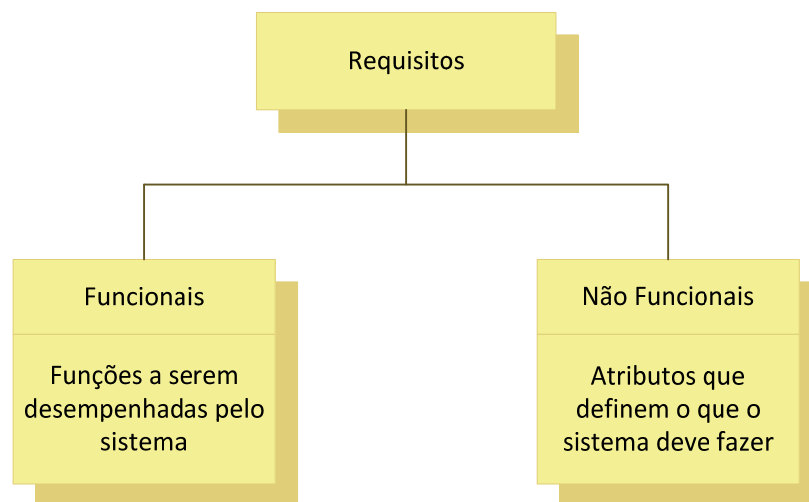


Figura 4.4 - Representação dos Requisitos

A figura 4.4 apresenta de uma forma simples a separação dos requisitos em requisitos funcionais e não funcionais e os objetivos de cada um deles.

Os requisitos funcionais descrevem as funções a serem desempenhadas pelo sistema, podendo ser requisitos do utilizador, do negócio e do sistema. Enquanto, os requisitos não funcionais referem as restrições na implementação dos requisitos funcionais, os quais podem ser atributos de qualidade, interfaces externas e outras. Estes requisitos estão relacionados com a política das organizações, com o desempenho do sistema, etc..

Afigura 4.5 seguinte apresenta com maior detalhe os requisitos não funcionais

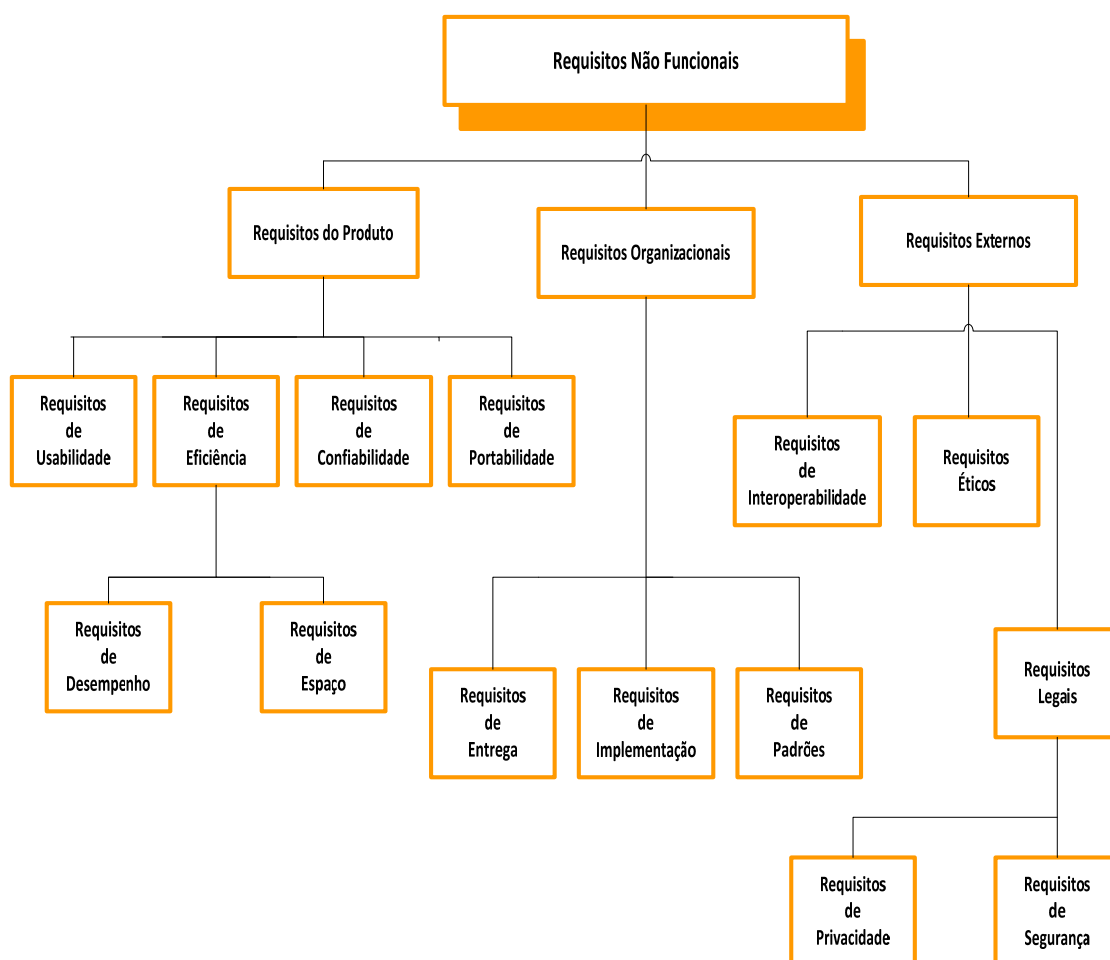


Figura 4.5 Características dos Requisitos Não Funcionais, (Sommerville & Sawyer, 1997) adaptado

É de salientar que a recolha dos requisitos passa por várias etapas, como se apresenta na figura 4.6. De forma esclarecedora, é exposto todo o processo de definição de requisitos. Estas etapas foram realizadas por nós na zona do ambiente das tarefas da conjugação do processo UID e do modelo HCI (ver figura 3.1).

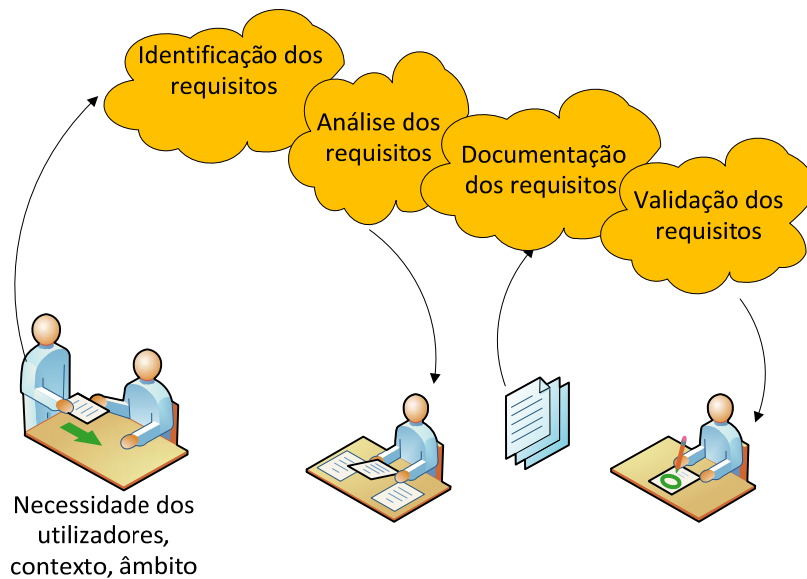


Figura 4.6 Etapas do processo de criação dos requisitos

A identificação dos requisitos tem como objetivo identificar quais são as características do sistema a desenvolver. A análise dos requisitos permite-nos entender como vai funcionar o sistema, como consequência, permite-nos separar os requisitos funcionais e não funcionais.

A documentação dos requisitos descreve os requisitos em termos técnicos de forma objetiva, para que equipa de desenvolvimento os possa concretizar. Até este ponto englobamos os processos UID-Requisitos do Sistema, UID - Análise ao Utilizador e UID – Definição das Tarefas do Utilizador (ver figura 3.1). Para Sommerville (Sommerville I. , 2001), a validação dos requisitos deve avaliar os requisitos quanto à sua consistência, relevância, clareza, organização, ambiguidades, complementaridade, entre outros. Esta etapa pode ser realizada por diferentes técnicas, tais como: criação de protótipos, criação de testes, revisão dos requisitos, segundo Pressman (Pressman R. S., 2005) o que está de acordo com a conjugação do processo UID e do Modelo HCI (figura 3.1).

Após a descrição sintética das etapas do processo de criação dos requisitos, passamos a referir que os requisitos funcionais e não funcionais do *software* “Palavras e Palavras” estão criados de acordo com os pontos 4.1.1 e 4.1.2 com o modelo em cascata (ver figura 2.21) no ambiente das tarefas na zona “UID- Requisitos do Sistema” (de acordo com a figura 3.1) que por sua vez se relaciona com o “modelo híbrido” (ver figura 2.25). Perante tal, passamos a apresentar os requisitos funcionais e não funcionais do *software* “Palavras e Palavras”.

4.1.1 Requisitos funcionais do Software "Palavras e Palavras"

Como já foi referido o enfoque deste estudo está no desenvolvimento de um novo método de desenvolvimento de *software* interativo, adaptável e orientado para o ensino, para tal tivemos de desenvolver um *software* que pudesse sustentar esse novo modelo. Logo, tivemos de analisar o sistema sob dois pontos de vista distintos, o utilizador aluno e o utilizador professor/educador, o que nos permitiu recolher dois tipos de informações distintas em relação aos requisitos funcionais. A tarefa de identificar estes requisitos faz parte da atividade 1 do ciclo de vida do *software* (referido no capítulo 2). Assim, a tabela 4.1 apresenta os requisitos funcionais do utilizador aluno.

Requisitos Funcionais – Utilizador Aluno	
O sistema deve permitir as seguintes funcionalidades	Inserir / Eliminar alunos
	Selecionar alunos
	Selecionar qualquer um dos jogos, selecionar atividade
	Selecionar nível na atividade
	Visualizar a sua pontuação parcial (respostas certas e erradas) em cada atividade
	Guardar as pontuações de cada aluno, respeitante a cada jogo
	Apresentar sob a forma de gráfico a evolução do aluno, nos diferentes jogos
	Imprimir fichas de trabalho, de acordo com os temas

Tabela 4.1 Tabela de Requisitos Funcionais - Utilizador Aluno

O resultado da análise efetuada para o para utilizador professor/educador, apontou para um conjunto de funcionalidades que o sistema deve possuir, face às necessidades deste utilizador. A tabela 4.2 apresenta os requisitos funcionais do utilizador professor/educador.

Requisitos Funcionais – Utilizador Professor/educador	
O sistema deve permitir as seguintes funcionalidades	Carregar dados para Descobre Palavras
	Carregar dados para Constrói frases
	Carregar dados para Constrói Palavras
	Inserir Tema
	Inserir Fichas de trabalho
	Visualizar gráficos de pontuação parcial (respostas certas e erradas) por aluno e por jogo

Tabela 4.2 Tabela de Requisitos Funcionais - Utilizador Professor/Educador

4.1.2 Requisitos Não Funcionais do Software “Palavras e Palavras”

Diferenciar um requisito funcional de um requisito não funcional é uma tarefa difícil e nem sempre é transparente. Esta situação deve-se ao facto de um requisito não funcional estar sempre ligado a um requisito funcional. Esta tarefa faz também parte da atividade 1 do ciclo de vida do *software* (referido no capítulo 2).

Para o nosso estudo nem todos os requisitos não funcionais foram implementados, visto que nem todos eram pertinentes para o “contexto ensino”. Na análise aos requisitos não funcionais, abrangemos:

- Os requisitos do produto os quais englobam:
 - Os requisitos de usabilidade, requisitos de eficiência, requisitos de confiabilidade e os requisitos de portabilidade.
- Os requisitos externos, nos quais focamos:
 - Os requisitos de legais, nos quais focamos:
 - Os requisitos de segurança
- Os requisitos organizacionais, nos quais englobamos:
 - Os requisitos de implementação.

A tabela 4.3 apresenta em detalhe os requisitos não funcionais do nosso sistema.

Requisitos Não Funcionais			
Requisitos organizacionais		Implementação	O sistema precisará de ser executado nas plataformas: Windows 98/Windows XP
Requisitos Externos	Requisitos Legais	Segurança	A informação relativa às atividades só pode ser inserida pelo utilizador com perfil professor/educador. A identificação do utilizador com o perfil aluno é inserida por este.
Requisitos do Produto		Usabilidade	Utilização de cores, imagens perceptíveis de identificar, sons, ajuda <i>online</i> , foco no utilizador, consistência na interface do utilizador O sistema deve usar gráficos que representem as respostas dos alunos O sistema deve usar cores na construção dos gráficos
		Desenho	Interface apelativo para os utilizadores em questão
		Sustentabilidade	Possibilidade de testes, manutenção, possibilidade de instalação, possibilidade de internacionalização
		Interface	Restrição de formatos (ficheiros: JPG, doc, docx, xls,xlsx)

Tabela 4.3 Tabela dos Requisitos Não Funcionais do Software “Palavras e Palavras”

4.1.3 Diagramas de Fluxo de dados

Os Diagramas de Fluxos de Dados (DFD) permitem-nos mostrar os processos do sistema e o fluxo de dados inerentes (fluxo de dados através do sistema); em cada passo (processo) os dados são transformados para o processo seguinte; pode ter mais que um nível de abstração ou seja poderão ser evidenciados diferentes níveis de abstração desde o mais abstrato ao mais detalhado, é uma ferramenta simples e intuitiva, mesmo para não informáticos. A simbologia utilizada na criação dos diagramas de fluxo de dados, foi a apresentada por Edward Yourdon (Yourdon, Modern Structured Analysis, 1989), como se apresenta na tabela 4.4.

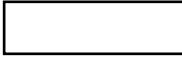
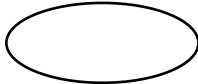

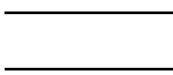
Símbolo	Designação	Descrição
	Entidade Externa (EE)	A entidade externa está fora do sistema, mas comunica com ele. Pode ser uma organização; uma pessoa; um grupo de pessoas ou algo, podendo ser outro sistema fora do sistema em estudo. Funciona como a origem ou o destino dos dados.
	Processo (P)	O processo é a representação da parte do sistema que transforma entradas em saídas.
	Fluxo de dados (FD)	O Fluxo de Dados demonstra o fragmento de informações de um ponto para outro do sistema.
	Depósito de Dados (DD)	Os depósitos de dados modelam depósitos de informação.

Tabela 4.4 Simbologia do DFD

Para a conceção de um DFD utiliza-se a terminologia “*Top-Down*”, visto que, cada um dos diferentes níveis de detalhe do sistema é mostrado através de diferentes níveis de DFD. Como tal, a primeira representação do sistema é realizada através do diagrama de contexto. Ou seja, o diagrama de contexto representa as entidades que interagem com o sistema, assim como o processo que dá origem a essa interação.

Após a criação do diagrama de contexto, este é decomposto e é elaborado o DFD nível 0 (zero), o qual representa os principais processos, fluxos de informação (entradas e saídas), os depósitos de dados e as entidades externas.

No DFD nível 0 (zero) todos os processos devem ter o mesmo nível de detalhe, por outras palavras devemos verificar se existe algum processo representado que não seja mais do que uma subactividade de um outro processo existente. Cada processo do DFD de nível 0 (zero) pode ser decomposto noutros processos, os quais apresentam em maior detalhe a lógica dos procedimentos.

Como tal, a decomposição do DFD nível 0 (zero), é designada por nível 1 (um), o que gera um novo DFD, o qual é denominado como DFD nível 1 (um). Por sua vez, o DFD nível 1 (um) ou ser decomposto origina outro nível de decomposição, nomeado como nível 2 (dois).

É de referir, que só existe um DFD nível 0 (zero), porém podem existir vários DFD’S de nível 1 (um), que correspondem a decomposições de processos do DFD nível 0 (zero). Obviamente,

esta observação vale para o DFD nível 2 (dois) e subsequentes, pois serão criados a partir de decomposições de processos de um nível anterior. Assim, a diferença de entre o diagrama de contexto, o primeiro DFD e DFD de níveis inferiores centra-se no maior ou menor detalhe que é feito do sistema. Como se pode verificar através da tabela 4.5.

DFD	Visão sob o sistema	Nível
Contexto	Muito geral Somente um processo	Superior
Primeiro	Global Processos Fundamentais Ligações Fundamentais	0 (zero)
Níveis Inferiores	Aprofundado Complexidade baixa – entre 2 a 3 níveis Complexidade média – entre 3 a 6 níveis Complexidade alta – entre 5 a 8 níveis	Inferior

Tabela 4.5 Diferenças entre os Níveis dos DFD'S

A figura 4.7 retrata teoricamente o desenvolvimento de um DFD, de acordo com o referido anteriormente.

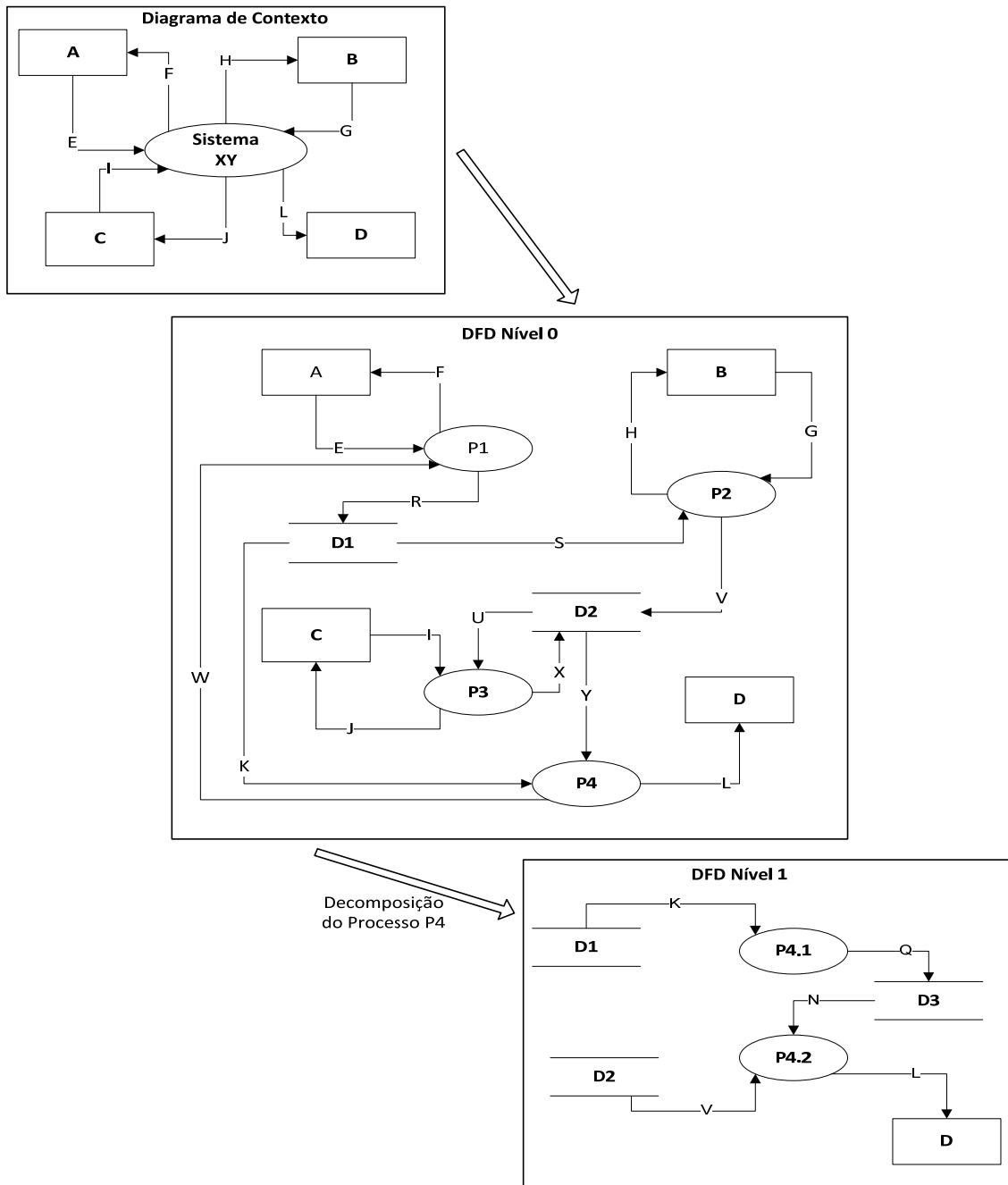


Figura 4.7 Representação da decomposição de Diagramas de Fluxo de Dados (DFD'S), adaptado de (Baldas, Campos, & Coutinho, 2006)

Após, breve descrição do conceito de Diagrama de Fluxos de Dados, apresentamos a modelagem do nosso estudo, através de DFD'S. Esta modelagem está em sintonia com o modelo híbrido (na fase de engenharia de software e análise de sistemas) e com a zona "UID – Análise de Sistema" da conjugação do Processo UID e do Modelo HCI. É de recordar que esta zona é composta pela análise dados e pela especificação de processos. A análise de dados está descrita neste ponto e no ponto 4.1.4, enquanto a especificação de processos está no ponto 4.1.5.

Passagem do Diagrama de Contexto do nosso estudo

A figura 4.8 apresenta o diagrama de contexto do nosso estudo, este revela as entidades externas “Aluno” e “Professor/Educador” que interagem com o sistema através do processo “Jogo Palavras e Palavras”.

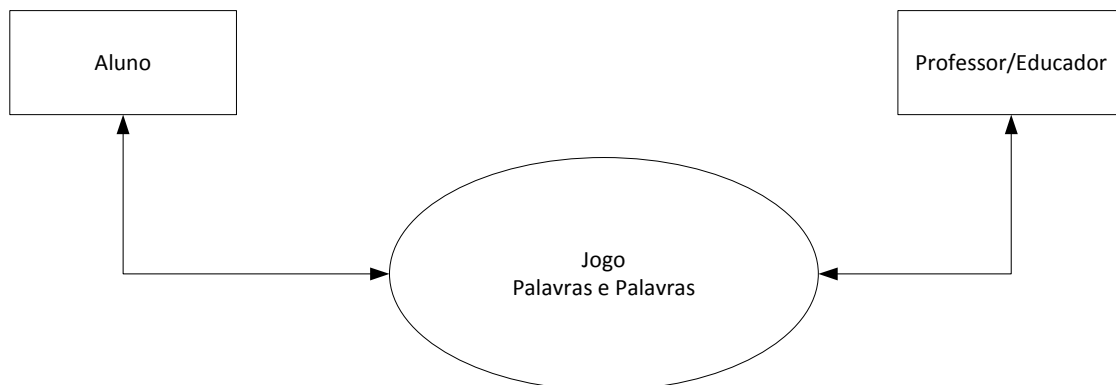


Figura 4.8 Diagrama de Contexto do nosso estudo, *Software "Palavras e Palavras"*

A passagem do diagrama de contexto para o diagrama de fluxo de dados DFD nível 0 (zero), apresenta oito processos, cinco depósitos de dados, diversos fluxos de dados e duas entidades externas, como se pode ver figura 4.9.

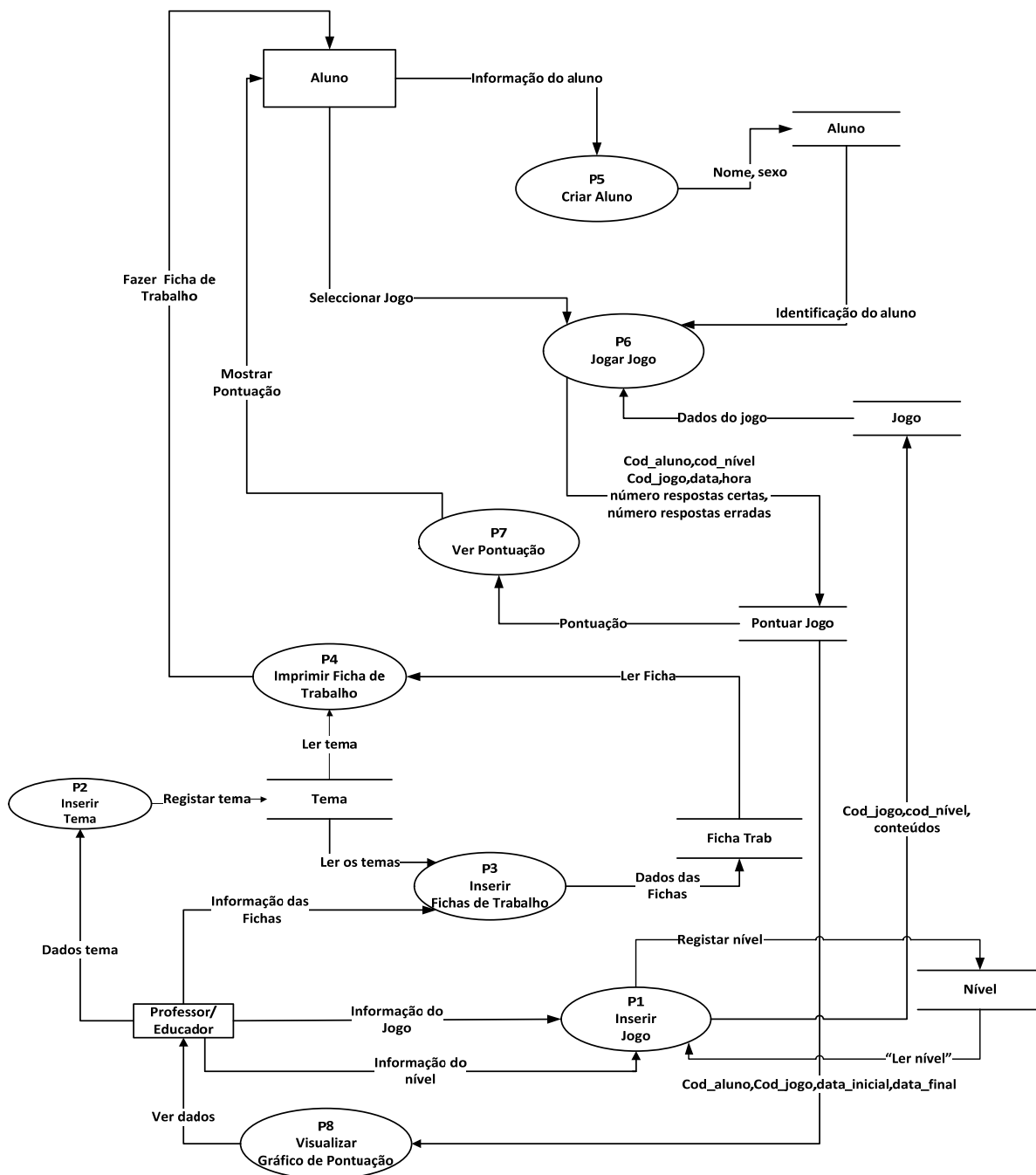


Figura 4.9 DFD Nível 0 do nosso estudo *Software "Palavras e Palavras"*

Após o desenvolvimento de DFD nível 0 (zero), houve necessidade de decompor alguns deles, de forma a descrever com maior precisão determinados procedimentos, o que permite analisar o sistema em detalhe.

Diagramas de Fluxo de Dados do Nível 1 do Software “Palavras e Palavras”

Pela sua complexidade, o primeiro processo que decomposemos foi o processo P1 denominado “Inserir Jogos”,. Este processo foi decomposto ao nível 1, em três subprocessos que correspondem aos jogos existentes no *software*, como se pode verificar na figura 4.10. Este processo diz respeito às atividades do professor e/ou do educador. O professor/educador tem de inserir todos os dados relativos aos próprios jogos, de acordo com as particularidades de cada aluno. Os dados a inserir e os níveis dos jogos estão diretamente relacionados.

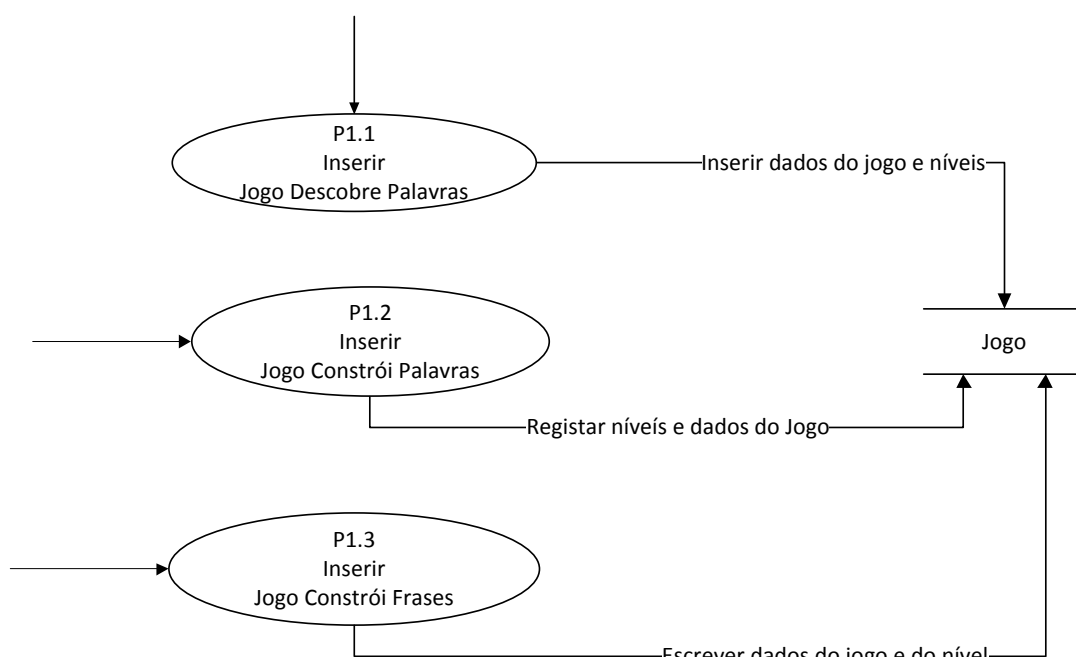


Figura 4.10 DFD Nível 1 do Processo P1 Inserir Jogo do Software “Palavras e Palavras”

Outro processo a ser decomposto é o processo P6 denominado “Jogar Jogo”, como se verifica na figura 4.11. Esta atividade está direcionada para os alunos. Após o aluno selecionar o seu nome deve escolher o jogo e por sua vez dentro deste deve escolher um dos níveis que o jogo apresenta. Este processo também envia a informação relativa às respostas dadas pelo aluno durante o jogo. Essa informação é registada no depósito de dados “Pontuar Jogo”.

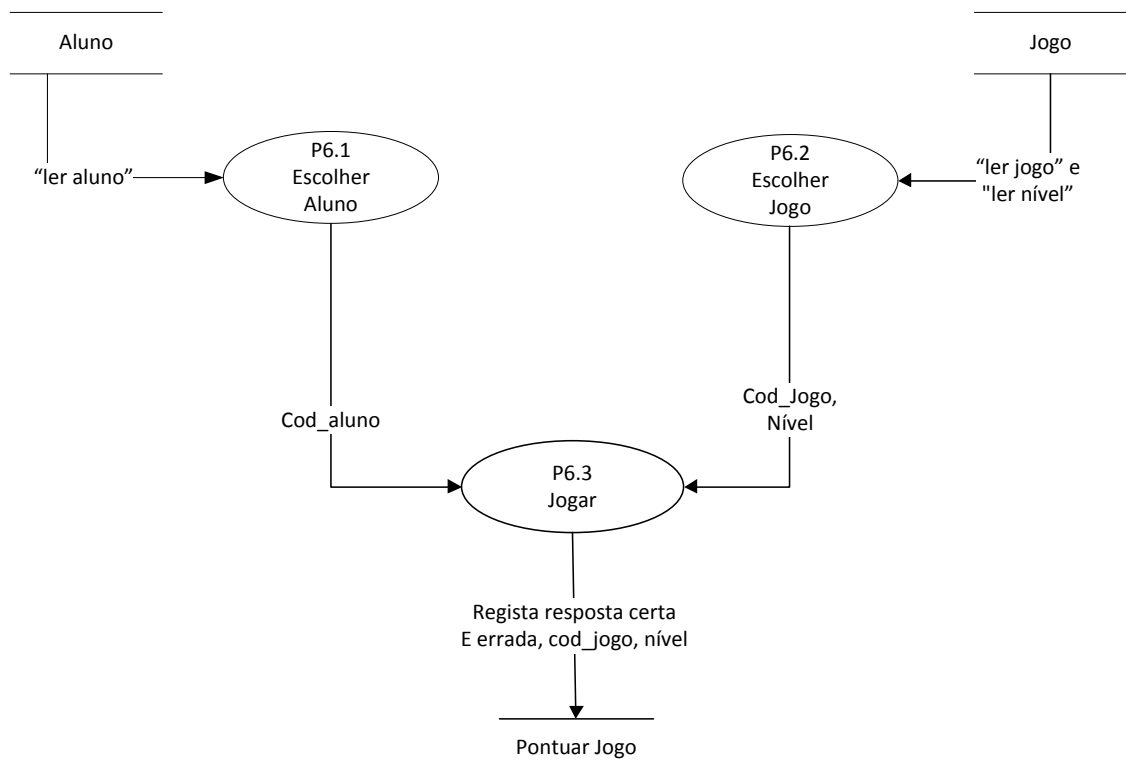


Figura 4.11 DFD Nível 1 do Processo P6 Jogar Jogo do *Software “Palavras e Palavras”*

4.1.4 Descrição dos processos, dos depósitos de dados e dos fluxos de dados do Software “Palavras e Palavras”

Para que haja um melhor entendimento dos diagramas de fluxos de dados, podemos considerar o dicionário de dados (DD) como o segundo elemento de análise de fluxos de dados, pois eles complementam os DFD’S na exposição da realidade a representar, aditando mais informação.

Um dicionário de dados é uma lista organizada dos elementos de dados apresentados nos DFD’S que pode ser apresentada de várias formas (por ex. através de uma tabela) onde é incluída uma descrição detalhada dos fluxos de dados (entradas e saídas), dos depósitos de dados, dos processos, das entidades externas e de alguns cálculos que possam existir.

Na análise estruturada os DD referente aos depósitos de dados têm uma notação que descreve de forma precisa e concisa cada componente de dados através de símbolos simples. No nosso caso, essa notação não foi utilizada visto que não foi usado o conceito de tabela, mas de ficheiro.

O modelo de descrição dos DD'S por nós optado é uma adaptação da modelação de Chris Gane e Trish Sarson (Gane & Sarson, 1987). As tabelas seguintes descrevem as entidades externas (EE), os fluxos de dados (FD) e os depósitos de dados (DD).

A tabela 4.6 descreve o significado dos processos representados no DFD's dos diferentes níveis.

Processos	Descrição
P1 Inserir Jogo	Permite ao professor inserir os dados dos três jogos de acordo com os níveis
P1.1 Inserir Jogo Descobre Palavras	Permite inserir 6 pares de imagens e o seu significado, de acordo com os níveis I,II e III
P1.2 Inserir Jogo Constrói Palavras	Permite inserir 4 imagens e identificá-las com diferentes cores, de acordo com os níveis
P1.3 Inserir Jogo Constrói Frases	Permite inserir 4 frases de cores diferentes, de acordo com os níveis I e II. Para o nível I – 2 frases de cor diferente, nível 2 – 2 frases de igual cor
P2 Inserir Tema	Permite ao professor inserir diversos temas, respeitantes à língua portuguesa
P3 Inserir Fichas de Trabalho	Permite ao professor inserir diversas fichas de trabalho de acordo com os temas posteriormente inseridos
P4 Imprimir Ficha de Trabalho	Permite ao aluno imprimir fichas de trabalho, para posterior resolução, através do processo de escrita manual
P5 Criar Aluno	Este processo permite criar o aluno, caso ele não exista
P6 Jogar Jogo	Este processo permite que um determinado aluno escolha o jogo e dentro deste o nível do jogo
P6.1 Escolher Aluno	Este processo recebe o nome do aluno que vai jogar
P6.2 Escolher Jogo	Este processo recebe a informação do jogo e do nível que o aluno selecionou
P6.3 Jogar	Permite registar todas as respostas certas e erradas que o aluno efetuou num determinado jogo de um determinado nível
P7 Ver Pontuação	Permite ao aluno ver as respostas certas e erradas que efetuou
P8 Visualizar Gráfico de Pontuação	Permite ao professor visualizar a evolução do aluno através de um gráfico, fase às respostas certas e erradas durante os jogos e num período de tempo

Tabela 4.6 Descrição dos Processos dos níveis 0 e 1 do Software "Palavras e Palavras"

A tabela 4.7 representa as entradas e saídas de informação dos processos e dos depósitos de dados dos níveis I e II.

Fluxo de Dados (Entradas)	Origem (de)	Destino (para)	Fluxos de Dados (Saídas)	Origem (de)	Destino (para)
Informação do Jogo	EE Educador	P1 Inserir Jogo	Dados do Jogo	P1 Inserir Jogo	DD Jogo
Dados do Tema	EE Educador	P2 Inserir Tema	Registrar tema	P2 Inserir Tema	DD Tema
Ler os temas	DD Tema	P3 Inserir Fichas de Trabalho	Dados das Fichas	P3 Inserir Fichas de Trabalho	DD Ficha Trab
Informação das Fichas	EE Educador	P3 Inserir Fichas de Trabalho	Dados das Fichas	P3 Inserir Fichas de Trabalho	DD Ficha Trab
Ler Ficha	DD Ficha Trab	P4 Imprimir Ficha de Trabalho	Fazer Ficha de Trabalho	P4 Imprimir Ficha de Trabalho	EE Aluno
Informação do aluno	EE - Aluno	P5 Criar Aluno	Dados pessoais	P5 Criar Aluno	DD Aluno
Identificação do aluno	DD Aluno	P6 Jogar Jogo	Cod_aluno	P6 Jogar Jogo	DD Jogo
Cod_Jogo	DD Jogo	P7 Ver Pontuação	Cod_jogo, Respostas certas e erradas	P7 Ver Pontuação	DD Pontuação
Pontuação	DD Pontuação	P7 Ver Pontuação	Mostrar Pontuação	P7 Ver Pontuação	EE Aluno
Cod_aluno, Cod_jogo, data_inicial, data_final	DD Pontuação	P8 Visualizar Gráfico de Pontuação	Ver Dados	P8 Visualizar Gráfico de Pontuação	EE Educador
“ler jogo”	DD Jogo	P6.1 Escolher Jogo	Jogo	P6.1 Escolher Jogo	P6.2 Escolher Nível
“ler nível”	DD Jogo	P6.2 Escolher Nível	Cod_jogo, Nível	P6.2 Escolher Nível	P6.3 Jogar
Regista resposta certa e errada, cod_jogo, nível	P6.3 Jogar	DD Pontuação			
Inserir dados do jogo e níveis	P1.1 Inserir Jogo Descobre Palavras	DD Jogo			
Registrar níveis e dados do jogo	P1.2 Inserir Jogo Constrói Palavras	DD Jogo			
Escrever dados do jogo e do nível	P1.3 Inserir Jogo Constrói Frases	DD Jogo			

Tabela 4.7 Entradas/Saídas de Fluxo de Dados do nível 0 do Software “Palavras e Palavras”

4.1.5 Especificação de Processos do Software “Palavras e Palavras” através de Fluxogramas

A palavra fluxograma advém do termo inglês “*Flow-Chart*”, donde:

Flow = Fluxo + *Chart* = Gráfico

Os fluxogramas representam de forma esquemática um processo e os fluxos entre os elementos que o compõem, através de símbolos convencionais. Como tal, um fluxograma revela, de forma clara e lógica, todos os fluxos dos processos por mais complexos que eles sejam, identificando assim alguma etapa problemática que possa existir.

A utilização de um fluxograma permite visualizar uma etapa específica assim como todas as outras que lhe estão interligadas, tal como a ordem em que devem ser executadas. Portanto o seu objetivo é conhecer profundamente todos os processos de um sistema.

Como tal, optamos por especificar detalhadamente os processos do Software “Palavras e Palavras” através de fluxogramas.

Os fluxogramas seguintes apresentam a especificação dos processos referentes à entidade aluno do nosso estudo.

Especificação dos Processos referentes aos Alunos através de Fluxogramas

O fluxograma geral representado pela figura 4.12 apresenta de uma forma geral os processos criados nos DFD nível 0, para a entidade Aluno.

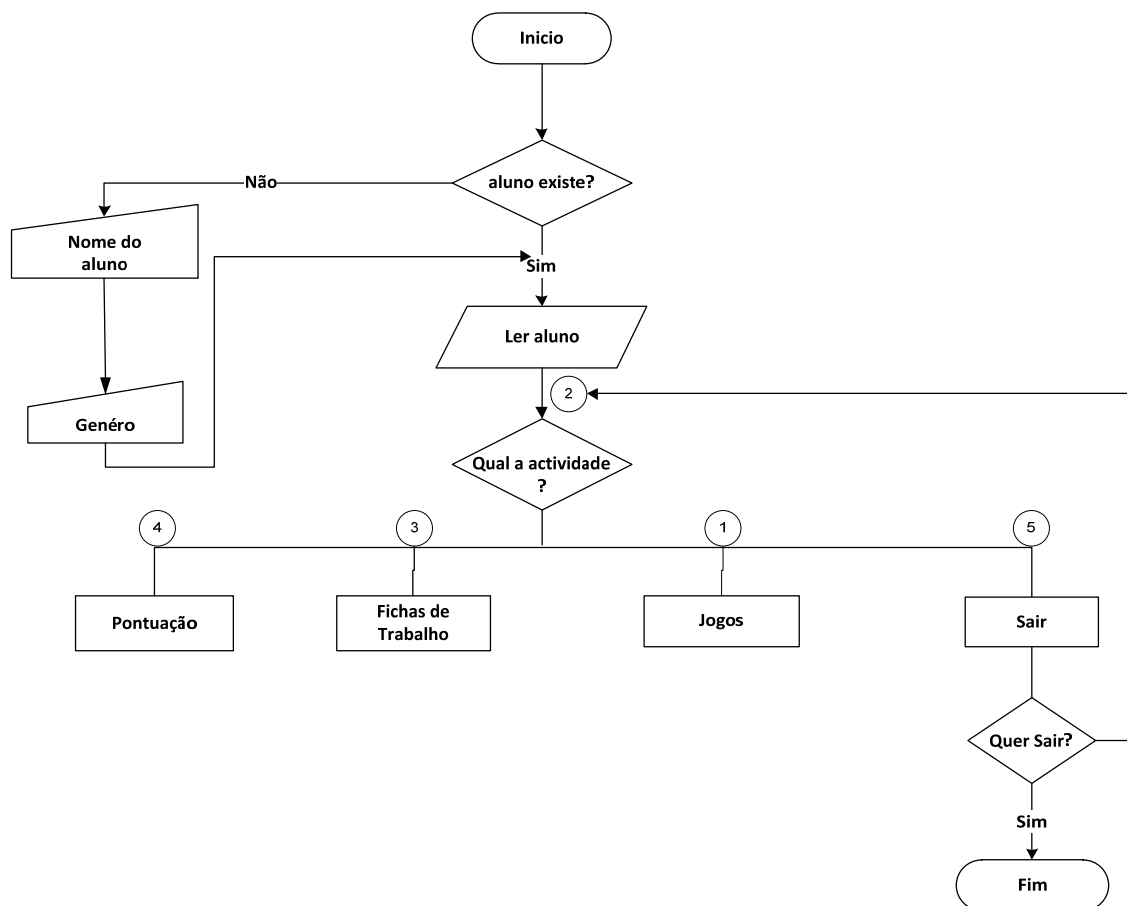


Figura 4.12 Fluxograma Geral do DFD nível 0 do *Software “Palavras e Palavras”*

É de referir que o fluxograma nível 0 (figura. 4.12) deve ser detalhado em maior escala, para que se possa entender os fluxos de informação que existem nas diferentes atividades. Assim, cada referência numerada existe nesta figura é tratada em detalhe nos fluxogramas seguintes.

Portanto, o Fluxograma Jogos (figura 4.13) representa o processo P6 “Jogar Jogos “, que está identificado no Fluxograma Geral do DFD nível 0 pela referência 1. Este fluxograma tem como objetivo permitir visualizar os passos que o programador tem de realizar para que o aluno possa jogar qualquer um dos jogos.

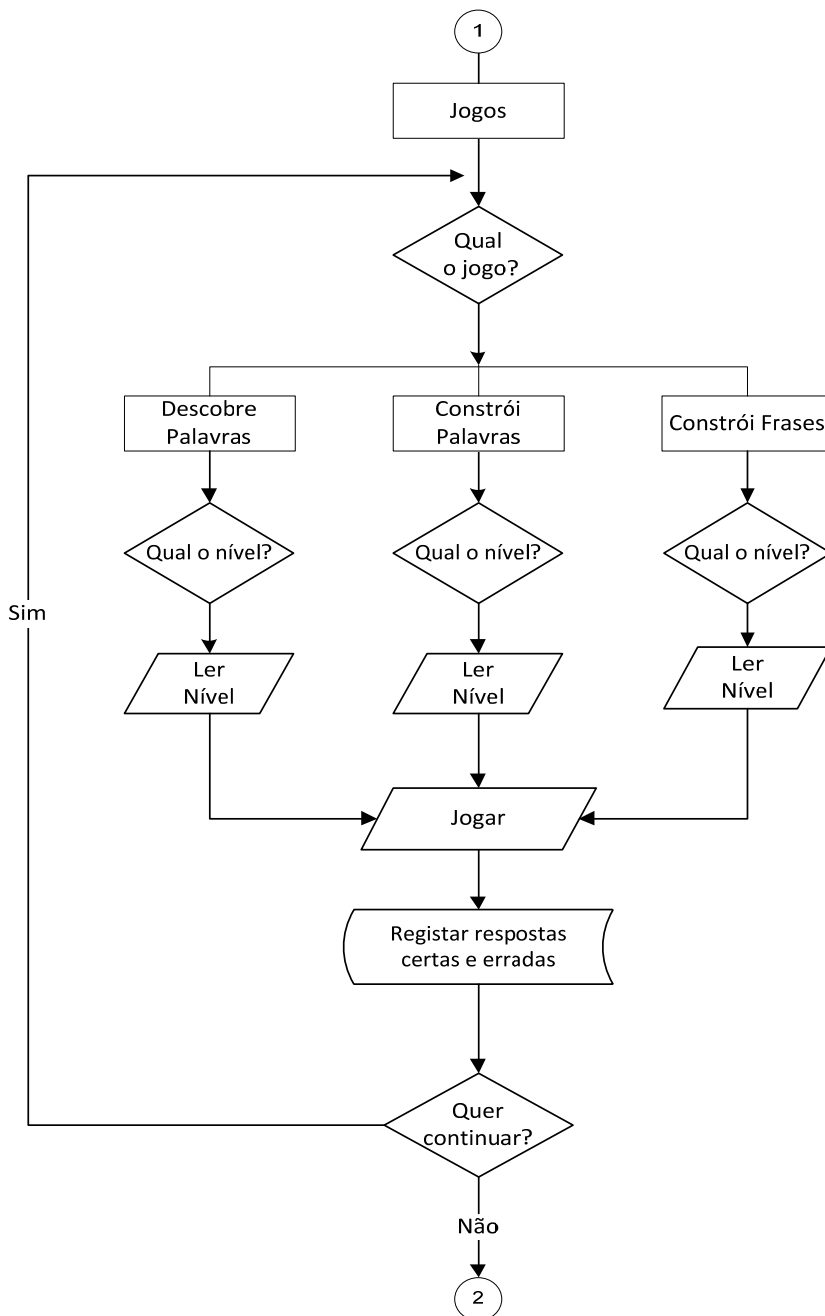


Figura 4.13 Fluxograma Jogos que representa o processo P6 do Software “Palavras e Palavras”

O fluxograma seguinte (figura 4.14) representa os passos da atividade fichas de trabalho, referentes ao processo P3 do DFD nível 0.

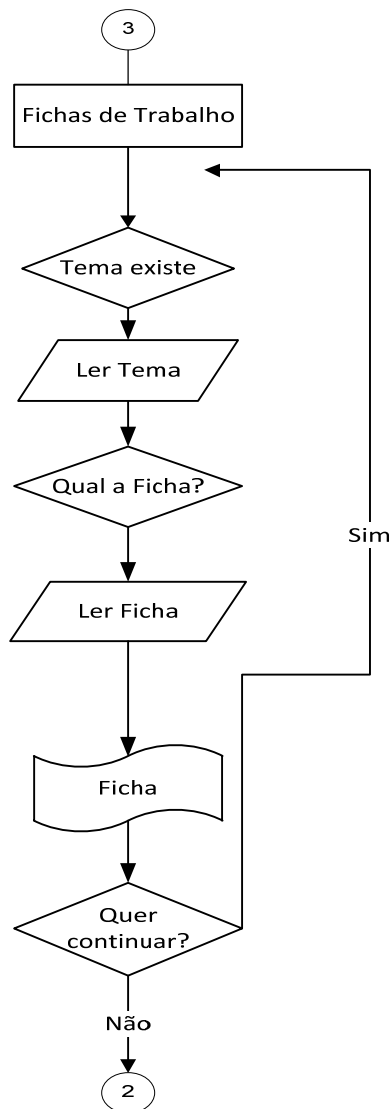


Figura 4.14 Fluxograma Fichas de Trabalho do *Software* “Palavras e Palavras”

Os Fluxogramas seguintes estão relacionados com DFD nível 0 e nível 1, para a entidade Professor/Educador do nosso estudo.

Especificação dos Processos referentes ao Professor/Educador através de Fluxogramas

Para que o aluno adquira determinados conceitos, através de *software*, temos de ter em conta, os meios que devemos utilizar, visto que, qualquer projeto de desenvolvimento de *software* multimédia envolve diferentes elementos, tais como o som, imagem, palavras escritas e todos os aspetos interativos próprios.

Se os conceitos a transmitir forem temporais o meio indicado é o auditivo. No entanto, se o conceito for espacial o meio é a visão. Porém, quando o conceito envolve noções temporais e espaciais, o meio indicado tem de envolver obrigatoriamente a visão e a audição, pois, só desta forma é possível a transmissão do conhecimento. Segundo (Collins, Hammond, & Wellington, 1997) a ligação entre estes elementos é muito importante, pois estabelece “*a ligação entre o que se vê e o que se ouve*”. Esta noção é essencial para a aprendizagem.

De acordo com Casas (Casas, 1987) “... *no hay aprendizaje sin percepción (visual, auditiva, táctil, olor, ...) que es la toma de conciencia de una sensación*”. Assim, a percepção depende tanto de aspetos sensoriais como de aspetos psicológicos, o que envolve o reconhecimento de padrões, pois estuda como uma imagem, uma palavra ou uma frase é entendida pelos indivíduos (neste caso os alunos). Também para Arnheim (Arnheim, 2002) “*ver significa captar algumas características proeminentes dos objectos..., onde umas simples linhas e pontos são de imediato reconhecidas como um rosto.*” Como tal, para o nosso estudo Casas e Arnheim revelam conceitos e noções muito importantes, como a percepção visual.

Assim, a percepção visual é orientada para a captação da informação, através dos olhos, por meio da luz e processada a nível cerebral. Para tal, esta percepção é orientada por padrões, o que é essencial no desenho de interfaces, segundo Boyle (Boyle, 1997) “*o desenho de ecrãs com padrões facilita a compreensão da informação*”, o que vai de encontro à Teoria da Forma, fundada pelos psicólogos da Gestalt. Esta teoria fundamenta que qualquer sistema é um todo constituído por partes correlativas que se podem decompor e ser analisadas de forma individual, no entanto qualquer alteração numa das partes, terá influência no conjunto (Eysenck & Keane, 2005).

Preece (Preece, Rogers, Sharp, Benyon, Holland, & Carey, 1994) foca a importância da percepção no estabelecimento da comunicação entre o utilizador (aluno) e o *software*. O desenho de interface tem pois de realçar as particularidades do raciocínio humano. Algumas destas particularidades constam na norma ISO 9241 (ver anexo J).

Afonso (Afonso, 2004) refere que “existem várias abordagens teóricas relacionadas com o reconhecimento de padrões”. Segundo o mesmo autor a maioria são as teorias de gabarito, teorias de protótipo, as teorias de atributo, a teoria computacional de Marr e a teoria da forma. A tabela 4.8 referencia os objetivos de cada uma destas teorias.

Teorias	Objetivos
Teorias de Gabarito	O estímulo é identificado em função do gabarito que produzirá a equiparação mais próxima ao seu <i>input</i>
Teorias de Protótipo	Cada estímulo pertence a uma classe de estímulos e que ele compartilha os atributos essenciais daquela classe
Teorias de Atributos	Cada estímulo visual é entendido como sendo composto por vários atributos, sendo o reconhecimento de padrões baseado na análise dos atributos do estímulo apresentado
Teoria Computacional de Marr	O estímulo visual é construído a partir de 3 etapas de processamento da imagem. O esboço primário – reconhece as características principais dos objetos. O esboço bi-e-meio-dimensional – reconhece a profundidade, a orientação e a distância entre os objetos. O esboço tridimensional – reconhece constâncias espaciais nos objetos e características ocultas
Teoria da Forma	Os elementos constituintes de um estímulo podem ser trocados ou alterados e o estímulo continua a ser reconhecido

Tabela 4.8 Teorias do Reconhecimento de Padrões

É de salientar, que a percepção por si só não é suficiente para a aquisição dos conhecimentos, esta tem de ser interpretada. Para tal, a interpretação baseia-se em induções reguladas pelas leis percetivas. Destacamos algumas destas leis na tabela 4.9.

Leis Perceptivas e sua descrição	
Lei da Simplicidade	Segundo (Arnheim, 2002), quanto mais simples for o padrão de uma imagem, maior será a tendência de captação dessa imagem. Portanto, a simplicidade implica sobriedade e ordenação.
Lei da homogeneidade ou semelhança de Cesare Musatti	Para (Arnheim, 2002), “A semelhança atua como um princípio estrutural apenas em conjugação com a separação, isto é, como uma força de atração entre coisas separadas”. Como tal, tem de existir diferentes agrupamentos de acordo com as suas semelhanças. Para o mesmo autor os agrupamentos podem ser por tamanho, proximidade, direção, cor e diferentes configurações
Lei da constância de forma e tamanho	Esta lei permite ver os objetos de acordo com a sua forma material objetiva.
Lei da coerência estrutural e pregnância	Esta lei só se manifesta quando se reconhece a estrutura do símbolo, mediante a confrontação com a estrutura do conceito visual, armazenada na memória (Villafañe 1998).

Tabela 4.9 Leis perceptivas

De acordo com as leis citadas para o ser humano, a seleção visual é uma das características da percepção, podendo manifestar-se a três níveis do processo perceptivo, segundo (Villafañe, 2006) através da sensação visual, memória visual e do pensamento visual. Cada uma destas fases perceptivas tem uma natureza, como se pode constatar através da tabela 4.10.

Fase perceptiva	Exemplo	Natureza
Sensação Visual	Obtenção da cor	Não cognitiva
Memória Visual	Qualquer conceito visual	Semi-cognitiva
Pensamento Visual	Reconhecimento de formas	Cognitiva

Tabela 4.10 Níveis do Processo Perceptivo da Seleção Visual

Ou seja, a forma como se apresenta a informação é determinante no processo ensino-aprendizagem do contexto ensino. A forma como o *software* apresenta os conteúdos através do texto, das imagens (animada ou em movimento ou imóvel) e do som, são vitais no processo

ensino-aprendizagem, de acordo com Cebrian (Cebrian, s/data) “(...)Según los estudios sobre el aprendizaje realizado con tecnología multimedia se estima que el tiempo empleado se reduce a un 50% debido a que la combinación de presentaciones visuales con explicaciones de audio facilita una mejor comprensión, y a que la interacción inmediata entre el alumno y la máquina permite un constante y efectivo refuerzo de estímulos durante el aprendizaje de conceptos y contenidos.”

Para tal, é necessário compreender melhor a estrutura das imagens visuais, donde precisarmos de raciocinar sobre os seus componentes. As imagens visuais contêm 3 tipos de informação: forma, cores e movimento. Porém, de que forma e em que quantidade o nosso cérebro processa essas imagens para as assimilar?

A teoria computacional de Marr é validada pelas evidências da neuropsicologia, onde a percepção visual toma o seu papel de referência através do córtex visual. A imagem 4.1 indica onde no nosso cérebro se situa o córtex visual.

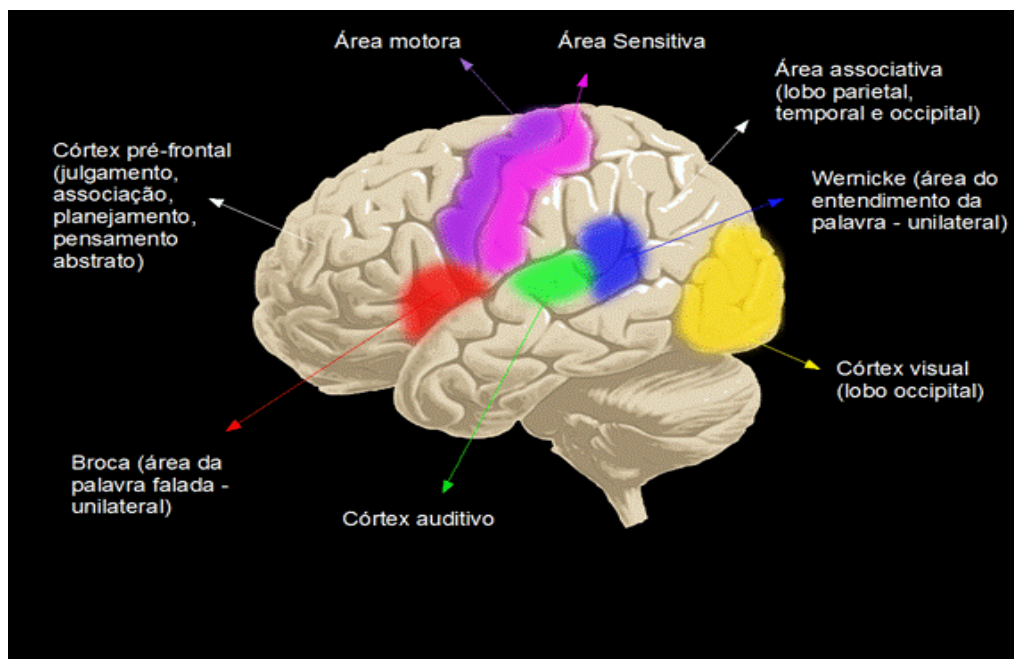


Imagem 4.1 Áreas do Cérebro Humano (Drews, 2007)

O processo de interpretação dos objetos ou imagens inicia-se no córtex visual situado no lobo occipital, nesta zona existem áreas específicas para processar a forma, cor e localização dos objetos ou das imagens(ver imagem 4.2).

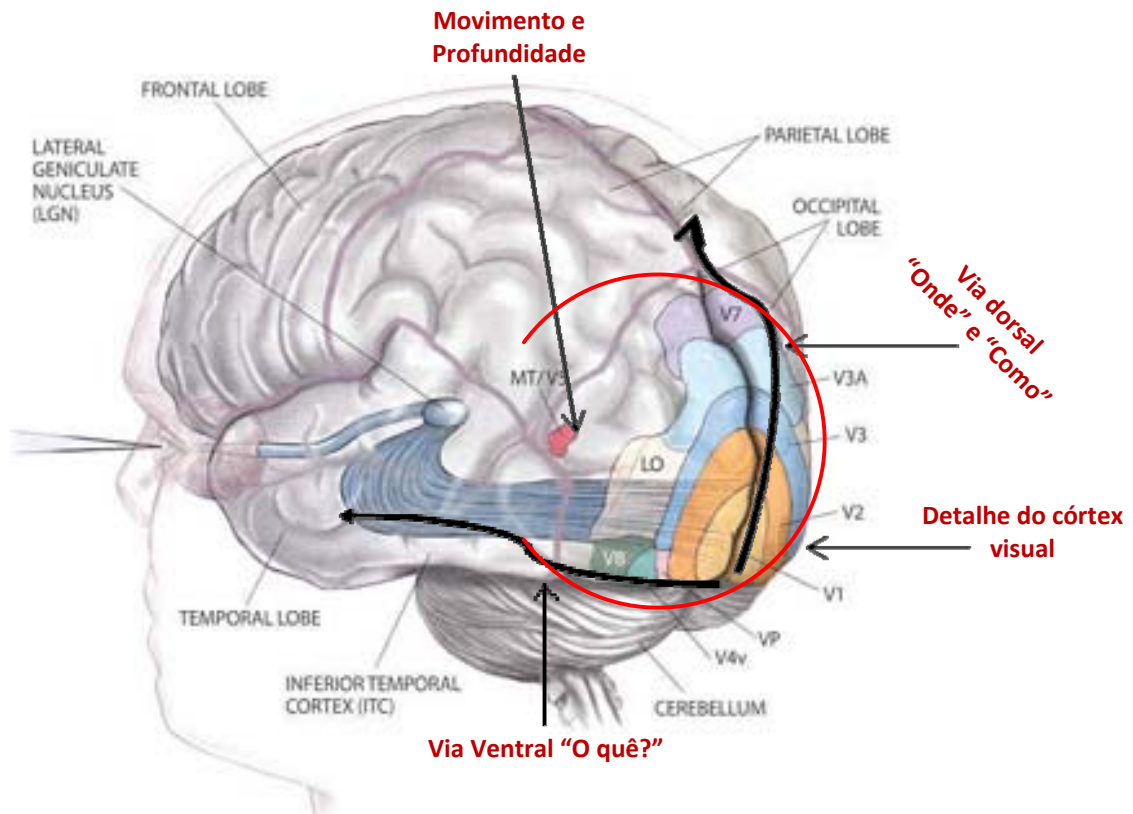


Imagem 4.2 Detalhe do Córtex Visual adaptado de (Hyvärinen, s/Data)

A interpretação dos objetos ocorre “desde os olhos até ao córtex visual primário numa via de dois sentidos na sua parte posterior, ... Existe um fluxo de informação desde o córtex visual primário (V1) para o corpo geniculado externo (LGN) e o calículo superior (CS).” (Hyvärinen, s/Data), ver imagem 4.3.

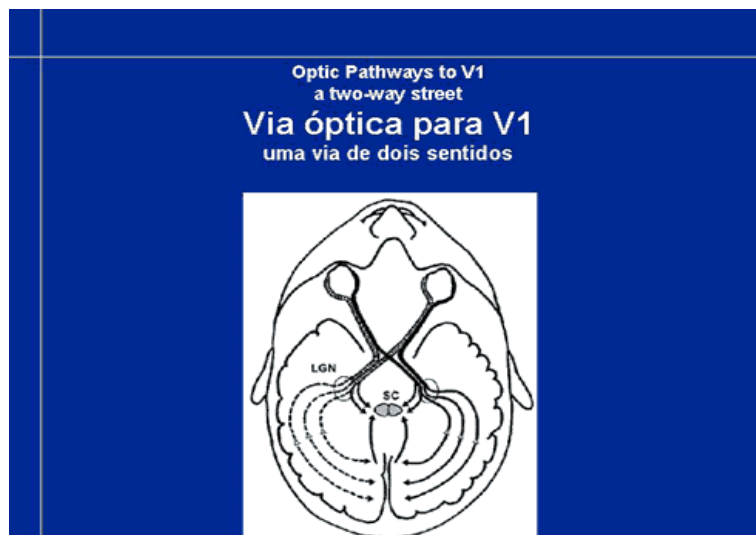


Imagem 4.3 Via óptica para V1, (Hyvärinen, s/Data)

A via dorsal e a via ventral estão representadas pelas setas pretas no hemisfério direito e esquerdo (ver imagem 4.2) do nosso cérebro.

A via dorsal “Onde e Como” (ver imagem 4.2) é representada pela seta com o sentido ascendente, que parte de v1 – córtex visual primário (anatomicamente equivalente à área 17 de Brodmann) em direção ao lobo parietal posterior (LPP), passando pelos extra-estreados v2 (faz a associação visual, a imagem é partida em 4 quadrantes, o que origina o mapa visual completo), v3, v5 (equivalentes a Brodmann 18,19) é responsável pelo reconhecimento da localização, movimento e profundidade do objeto, controlo dos olhos e membros superiores.

A via Ventral “o quê?” começa em v1 passa através de v2 vai para a frente e passa através de v4 até ao lobo temporal inferior (LTI), descodifica o objeto, é identificada a sua forma e a cor (V4) e está associada com o armazenamento da memória a longo prazo. Enquanto a memória a curto prazo está associada somente ao córtex parietal posterior, região do cérebro conhecida pelo velho adágio popular “*o que os olhos não vêem o coração não sente*”.

Segundo Marois e Todd (Marois & Todd, 2011) a capacidade de armazenamento da memória visual a curto prazo é de 4 objetos, “*A memória a curto prazo é um componente chave de muitas funções perceptivas e cognitivas e é suportada por uma larga rede neural, mas ela tem uma capacidade de armazenamento limitada*”.

No entanto, para Castro (Castro, 2005) “*o limite da capacidade de armazenamento de informação foi calculado à volta de sete unidades de informação*”, ou seja, conseguimos armazenar 7 objetos na memória a curto prazo.

Para Miller, (Miller, 1956) o homem só consegue processar entre 5 a 9 elementos por vez, o que implica que o conhecimento a transmitir tem de ser faseado para poder ser reconhecido como um todo.

O nosso *software* “Palavras e Palavras” engloba algumas atividades em que a memória a curto prazo é muito importante, porque é na memória, com duração de pouquíssimos segundos ou minutos, que a informação é mantida por processos de atenção e ensaio. É esta a memória que os alunos utilizam em 2 dos jogos (dependendo dos níveis) do *software* “Palavras e

Palavras”. Um exemplo desta memória é a aptidão de recordar acontecimentos vivenciados nos últimos minutos.

Tendo em conta que o público-alvo do nosso estudo são as crianças com uma faixa etária compreendida entre os 6 e os 10 anos e que usam ambos os olhos em conjunto binocularmente, quantos objetos ou imagens em simultâneo conseguem descodificar e memorizar?

Para termos resposta às nossas próprias questões temos de compreender a qualidade da imagem usada; compreender como a criança usa a imagem nas funções visuais cognitivas; como é que as alterações na qualidade da imagem contribuem para a aprendizagem da criança; compreender como funciona o campo visual em simultâneo com a memória. Estes pontos são essenciais para planear as estratégias de aprendizagem no ensino através de *software*.

Como existem opiniões controversas em relação à quantidade de objetos memorizados ou descodificados pelo ser humano, criámos 2 jogos (ou atividades) com diferentes números de imagens. O jogo “Descobre Palavras” para o aluno apresenta 12 imagens (ver figura 4.16 e a imagem 4.8), enquanto o jogo “Constrói Palavras” apresenta 4 imagens (ver figura 4.17 e a imagem 4.9). Em ambos os jogos estão associadas palavras às imagens, pelo que tivemos de ter em conta a forma de apresentação do texto e a cor do mesmo, associado a cada imagem.

De seguida, apresentamos os fluxogramas respeitantes às atividades dos professores/educadores. Estes fluxogramas abrangem o papel do córtex visual e especificam os processos P1, P2, P3 e P8 do DFD nível 0 do *software* “Palavras e Palavras” de acordo com a figura 4.12.

Fluxograma Geral Professor/Educador do jogo “Palavras e Palavras” representa as atividades do professor/educador. Este fluxograma (ver figura 4.15) exhibe indentificadores numerados, que indicam que atividade é descrita em maior detalhe num outro fluxograma.

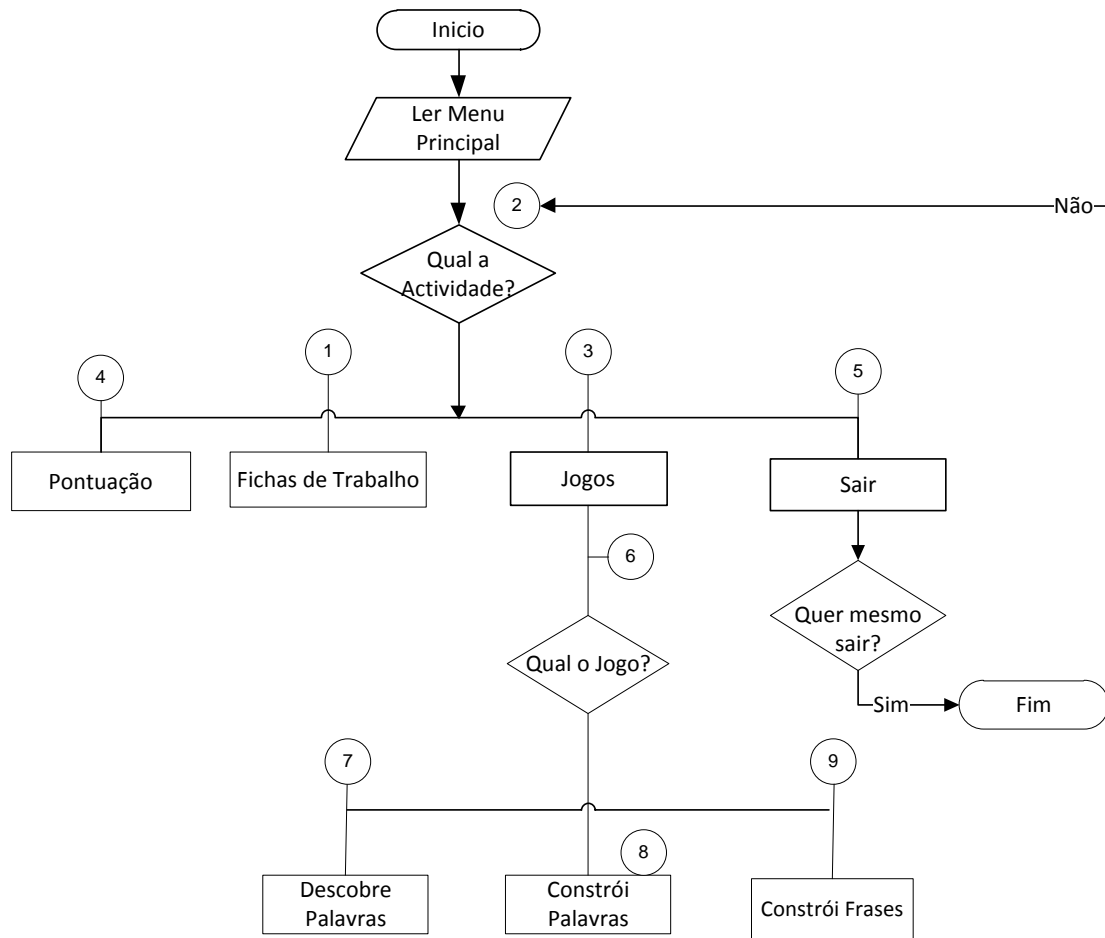


Figura 4.15 Fluxograma Geral Professor/Educador do Software “Palavras e Palavras”

No jogo “Descobre Palavras” existem 3 níveis. No nível 1, o professor deve inserir seis pares de imagens e a sua identificação. Os pares iguais têm de ter o nome escrito com a mesma cor, mas esta varia de par para par. No nível 2, as imagens para formar pares são iguais às do nível 1, mas o professor tem de inserir o nome, de cor igual para todas as imagens. No nível 3, o professor introduz somente as palavras a pares, respeitantes às imagens introduzidas anteriormente. No fluxograma “Descobre Palavras”, figura 4.16, poderá ser seguida esta situação.

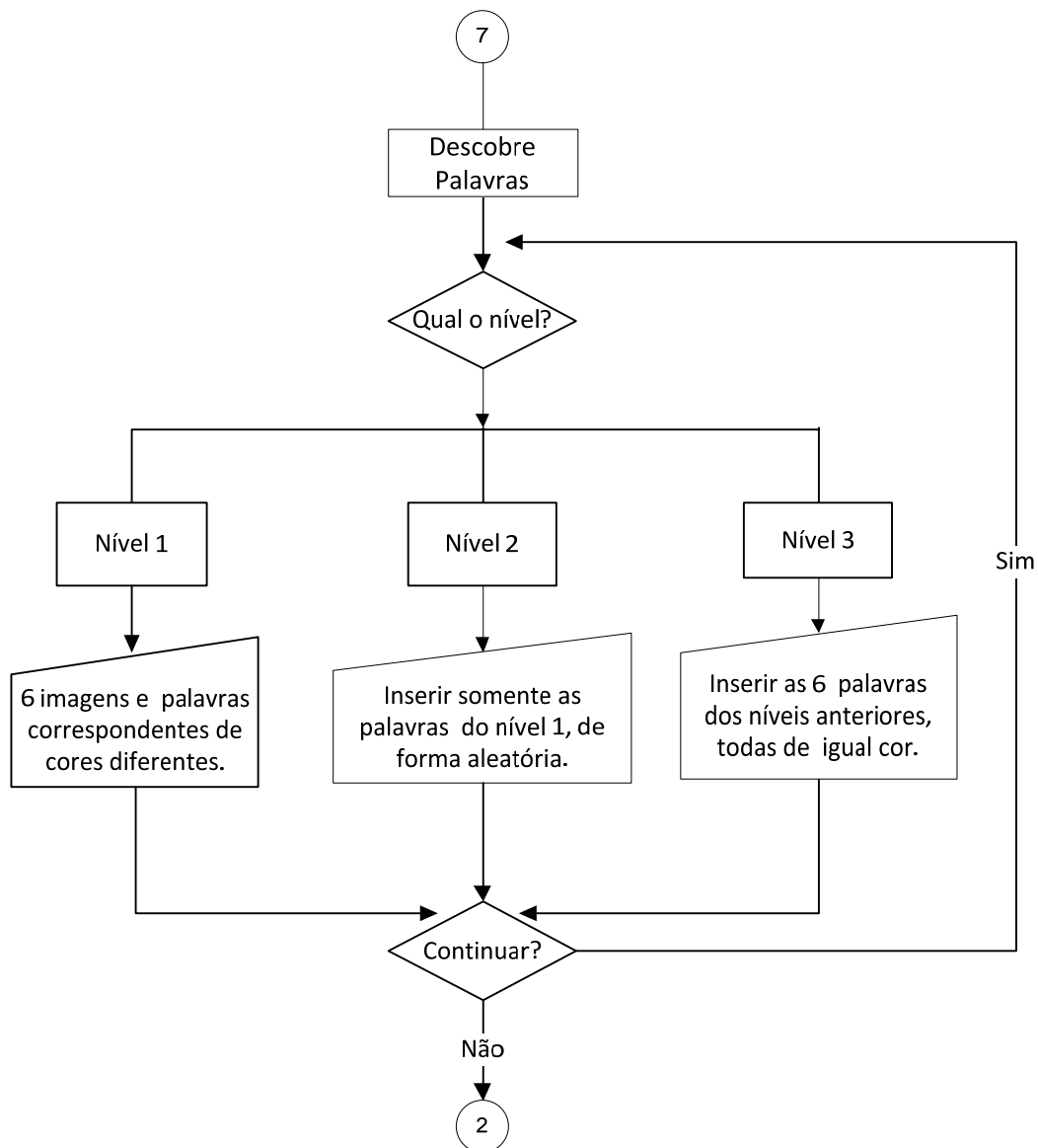


Figura 4.16 Fluxograma respeitante à atividade do professor no jogo “Descobre Palavras” do Software “Palavras e Palavras”

No jogo “Constrói Palavras” existem igualmente 3 níveis. No nível 1, o professor/educador deve introduzir quatro imagens, cada uma identificada com uma cor diferente. No nível dois, o professor/educador introduz somente as palavras relacionadas com as imagens introduzidas no nível 1, colocando as palavras com a cor usada na imagem do nível anterior. No nível 3, as palavras relacionadas com os objetos introduzidos no nível 1 têm igual cor, conforme se apresenta na figura 4.17.

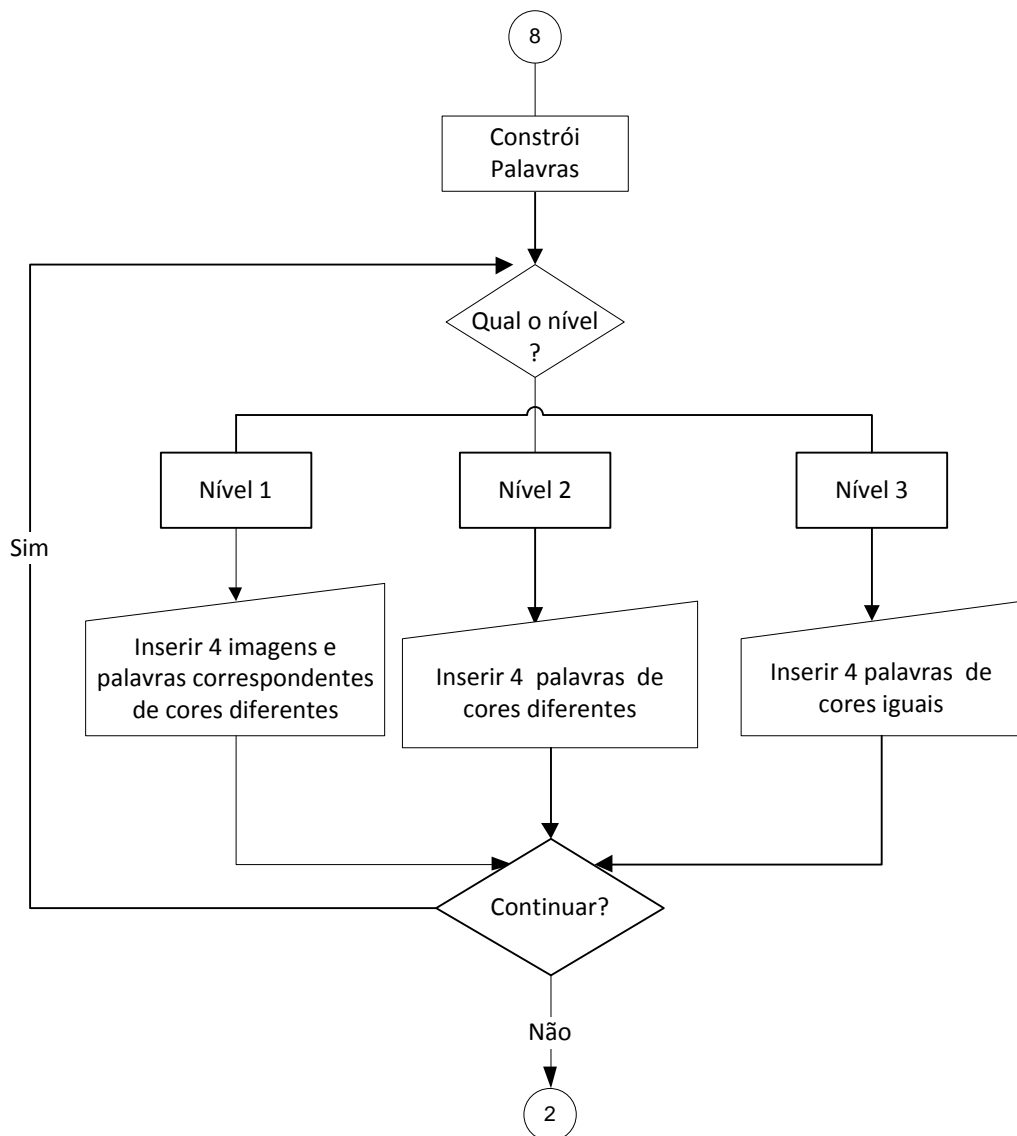


Figura 4.17 Fluxograma respeitante à atividade do professor no jogo “Constrói Palavras”

No jogo “Constrói Frases” existem 2 níveis. No nível 1, o professor/educador deve introduzir 2 frases de diferentes cores. Os dados a introduzir referentes às palavras constituintes de cada uma das frases são de igual cor, ou seja as palavras, que constituem a frase 1 são vermelhas enquanto as palavras da frase 2 são azuis. No nível 2, o professor/educador introduzir 2 frases de igual cor, neste caso, independentemente da frase as palavras introduzidas são da mesma cor conforme se apresenta na figura 4.18.

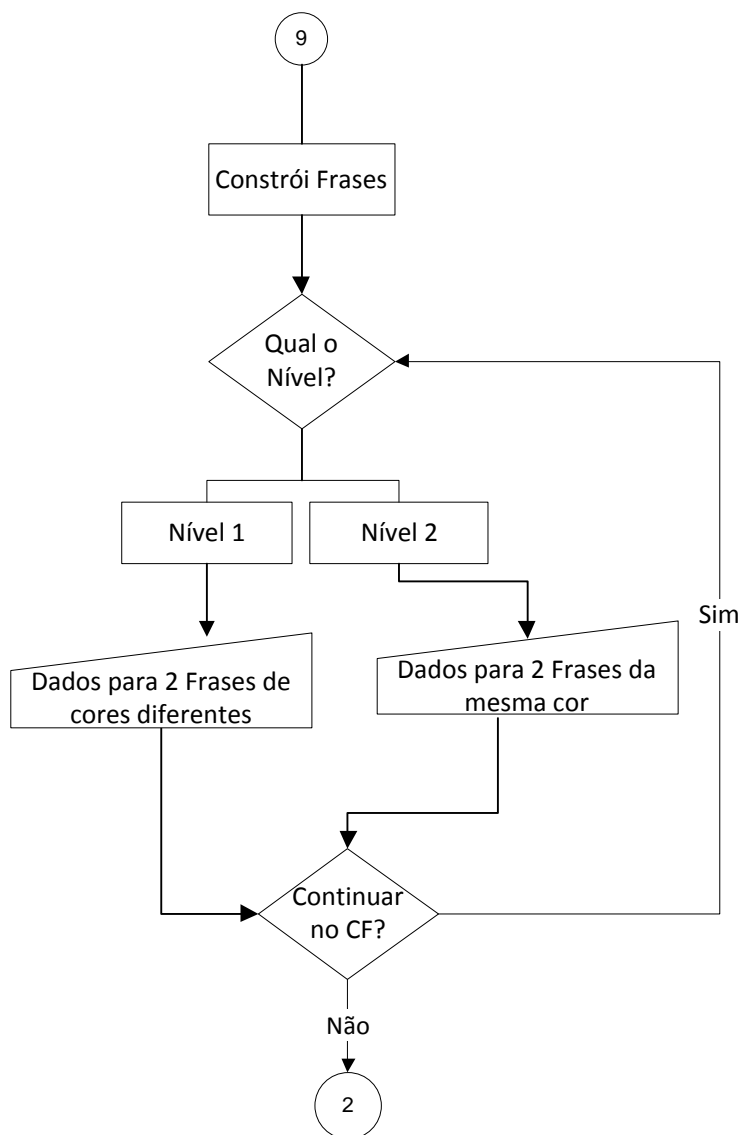


Figura 4.18 Fluxograma respeitante à atividade do professor no jogo “Constrói Frases”

O professor, em qualquer momento, pode verificar a evolução do aluno através da visualização das respostas certas e erradas que um determinado aluno deu, no decorrer de um determinado jogo, podendo ou não seleccionar um intervalo de tempo. Como se pode verificar através do fluxograma representado pela figura 4.19 e da imagem 4.17.

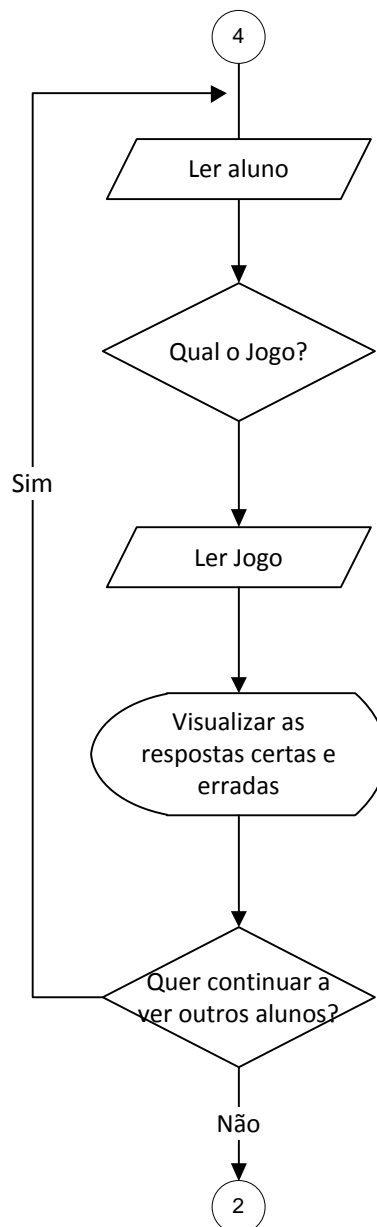


Figura 4.19 Fluxograma respeitante ao professor na atividade “Pontuação” no Software “Palavras e Palavras”

Outra atividade permitida ao professor é a inserção de fichas de trabalho. Esta atividade permite inserir fichas de trabalho para o aluno de acordo com os temas lecionados na sala de aula. Para tal, o professor, em primeiro lugar, tem de verificar se o tema já existe, caso não exista deve inseri-lo primeiro e só depois a ficha de trabalho, porque as fichas de trabalho são inseridas de acordo com os temas como se pode verificar no fluxograma representado pela figura 4.20.

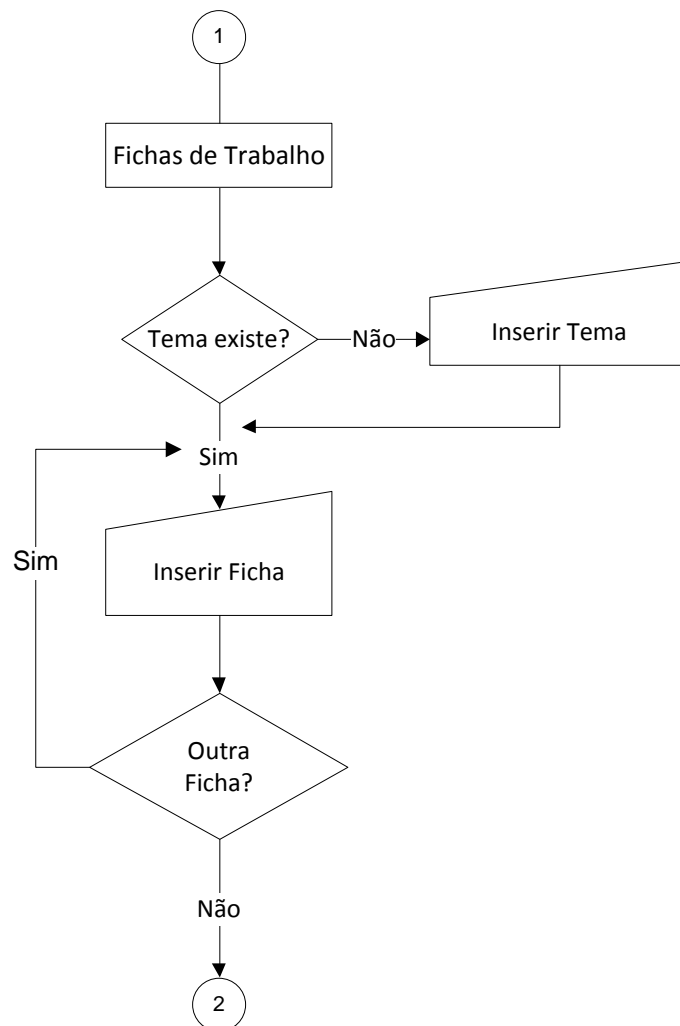


Figura 4.20 Fluxograma respeitante ao professor na atividade “Fichas de Trabalho” no *Software* “Palavras e Palavras”

4.2 Desenvolvimento do Software “Palavras e Palavras”

"Todo animal deixa vestígios do que foi; O homem é o único que deixa pistas do que criou"
Jacob Bronowski (Bronowski, 1976).

Este *software* teve a colaboração do aluno Daniel Grilo do curso de Engenharia Informática e da *designer* gráfica Cristina Rodrigues, cujo trabalho foi publicado (Rodrigues, Grilo, & Rodrigues).

O desenvolvimento do *Software* “Palavras e Palavras” tem como princípio o Modelo Interativo Adaptável Orientado para o Ensino (figura 3.4) o Método Interativo Adaptável Orientado para o Ensino (MIAOE) de acordo com a figura 3.6 e com a tabela 3.4. O processo UID – Análise ao Utilizador conjuga as metodologias de avaliação de *software* educativo (a taxonomia de Bloom revista por Roberto Marzano, a metodologia pedagógica de Reeves e o modelo de avaliação de Fernando Costa (Quadro III e Quadro IV dos anexos G e H). O MIAOE também relaciona as teorias de aprendizagem no processo UID – Interfaces Específicos para o Utilizador. Assim, como os modelos de qualidade, os protótipos de baixa/alta-fidelidade que compõem o processo UID – Objetos Visuais de Desenho.

Em relação às metodologias de desenvolvimento de *software* utilizadas optamos pelo modelo em Cascata, associando-lhe o modelo de prototipagem e o modelo em V, o que está de acordo com a figura 3.5.

O *software* “Palavras e Palavras” tem de obrigatoriamente estabelecer uma interação e interatividade entre os alunos e o *software*, entre os professores/educadores e o *software*. Portanto os ecrãs deste *software* têm de forçosamente respeitar estas premissas. O objetivo do nosso *design* de aprendizagem-ensino é de promover a criação de ecrãs que estabeleçam uma interatividade, usabilidade e adaptabilidade. Como tal, analisamos determinados princípios de composição muito importantes tais como: o equilíbrio visual, o ponto focal, a unidade/harmonia e a cor, para desenvolvermos os nossos protótipos de alta-fidelidade.

4.2.1 Equilíbrio Visual

O equilíbrio visual está relacionado com a importância dada a cada objeto visual e a distribuição desses objetos no ecrã. Segundo Boyle (Boyle, 1997) e Szabo e Kanuka (Szabo & Kanuka, 1999) o equilíbrio visual pode ser obtido através de diferentes processos, entre os quais: simetria e assimetria. A figura 4.21 apresenta a estrutura base de um ecrã, do lado esquerdo existe uma parte superior (título) que referencia o assunto central do ecrã (que está dividido em 2 blocos simétricos), a parte inferior pode indicar uma barra de navegação ou eventualmente o autor/data da realização/etc., o que transmite equilíbrio visual. Enquanto o lado direito do ecrã transmite desequilíbrio visual, provocada pela eliminação de um dos blocos.



Figura 4.21 Equilíbrio visual e desequilíbrio visual, adaptado de (Boyle, 1997)

Para que o equilíbrio visual seja restaurado (ver figura 4.22) é necessário que no lado direito se integre alguns objetos que ocupem a área a cinzento, como se verificara através da figura 4.22.

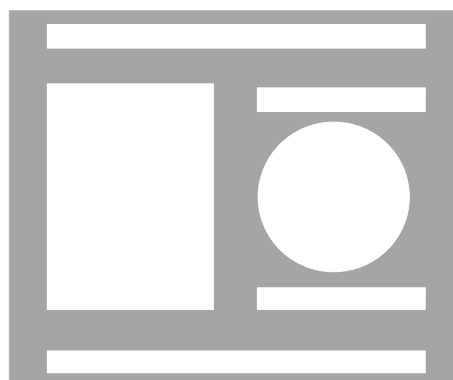


Figura 4.22 Variação na obtenção do equilíbrio visual, adaptado de (Boyle, 1997)

Como tal, no desenvolvimento dos ecrãs do *Software* “Palavras e Palavras” respeitamos as noções transmitidas por Boyle e Szabo e Kanukao em relação ao equilíbrio visual.

4.2.2 Ponto Focal

Porém nos ecrãs deve existir um ponto de destaque, este é referenciado como o ponto focal, logo o utilizador deve ter a sua atenção atraída para esse ponto (Boyle, 1997) e (Szabo & Kanuka, 1999). Contudo deve existir um cuidado na quantidade de objetos a inserir nos ecrãs, pois quantos mais objetos houver mais difícil se torna alcançar esse ponto.

Szabo e Kanuka (Szabo & Kanuka, 1999) fazem alusão a duas técnicas passíveis de dirigir a atenção do utilizador para o ponto focal. O contraste (figura 4.23), que se pode verificar pela diferença existente no terceiro objeto a contar da esquerda. No que refere ao isolamento, um objeto é separado dos restantes objetos, tornando-se no ponto focal mesmo não existindo diferença entre os objetos, como se verifica através da figura 4.24.

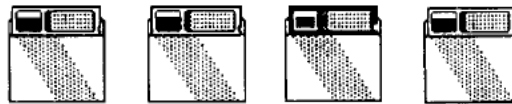


Figura 4.23 Ponto focal por contraste, adaptado de (Szabo & Kanuka, 1999)



Figura 4.24 Ponto focal por isolamento, adaptado de (Szabo & Kanuka, 1999)

Como tal, tivemos alguns cuidados na quantidade e na localização dos objetos a inserir nos ecrãs, de forma a ter presente a unidade/harmonia.

4.2.3 Unidade/Harmonia

A unidade/harmonia deriva do *design* dos objetos, onde a relação entre os objetos é vista como um todo e não separadamente. Uma estrutura conceptual bem delineada em ecrãs permite manter o sentido de unidade e harmonia (Boyle, 1997). Para que a exista unidade e harmonia o espaço entre cada objeto deverá ser menor do que a largura de cada objeto, de

acordo com a figura 4.25. Situação que analisámos em detalhe, para que os objetos inseridos no *software* “Palavras e Palavras” mantenham uma unicidade, equilíbrio e harmonia.

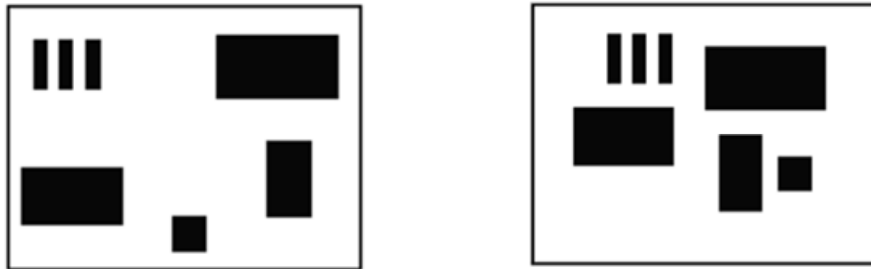


Figura 4.25 Exemplo de uma composição desunificada (à esquerda) e uma composição unificada (à direita), adaptado de (Szabo & Kanuka, 1999)

4.2.4 A cor

É de referir a importância da percepção da cor na visualização da informação, pois está relacionada com o córtex visual, visto o olho humano ter a capacidade de distinguir a cor. A cor é uma realidade sensorial, visto transmitir sensações de movimento, distância, proximidade, espaço, equilíbrio e peso, como tal pode ser portadora de uma mensagem. Para Arnheim (Arnheim, 2002) a cor é um elemento fulcral de qualquer interface de hipermédia, por ser “*a mais eficiente dimensão de discriminação*”, o exemplo que refere é duma bola vermelha a rolar na relva. Esta pode ser localizada e identificada com maior facilidade, porque se distingue do verde da relva.

Para Moderno “*A cor e a luminosidade ou intensidade luminosa são atributos fundamentais de uma mensagem, devendo ser controlados e conhecidos pelo professor. A cor é uma excelente variável seletiva. Combina-se facilmente com outras variáveis e é eminentemente memorável*” (Moderno, 1992).

Desta forma, a cor funciona psicologicamente aumentando ou diminuindo o impacto da informação. Este facto pode ser analisado na perspectiva das cores frias e quentes, visto estas cores nos despertarem sensações. As cores frias vão desde a cor verde passando pelo ciano até à cor violeta e transmitem-nos a sensações de distância, transparência e calma. As cores quentes

vão desde a magenta até ao amarelo, estas cores dão-nos a sensação de estarem mais próximas, densas e estimulantes.

Apesar da importância da cor, esta deve ser usada com moderação, o excesso de cor pode contribuir para desviar a atenção do aluno fase aos conteúdos didáticos que o professor expõe. Porém não existe consenso no número de cores a utilizar num *software*. Para Shneiderman (B.Shneiderman, 1998) o limite máximo é de 4 cores por ecrã e 7 em toda a aplicação, enquanto Stemler (Stemler, 1997) aconselha entre 3 a 6 cores por ecrã. A seleção da cor para o fundo é importante, visto que tem influência nas restantes cores a usar. Para Stemler (Stemler, 1997) e MacFarland (MacFarland, 1995) a cor de fundo deve ser neutra para que sobressaíam as imagens e o texto. Mas, Boyle evoca que o azul pode ser uma ótima cor de fundo “ *Blue, set at the right level of saturation and brightness can be a very effective background colour*” (Boyle, 1997).

Macfarland (MacFarland, 1995) e Stemler (Stemler, 1997) aconselham a utilizar uma estrutura de cor familiar, consistente e com referências em toda a aplicação. No caso de se utilizarem esquemas de cores diferentes em áreas específicas de apresentação, MacFarland aconselha o uso de cores complementares porque aumenta a legibilidade da mensagem. A relação entre o esquema de cores diferentes e o uso de cores complementares denomina-se por contraste, que não é mais do que uma combinação entre cores e tons. Esta combinação tem como base as cores primárias (vermelho, amarelo e azul) visto que cada uma destas cores tem uma cor complementar (verde, laranja e púrpura). É de realçar que a visibilidade das cores diminui à medida que se associam, como tal, devem ser usadas separadamente, (Stemler, 1997).

4.3 Protótipos de Alta-Fidelidade

Tendo em conta estes princípios de composição da cor, do equilíbrio visual e a unidade/harmonia para o desenvolvimento do nosso *software* “Palavras e Palavras”, é um *software* educacional, por isso, tem como base duas vias distintas e complementares:

- A interface cognitiva, relaciona-se com apreensão dos conteúdos apresentados pelo professor/educador, em que uma cor pode gerar emoções; obviamente um conjunto de cores também o faz. Pode fazê-lo de forma harmónica ou por

contraste. Assim, utilizamos as cores quentes para evidenciar o ícone “mocho”. O que está de acordo com as metodologias referidas no capítulo 2 e as metodologias adotadas referidas no capítulo 3 (tabela 3.1);

- A interface física relaciona-se com a percepção do suporte, procura estimular o processo de recepção de conteúdos através da combinação de aspetos visuais e sonoros, de acordo com as metodologias acolhidas referidas no capítulo 3 e de acordo com Boyle na seleção da cor de fundo e com Boyle, Macfarland e Stemler. Ao escolhermos a cor de fundo azul (é uma cor fria) em contraste às cores quentes, criamos o efeito de destaque desejado.

A estrutura de formatação da interface é particularmente importante, visto que, possibilita ao aluno identificação dos itens mais rapidamente, se estes se apresentarem sempre no mesmo sítio, refere Stemler (Stemler, 1997). Logo, criamos a nossa estrutura de interface, de acordo com Stemler e com as “Oito Regras de Ouro de Shneiderman” como se pode ver na figura 4.26. Por outro lado a estrutura de formatação também foca a memória (ver tabela 3.1), situação validada pelo *Software* “Palavras e Palavras”.

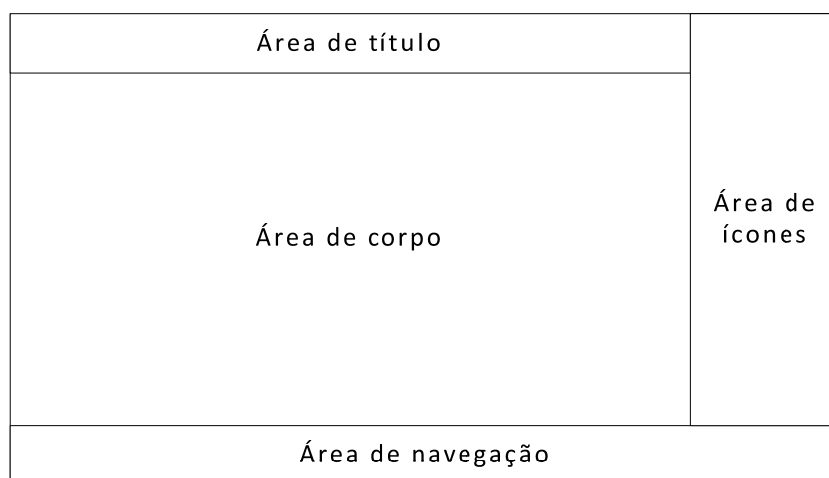


Figura 4.26 Estrutura de Formatação da Interface do *Software* “Palavras e Palavras”

Outro elemento importante no *design* da interface é a forma de disposição da informação na organização visual do ecrã, pois permite que a informação seja localizada como maior ou menor dificuldade. Como tal, houve da nossa parte o cuidado de manter uma consistência da localização da informação ao longo da aplicação, denominada *Software* “Palavras e Palavras”, o que está de acordo com Modelo de Qualidade de *Software* de MacCall, no item referente ao comportamento do produto, essencialmente no subitem usabilidade.

Para além do referido em termos de localização da informação, Dondis (Dondis, 2007) refere que o impulso precativo primário, ou seja, o olho privilegia a zona inferior esquerda de campo visual. Tendo em conta o referido, o córtex visual e de acordo com Leonardo DaVinci no uso da proporção áurea e da regra dos terços. Se aplicarmos este princípio à imagem, as linhas a vermelho na figura 4.27 descrevem as linhas de interesse para a perceção do indivíduo, o cruzamento dessas linhas originam os pontos áureos, considerados os centros de interesse. Inconscientemente o nosso olhar é dirigido para essa zona, ou seja, é nesses centros de interesse e nessas partes das linhas que devem ser colocados os elementos primordiais para a atenção do aluno. Obviamente, não é obrigatório colocar todos os elementos nessa referência, basta haver uma aproximação para a zona central de interesse.

4.3.1 Zona de Interesse

Assim, como se pode verificar através da figura 4.27, a zona de interesse situa-se no cruzamento das linhas, na qual está inserida a parte principal da área de corpo do *software* “Palavras e Palavras”, no quadrante direito em relação à zona de interesse está a área de ícones, a área de título está no quadrante acima da zona de interesse (área de corpo), enquanto a área de navegação se situa no quadrante abaixo da zona de interesse (área de corpo). Porém todas estas áreas estão próximas da área de corpo (zona de interesse).

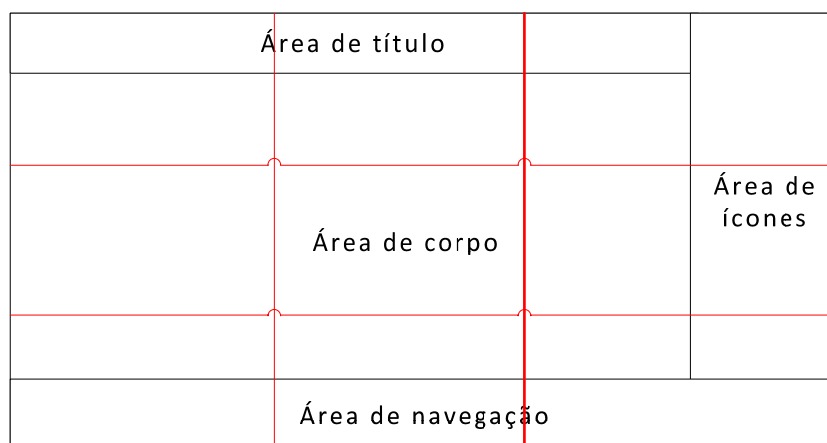


Figura 4.27 Zona de interesse e áreas adjacentes de interesse para o aluno no *Software* “Palavras e Palavras”

Portanto o estímulo visual cria uma moldura de referência abrangida por uma área de intensidade visual, como se pode verificar na figura 4.28, que no caso do *Software* “Palavras e Palavras” se situa na área de corpo.

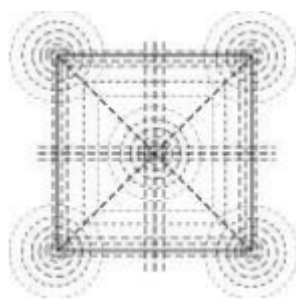


Figura 4.28 As áreas delimitadas a tracejado correspondem a campos de intensidade visual, adaptado (Castela, 2004)

Ao nível da aprendizagem a densidade da informação é fulcral pois quantidade de informação que se apresenta pode tornar processo de aprendizagem hostil ou favorável. Assim, os ecrãs devem ser o mais simples possível. Este aspeto é consensual, porém a quantidade de informação depende de algumas variáveis, tais como: a idade e o nível de conhecimento dos utilizadores (MacFarland, 1995), para nós, o elemento idade é muito importante pois está diretamente relacionada como os aspetos cognitivos e metacognitivos focados no capítulo 2.

4.3.2 Texto Escrito

É de referir que no *software* educacional o “texto escrito” é um dos formatos mais utilizados na apresentação da informação, na ajuda, em comandos, etc. O que contribui para a compreensibilidade e rigor da informação (Alonso & Gallego, 2000). A mutabilidade de visualização do texto (tamanho, cor, tipo, estilo, maiúsculas, minúsculas, etc.) em conjunto com a facilidade de processamento dos dados e armazenamento dos mesmos, tornaram o texto num dos formatos mais utilizados na nossa época, que se pode considerar como a “era do homem digital”.

Porém, a apresentação do texto escrito em interfaces devem ser diferente do texto escrito em papel “... *de datos se pueden ver notablemente modificados si en lugar de analizar la lectura sobre papel, el modo convencional de presentación de la información verbal escrita, se estudia la lectura sobre pantalla ...*” (Diéguez, 1995).

O que vai de encontro à proposta de Shneiderman (Shneiderman B. , 1989), as “três regras de ouro” segundo as quais a informação deve ser organizada em vários fragmentos; os fragmentos relacionam-se entre si; o utilizador precisa somente de uma pequena unidade de informação de cada vez.

Partindo de ambos os princípios e de acordo com a metodologia de ensino em contexto de sala de aula (aplicada pelas professoras do nosso estudo), a construção de uma frase não deve ter palavras inúteis ou desnecessárias.

No *software* “Palavras e Palavras”, a apresentação do texto tem duas vertentes: a relacionada com os conteúdos em contexto de sala de aula e o texto inserido em ícones. Portanto, o texto apresenta diversas funções da linguagem verbal, tal como indica Jakobson (Jakobson, 2003). Das funções indicadas por Jakobson optamos pela função informativa e pela função apelativa. Das funções indicadas por Henriques (Henriques, 2007) selecionamos a função de identificação/associação e a função de seriação. Apresentamos uma breve descrição de cada uma destas funções, através da tabela 4.11.

Funções do Texto	
Função	Objetivos
Função Informativa	Está orientada para o contexto, existe uma informação dos aspetos mais importantes
Função Apelativa	O locutor (mocho) tenta persuadir o aluno (alocutário) a adotar uma determinada ação. O exemplo do botão “Continuar”
Função Seriação	Pretende ordenar palavras, de acordo com uma sequência convencionada, como o alfabeto
Função de identificação/associação	Quando se pretende relacionar a grafia das letras/palavras ou a fonética à imagem referente

Tabela 4.11 Funções do texto no *Software* “Palavras e Palavras”

Para o nosso estudo, o tamanho das letras também é um fator importante, visto que, os estudos de Tinker, referidos por (Diéguez, 1995) indicam que deve existir uma relação entre a idade e o tamanho da letra, como se pode constar na tabela 4.12. O que está de acordo com Jean Piaget e aceite por nós, como uma das teorias de aprendizagem tida em conta (ver a tabela 3.1)

IDADE	CORPO	IDADE
6 Anos	Tipo 18 Tipo 14	3 mm. 2,3 mm.
7 Anos	Tipo 16 Tipo 14	2,7 mm. 2,3 mm.
9 Anos	Tipo 12	2 mm.
10 Anos	Tipo 12 Tipo 10	2 mm. 1,7 mm.
Adultos	Tipo8	1,3 mm.

Tabela 4.12 Valores compreendidos do tamanho da letra em relação à idade, adaptado (Diéguez, 1995)

Obviamente, que não nos podemos esquecer de relacionar a letra à cor, pois esta pode distinguir os diferentes tipos de informação, estabelecer a ligação entre blocos de informação relacionados, entre outros. Contudo, deve existir um contraste entre a cor da letra e cor de fundo do ecrã, para chamar a atenção do aluno e ajuda-lo a recordar-se da localização de determinada informação anteriormente lida. Logo, a combinação da cor das letras/palavras com a cor fundo deve ser de forma a assegurar um elevado contraste entre ambos, como se pode verificar nas imagens relacionadas com o *Software* “Palavras e Palavras”.

4.3.3 As Imagens versus Cor

“... el concepto de imagen comprende otros ámbitos que van más allá de los productos de la comunicación visual y del arte; implica también procesos como el pensamiento, la percepción, la memoria, en suma la conducta. Es, por tanto, un concepto más amplio que el de la representación icónica y ...conviene achar la vista atrás e intentar una explicación del fenómeno partiendo de los orígenes del mismo” (Villafañe, 2006).

No processo ensino-aprendizagem as imagens são uma fonte de aprendizagem e de destaque, independentemente do meio utilizado na transmissão do conhecimento. A imagem só por si transmite um conjunto de informações, por vezes é meramente informativa e é intemporal.

Guerra (Guerra, 1994) considera que a imagem é definida por um conjunto de características: não é mais do que um reflexo que reproduz uma realidade, portanto torna-a

ausente e presente; é simultaneamente um sonho e uma realidade; informa e apela ao sentimento; comporta informações visíveis e ocultas, porque traduz algo explicitamente e algo que é necessário desvendar; a imagem é passado, presente e futuro; a imagem é um fenómeno social e individual; a imagem pode tomar dois sentidos ser concreta e abstrata: concreta porque tem um suporte material, representa um objeto e simultaneamente é um objeto; abstrata porque a imagem pode dizer mais do que está representado, o que nos recorda o adágio popular “ *uma imagem diz mais do que mil palavras*”.

Berrocoso (Berrocoso, 2001) define a imagem segundo três aspetos fundamentais: o suporte da imagem, ou seja o material que (tela, papel, etc.) foi utilizado para materializar a representação icónica da realidade; grau de fidelidade que representam; legibilidade que indica a maior ou menor dificuldade para interpretar a informação visual descrita na imagem.

Mayer (Mayer, 2001) defende que os estudantes, através das imagens, realizam uma aprendizagem construtivista quando selecionam mentalmente material verbal e material gráfico. Para tal, baseia-se na teoria da aprendizagem cognitiva, que defende a existência de dois canais separados e distintos: um de informação visual e outro de informação verbal.

Segundo Pamela Faber (Faber, 2011) as imagens enriquecem as representações linguísticas de conceitos de multimédia fornecendo uma configuração conceitual visual e dinâmica.

Para Rivlin (Rivlin, Lewis, & Davies-Cooper, 1990) as imagens desempenham três tipos de papéis no *design* da interface, independentemente da área para a qual as imagens vão ser usadas, de acordo com a tabela 4.13. A inclusão de uma imagem deve traduzir o objetivo em causa, este por sua vez deve ser decidido desde o início, pois “*uma imagem só será eficaz se desempenhar exatamente o papel para que foi prevista*” (Rivlin, Lewis, & Davies-Cooper, 1990).

Imagem	
Papel	Função
Absoluto (a informação é inserida na própria imagem)	Convencer; Informar
De colocação estratégica (em relação ao programa)	Focar a atenção; Encaminhar; Realçar a informação
De reação do utilizador (qualidades catalíticas)	Desenvolver informação visual complementar; Estimular ideias; Deslocar para outra parte do programa

Tabela 4.13 Papel da imagem versus função no *design* do ecrã, adaptado de (Rivlin, Lewis, & Davies-Cooper, 1990)

Para as teorias cognitivas de Gestalt, a imagem é uma concretização material de uma série de formas globalmente reconhecidas pelo receptor. Portanto para o nosso *software* a imagem pode ser “tudo ou nada”: tudo se o aluno a reconhecer; nada se o aluno não a conseguir identificar. Logo, a imagem é um recurso educativo preponderante, e constitui o foco essencial duma grande parte das aplicações a nível de *software* educativo/educacional.

No nosso estudo, a imagem é um elemento fulcral no processo ensino-aprendizagem, é usada de forma interativa, permite explicar determinadas ações e estimular a abordagem aos conteúdos lecionados em contexto de sala de aula. Para além da sua utilização em termos didáticos, a imagem assume um papel crucial ao nível de funcionamento e funcionalidades quando é usada em menus, botões/ícones.

Ao integramos imagens no *software* “Palavras e Palavras” de acordo com Diéguez (Diéguez, 1995), seguimos a sua proposta de utilização de imagens pelas suas funções didáticas. Diéguez propôs sete funções para a utilização de imagens (função motivadora, função vicarial, função catalisadora de experiências, função informativa, função explicativa, função redundante, função estética). No entanto, nós optámos por usar três funções: função catalisadora de experiências, função informativa, função estética.

Henriques (Henriques, 2007) acrescenta outras funções didáticas para a imagem, sugere cinco funções, sendo elas: a função de identificação/associação, função de classificação, função de seriação, função espacial e a função aritmética. De entre todas optamos por uma: função de identificação/associação.

A tabela 4.14 apresenta uma descrição resumida dos objetivos das funções das imagens que tivemos em conta no desenvolvimento do *software* “Palavras e Palavras”. As restantes funções de Henriques e de Diéguez não foram utilizadas, visto que não conferiam a este *software* mais alguma valia em termos didáticos.

Funções Didáticas da Imagem	
Função	Objetivos
Função Catalisadora de Experiências	A imagem deve traduzir uma realidade com o objetivo de facilitar a sua compreensão
Função Informativa	A imagem assume em primeiro lugar o discurso didático, deixando para segundo plano o discurso verbal
Função Estética	A imagem é utilizada para dar equilíbrio a um texto, dar cor a um espaço
Função de Identificação/Associação	A imagem é utilizada para reconhecer características comuns noutra imagem

Tabela 4.14 Funções didáticas da imagem adotadas no *Software* “Palavras e Palavras”

É de referir que a cor nas imagens assume um papel importante, pois a cor permite que a imagem possa assumir um caráter atrativo ou não. Podemos concluir que as cores também comunicam, basta observarmos que as cores das flores atraem os insetos para a polinização. As cores acompanham-nos no nosso dia a dia, quantas vezes ouvimos dizer: “*fiquei roxa de raiva*”; “*o dia está tão cinzento*”. De certa forma as cores immortalizam a espécie.

Mehta e Zhu (Mehta & Zhu, 2009), no estudo “*Blue or Red? Exploring the Effect of Color on Cognitive Task Performances*” sobre o efeito das cores nas tarefas cognitivas, concluíram que o vermelho favorece as atividades que requerem maior precisão enquanto o azul promove a criatividade. Fator que tivemos em consideração, na cor escolhida para o fundo do ecrã e de acordo com (Boyle, 1997), como já foi referido.

A utilização de cores em *software* educativo implica chamar a atenção para determinadas situações e ícones. Logo, a cor pode estar ligada à função explicativa das imagens didáticas, portanto o apelo à cor é praticamente irresistível.

4.3.4 As Imagens versus Movimento e Som

Para Monet (Monet, O Multimedia, 1997) “*O movimento e a expressão visual, veiculados pela imagem animada, são necessários para a compreensão e comunicação*”.

As imagens podem ser estáticas e de sequências animadas o que permite serem usadas em ambientes de aprendizagem. Assim, ao associarmos movimento à imagem tornámo-la dinâmica o que reforça o processo de aprendizagem, visto facilitar a memorização da informação e fortalecer a motivação e a atenção. Como tal, constitui um excelente recurso pedagógico.

O movimento nos ecrãs permite concretizar alguns objetivos das aplicações, visto que sugere uma ordem de visualização e exploração da informação; sugere ao utilizador a seleção de determinada ação; passa de uma transição visual para outra de forma suave.

Para tal, o movimento deve ser fluido e agradável, logo, as imagens devem suceder-se a uma velocidade de 16 *frames* por segundo (Santos, 2003). Santos realça que a animação deve poder ser interrompida, exceto se fizer parte de um processo, logo, deve ser independente. Situação auferida no *Software* “Palavras e Palavras”.

Se a esse movimento estiver associada a interatividade, a motivação também pode ser adquirida através do som, então o processo de aprendizagem é duplamente reforçado, visto que pode transmitir sensações, emoções, dar ritmo e movimento (Monet, O Multimedia, 1997), o que permite aumentar o nível de motivação e aumentar o impacto da informação nas situações de aprendizagem. Os efeitos sonoros criam situações de maior realismo.

O ato de aliar imagem ao som vai funcionar como um segundo comentário o que completa e aprofunda o seu significado, chama a atenção e destaca a ação. Assim, e segundo Santos (Santos M. , 2003) o som pode contribuir no aumento da motivação do aluno desde que seja usado com moderação.

Boyle (Boyle, 1997) divide o som em categorias distintas: fala, música e efeitos sonoros. Por sua vez, Santos (Santos M. , 2003) partilha da opinião que o discurso oral (fala) tem diversas funções, tais como: comentário, narração, diálogo e discurso direto.

Situação criada no *software* “Palavras e Palavras” no qual a voz foi usada para explicar os diversos elementos da interface, a forma de interagir com as atividades propostas através das imagens no ecrã, dar instruções e *feedback* perante as ações do aluno, para que este fosse incentivado ou premiado perante os seus atos. Nesta situação, optamos por utilizar o discurso direto, a personagem “mocho” fala diretamente com o aluno, motiva-o utilizando expressões tais como: “*És um crânio!*”, “*És o máximo!*”, “*Tu consegues, tenta mais uma vez*”, “*Tu és capaz!*”.

Consideramos ainda a narração na função informativa, no sentido de incluir orientações relativas à imagem que é apresentada, na qual o texto e a narração devem ser homogéneas. A narração no *Software* “Palavras e Palavras” foi usada para expor informação concreta, que

incluiu orientações para completar uma tarefa ou informação descritiva respeitante à imagem que é apresentada.

Como tal, no *software* “Palavras e Palavras” o “mocho” conduz a atenção do aluno para o ponto focal da imagem a visualizar, orientar e elucidar, o que está de acordo com Santos (Santos M. , 2003).

Desta forma o “mocho” orienta e informa o aluno como é que ele deve interagir com a aplicação. A função apelativa de Jakobson também foi utilizada, assim como a função poética no sentido de apelar e motivar o aluno, perante uma futura ação ou perante o resultado de uma ação. A função indagadora também foi apadrinhada por nós, no momento em que o aluno responde, através de um clique a uma determinada situação, como o querer “*continuar*” ou “*sair*”, Henriques (Henriques, 2007) . Como se pode verificar através da tabela 4.15.

Funções do Som (Discurso direto)	
Função	Objetivo
Função Informativa	Está orientada para o contexto, existe uma informação dos aspetos mais importantes
Função Apelativa	O locutor (mocho) tenta persuadir o aluno (alocutário) a adotar uma determinada ação. O exemplo do botão “Sair”
Função Poética	Quando o locutor (mocho) se apoia na mensagem, nos sons para transmitir sugestões e sentimentos, exemplo “Tu és capaz”
Função Indagadora	O alocutário tem de responder a uma questão formulada, mesmo que a sua resposta seja efetuada por uma ação, exemplo de clicar sobre um ícone/botão

Tabela 4.15 Funções do som (através do discurso direto) no *Software* “Palavras e Palavras”

Segundo Boyle (Boyle, 1997) o discurso ao associar o canal auditivo com o córtex visual melhora o conteúdo da informação, visto estimular a natureza multissensorial do ser humano.

As citações e indicações de Santos, Boyle, Diéguez, Stemler tocam as teorias e os modelos de aprendizagem: a teoria cognitiva, a teoria de inteligências múltiplas (as quais abrangem uma determinada zona do cérebro), teoria Behaviour de Watson, Skinner e de Bandura, teoria comportamental, a memória e os modelos mentais. Todas estas teorias de aprendizagem foram adotadas para o nosso estudo do modelo IAOE.

No *Software* “Palavras e Palavras” o som por vezes é uma tradução da imagem, visto que reforça o papel da mesma, outras vezes o som é paralelo à imagem, em que os signos linguísticos e icónicos se combinam de forma a fomentar a eficácia das duas linguagens. No entanto, foi usado com moderação no auxílio do *design* e dos conteúdos de informação de forma a contribuir no processo de aprendizagem e simultaneamente permitir uma interatividade e usabilidade.

Naturalmente que as funções descritas acima (funções do som, funções didáticas da imagem e as funções do texto) estão diretamente relacionadas com o feedback, que por sua vez está a “alimentar” a interatividade do *Software* “Palavras e Palavras”.

De acordo com tudo o que foi referido o Modelo SIOE é testado através do *software* “Palavras e Palavras”, que permite um ambiente de ensino-aprendizagem, ao criar situações de tirocínio e de avaliação de acordo com as características individuais dos alunos o que aponta para a metodologia pedagógica de Reeves e para o modelo de avaliação de Fernando Costa (ver tabela 3.2).

4.3.5 Desenvolvimento dos Protótipos de Alta-Fidelidade (versão do aluno)

Para nós, o aspeto visual da interface do *Software* “Palavras e Palavras” foi muito importante, pois é através dela que o aluno se pode motivar ou desinteressar, por isso, contemplamos os aspetos descritos no início deste capítulo. Como tal, toda a interface do *software* “Palavras e Palavras” foi desenhado atendendo às necessidades específicas dos alunos.

Este tipo de interfaces tem uma natureza multimodal, pois fornece ao aluno (utilizador) uma multiplicidade de interações e de dispositivos de entrada, com o objetivo de facilitar o acesso e exploração da informação. Os interfaces relacionados com o *software* educacional que permitem uma interação com o aluno devem integrar: menus, janelas, ajudas à navegação, ajudas ao documento (caso haja), caixas de diálogo, botões, refere Jones e Okey (G. Jones & Okey, 1999) e Ribeiro (Ribeiro, 2007), situação existente no nosso *software* “Palavras e Palavras”, como se pode verificar através das imagens que evocam o *software*.

As ajudas verbais do *Software* “Palavras e Palavras” transmitidas através do “mocho” desenvolvem a autonomia e orientam o aluno sob o modo como tem de navegar e interagir com esta aplicação.

Ao executar a aplicação surge o ecrã inicial do *software* “Palavras e Palavras” com uma pequena animação que consiste no aparecimento das letras, uma a uma, até formarem a frase “Palavras e Palavras” que foi implementada com base na da tecnologia Flash, como se pode constatar pela imagem 4.4. Existe também uma pequena canção a convidar o aluno a entrar no *software* ver uma parte do excerto do código na ilustração 4.1.

```
private void ABC_Load(object sender, System.EventArgs e)
{
    comum.PlaySound("jogoabcde.wav",Application.StartupPath);
    myResources.ClassResources mr = new myResources.ClassResources();
    string tmp_file = mr.CreateTemp("mouse.ico",Application.StartupPath);
    this.Cursor = new Cursor(Application.StartupPath+"\\mouse.ico");
}
```

Ilustração 4.1 Excerto do código da canção do *Software* “Palavras e Palavras”

Em relação à cor de fundo do *design* de interface de toda a aplicação, optamos pela indicação do fundo azul proposto por (Boyle, 1997), e pela utilização da gama de cores quentes, da gama cores frias e do uso de cores complementares (cada cor primária tem uma cor complementar), o que é sugerido por Shneiderman (Shneiderman B. , 1989)) e por (MacFarland, 1995).



Imagem 4.4 Ecrã de Inicial do Software “Palavras e Palavras” (Rodrigues, Grilo, & Rodrigues)

O *software* “Palavras e Palavras” permite ao aluno navegar pelo espaço de informação através de acesso a botões, zonas interativas e sob a forma de opções de um menu. A interatividade é representada pelo ícone o “mocho”, no entanto este ícone assume papéis distintos no decorrer do *software*, como se irá descrever ao longo deste ponto.

A imagem 4.5 representa o segundo ecrã do *software* “Palavras e Palavras”, que mostra a lista de todos os alunos já adicionados à aplicação (ver ilustração 4.2), o que permite escolher um desses alunos, para poder realizar as atividades propostas no *software*. Para tal, o aluno deve “clicar” no botão “continuar”.

```
// Show player names into a select box
private void ShowPlayers ()
{
    lstjogadores.Items.Clear();
    for (int i=0; i < jogadores.Count; i++) {
        Player j = (Player)jogadores[i];
        lstjogadores.Items.Add(j.Name.ToString());
    }
}
```

Ilustração 4.2 Excerto do código que mostra a lista dos alunos no *Software* “Palavras e Palavras”

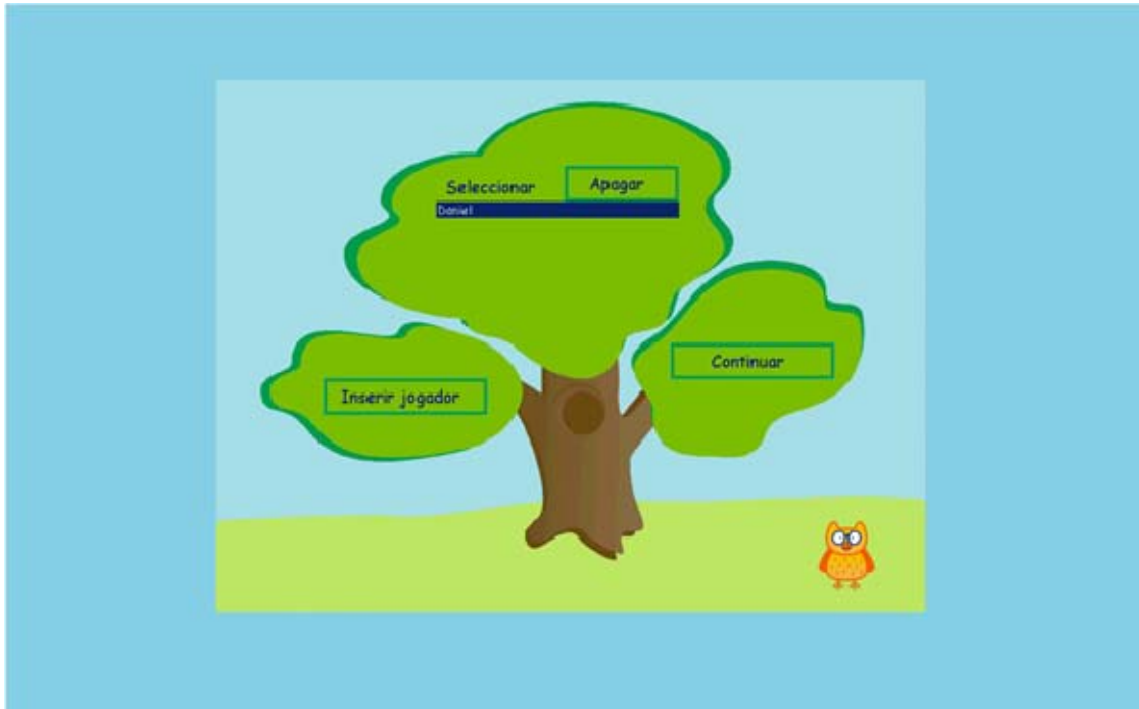


Imagem 4.5 Segundo ecrã do Software “Palavras e Palavras” (Rodrigues, Grilo, & Rodrigues)

É de referir que os botões são áreas individuais e isoladas do ecrã, e que podem ser seleccionados pelo aluno através do “clique” no rato. O Software “Palavras e Palavras” tem botões permanentes que surgem no mesmo local nos diferentes ecrãs, para assim facilitar ao aluno interiorizar a sua localização e a sua função.

Neste ecrã é permitido apagar todos os alunos criados, através do botão “Apagar” que se encontra em cima da lista de utilizadores. Se o utilizador pretender apagar apenas um jogador, basta que selecione o jogador que pretende remover e pressione a tecla “delete” do seu teclado. No mesmo ecrã, há a possibilidade de criar novos alunos para interagirem com o *software* de acordo com a imagem 4.6 e com a ilustração 4.3. No momento da criação de um novo aluno existe a componente da distinção do sexo, ao “ao clicar” nas imagens de menino ou menina. É de salientar, que o jogo interage com o aluno (utilizador) através de som que contém uma explicação breve das ações que este pode realizar. Essa interação é efetuada através do “mocho”.

```

// Add new player
private void lblInsererJogador_Click(object sender, System.EventArgs e)
{
    NewPlayer np = new NewPlayer(this.jogadores);
    np.Show();
    this.Hide();
    this.Close();
}

```

Ilustração 4.3 – Excerto do código para criar um jogador do Software “Palavras e Palavras”

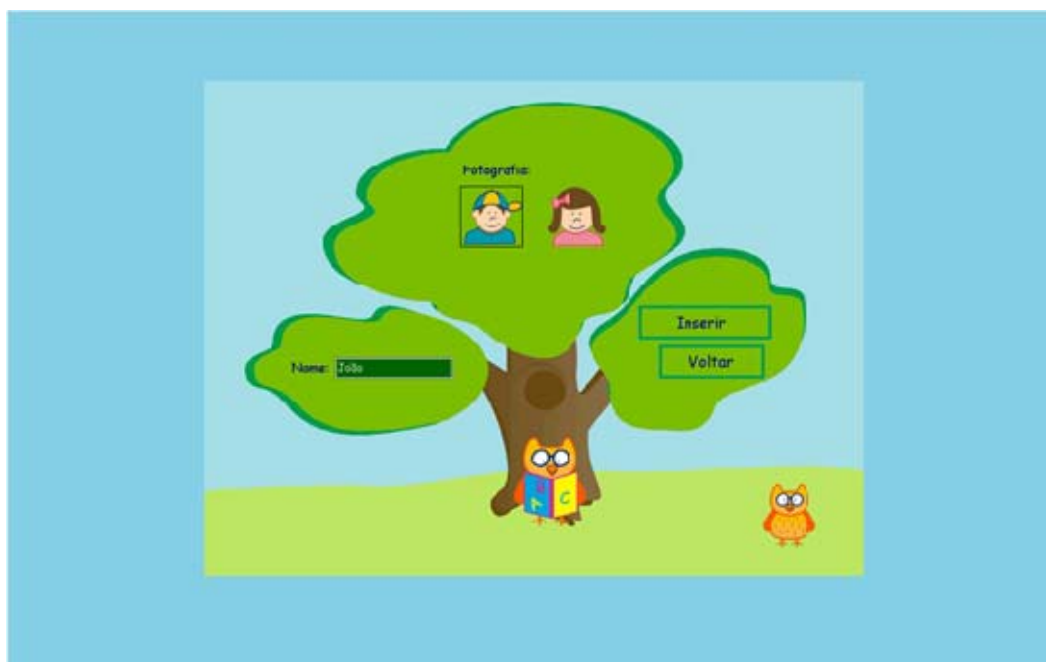


Imagem 4.6 Ecrã de inserção do aluno no Software “Palavras e Palavras” (Rodrigues, Grilo, & Rodrigues)

Após o aluno selecionar o seu nome passa para o próximo ecrã, como se pode verificar na imagem 4.7. Na área do corpo (zona de enfoque) temos a árvore com as três copas, cada uma delas representa um dos três diferentes jogos. A barra inferior é representada pela sensação de visualização de relva, e simboliza o menu onde existem todas as opções relativas a outras atividades a realizar pelo aluno.

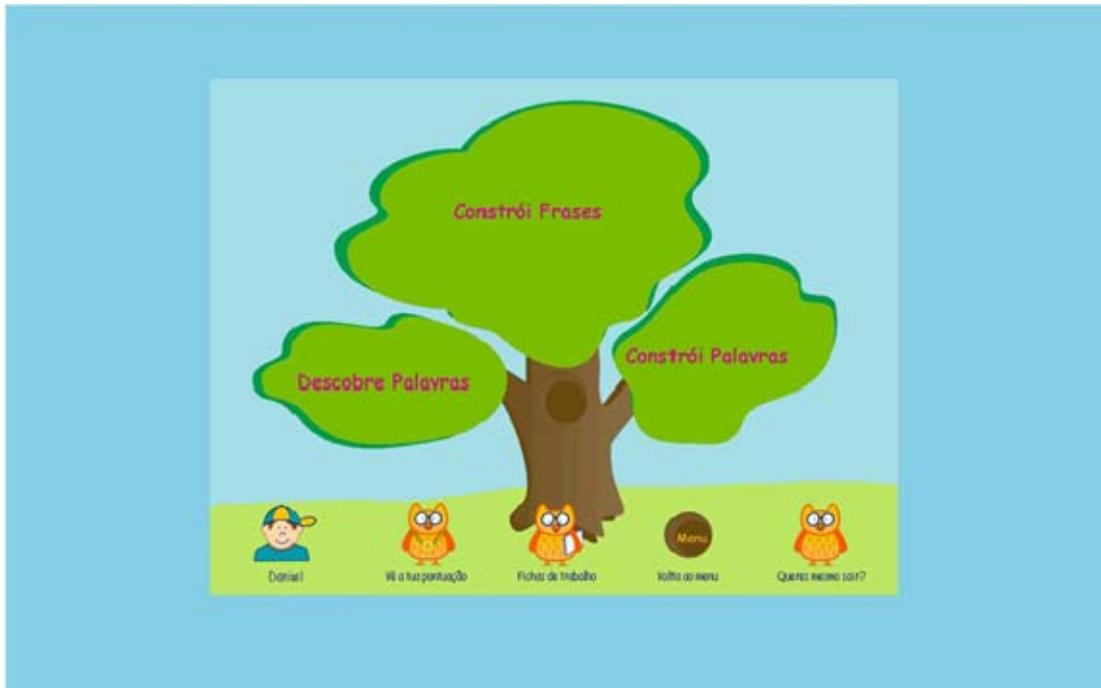


Imagem 4.7 Ecrã das tarefas a realizar pelo aluno no Software “Palavras e Palavras” (Rodrigues, Grilo, & Rodrigues)

O ecrã representado pela imagem 4.7 funciona de forma um pouco diferente dos ecrãs anteriores, pois é composto por duas partes distintas: o “menu”, e a zona que contém os jogos a selecionar pelo aluno.

Convém referir que a partir do momento em que nos encontramos neste ecrã, a zona referida como “menu” é comum para os restantes ecrãs do *software* “Palavras e Palavras”.

O “menu” consiste num componente especial denominado *usercontrol*. Estes componentes podem ser incluídos em formulários e interagir com eles através de alguns atributos, como por exemplo o *parentForm*. O “menu” apresentado na imagem 4.7 é do tipo “*menus pin-up*”, porque se situa numa determinada posição do ecrã e permanece nessa posição até ser eliminado do mesmo, (Ribeiro, 2007). O componente é composto por vários ícones de mochos, que correspondem a ferramentas distintas. Estas ferramentas são ativadas por “*clicks*” do rato sobre o “mocho” que se pretende selecionar.

Horton (Horton, 1994) define ícones como “*pequenos símbolos picturais utilizados em menus, janelas e ecrãs de computador para representar determinadas habilidades do sistema, cujo uso é determinado pelos mesmos.*”

A restante zona do ecrã é um formulário vigilante a todas as movimentações que o aluno faz com o rato. Assim, quando o rato passa sobre determinadas zonas do ecrã, o *software* interage com o utilizador através de sons que descrevem cada um dos três jogos. O que permite ao aluno escolher um dos jogos.

O Jogo Descobre Palavras (DP)

Este jogo tem como objetivo desenvolver a memorização dos alunos. Após escolher o jogo, surge um novo ecrã onde existe uma lista de jogos, representados por uma letra do abecedário, de acordo com a imagem 4.8. Ao escolher uma dessas letras, implica que esse jogo só tem imagens e palavras ou somente palavras (depende no nível) que comecem com essa letra. Porém, o jogo escolhido é selecionado aleatoriamente da lista de jogos que começam por essa letra. Estes jogos foram anteriormente adicionados à aplicação através da versão docente.

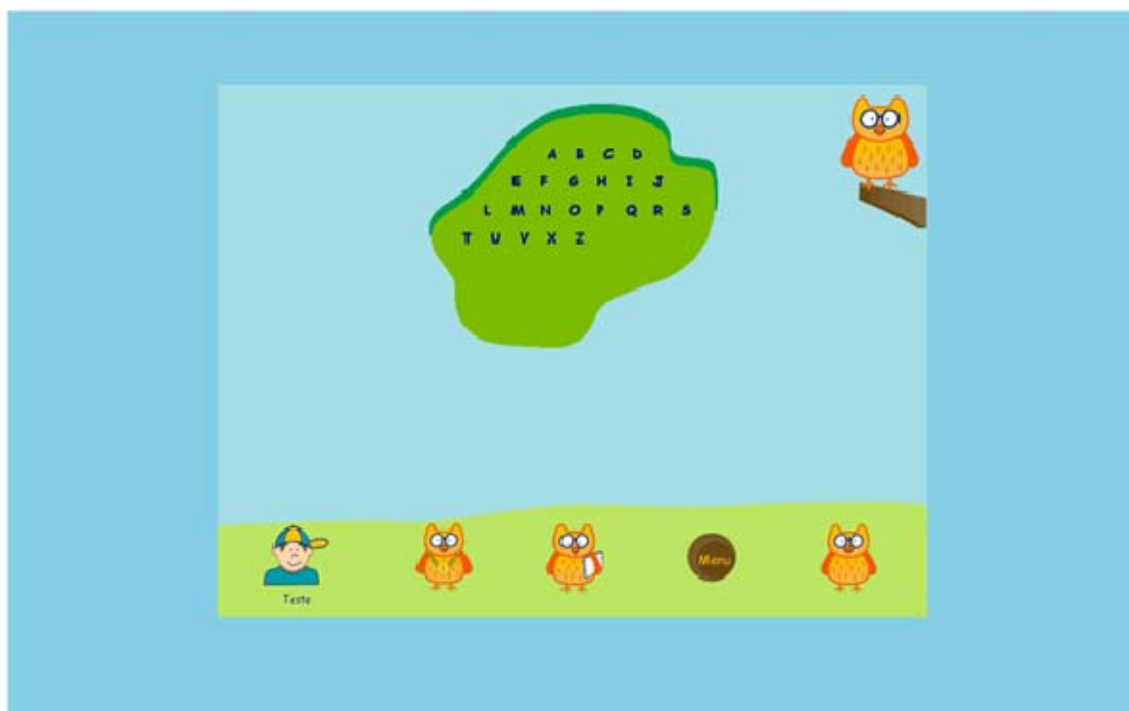


Imagem 4.8 Ecrã de seleção da letra para o Jogo DP do Software “Palavras e Palavras”

De acordo com a análise de requisitos efetuada, a correspondência entre as imagens é realizada entre 2 imagens e 2 duas palavras iguais associadas às imagens, porém estas surgem sempre em posições diferentes no ecrã. Todavia este jogo apresenta 3 níveis de dificuldade (ver figura 4.16). Para o nível 1, o jogo consiste em ter 12 janelas onde aparecem as várias imagens e

correspondentes palavras, como se pode verificar através da imagem 4.9. Neste ecrã podemos constatar o papel das funções de texto, funções de imagem e as funções de som.



Imagem 4.9 Ecrã do jogo DP nível 1 do Software “Palavras e Palavras”, versão aluno

Ao passar com o rato pelo ícone “Níveis” funciona como um “*menu-fall-down*”, visto que se expandem para baixo mal são selecionados. Neste jogo e em todos os outros, o ícone “Níveis” surge no quadrante superior direito do ecrã, assim como o “mocho” que tem como objetivo motivar e congratular o aluno nos diferentes jogos. Logo, o “mocho” está em completa interatividade com o aluno.

Após surgir este ecrã, o aluno tem durante algum tempo a oportunidade de memorizar as janelas que têm os mesmos objetos e palavras associadas. Visto que, passado esse tempo, as janelas são fechadas e o aluno vai ter de descobrir, o que cada janela esconde. Esta técnica é realizada com a utilização de *timers* (temporização de processos), como se verifica através do excerto do código, representado pela ilustração 4.4. O aluno tem 5 segundos para memorizar a posição dos objetos e as respetivas palavras.

```
ShowWindows();
this.timer3_.Interval = 999 * 5;
this.timer3_.Start();this.timer3_.Tick += new EventHandler(Timer3__Tick);
```

Ilustração 4.4 Excerto do código da utilização de timer no jogo DP do Software “Palavras e Palavras”

Antes de as janelas serem mostradas, é imperioso associar posições diferentes para cada imagem. Como tal, o sistema faz uso de uma função que passa por cada uma das 6 janelas do jogo e associá-la a duas posições forçosamente diferentes entre elas, ou seja, as janelas são abertas aleatoriamente entre intervalos, como se pode na imagem 4.8 e no excerto do código desenvolvido através da ilustração 4.5.

```
private void RandomizeWindows () {
    // initialize windows with -1
    janelas_associadas[0] = -1;
    for (int i=1; i < janelas_associadas.Length; i++) {
        janelas_associadas[i] = -1;
    }

    for (int i=0; i < this.janelas.Count; i++) {
        int rand = getRandomNumber();
        int rand2 = getRandomNumber();
        while (janelas_associadas[rand] > -1)
            rand = getRandomNumber();
        while (janelas_associadas[rand2] > -1)
            rand2 = getRandomNumber();
        janelas_associadas[rand] = i;
        janelas_associadas[rand2] = i;
    }
}
```

Ilustração 4.5 Código para a associação das janelas no jogo DP do Software “Palavras e Palavras”

Outra função importante deste ecrã é a do evento dos “clicks” do rato em cada janela. Esta função é de extrema importância para o funcionamento correto da aplicação. Ela é chamada cada vez que há um “click” do rato (ou noutro dispositivo de *input* que funcione da mesma maneira), em cada uma das janelas. Uma vez que o procedimento é igual para todas as janelas, esta é executada sobre todas elas e posteriormente faz a distinção para saber concretamente a

janela em que o utilizador carregou, como se pode ver partes dos excertos da função no anexo M.

Esta função também trata de outras situações existentes no *software*, entre elas: a situação das imagens inseridas serem maiores que o espaço disponibilizado pela própria janela; A situação do mocho ter movimentos nas asas e no bico à medida que a aplicação interage com o utilizador através de um som (função *Mocho_Fala()*) que consistem em 2 *timers*, são executados com tempos ao acaso até 500ms); A situação do aluno errar a resposta certa, a aplicação espera 2 segundos para que o mesmo tenha tempo de ver o que errou e possa relembrar a posição das janelas que seleccionou.

No primeiro nível 1 deste jogo, o aluno não precisa de saber ler, visto que as imagens surgem aos pares com as palavras escritas da mesma cor e utilizando cores diferentes para imagens diferentes. O aluno poderá jogar através da memorização de imagens ou através da correspondência entre as palavras de cores iguais. No segundo nível, as cores das palavras são todas da mesma cor (preto) o que faz com que o grau de dificuldade aumente, apesar das imagens se manterem. Por último, no terceiro nível, o jogo é apenas composto por palavras da mesma cor, estando elas relacionadas como os objetos mostrados nos níveis anteriores. Assim, o aluno tem que conseguir fazer corresponder palavras iguais.

Aquando da seleção dos níveis do jogo, utilizamos o tipo de “*menu fall-down*” que se expandem para baixo mal são seleccionados.

O Jogo Constrói Palavras

Este jogo permite ao aluno formar palavras a partir das letras do alfabeto, em que cada letra associada por uma determinada ordem constitui a palavra.

Assim, o aluno tem arrastar as letras que estão numa posição calculada ao acaso na copa (elemento constituinte da árvore, figura simbólica da aplicação do jogo) para a sua posição certa a fim de constituir a palavra. As letras aparecem no ecrã do jogo em posições aleatórias cujas coordenadas x e y são definidas numa função de seu nome *getRealPosition()*. Esta função faz corresponder a 47 posições de um *array*, as 47 localizações de cada letra. Portanto, o jogo está limitado a um número máximo de letras para cada uma das 4 palavras o que resulta numa média

de praticamente 12 letras por palavra. A questão do tamanho da letra é de acordo com a proposta das professoras que participaram neste projeto e com as indicações propostas na tabela 4.12.

Este jogo também tem três níveis de dificuldade. Sendo, o primeiro nível composto por palavras com cores diferentes e uma imagem que corresponde a essa palavra, como se pode verificar através da imagem 4.10.



Imagem 4.10 Ecrã do jogo CP do Software “Palavras e Palavras” (Rodrigues, Grilo, & Rodrigues)

No segundo nível, apenas são apresentadas as letras de cores diferentes. Neste caso, o aluno terá de construir as palavras fazendo correspondência com as cores das letras. No terceiro nível, as palavras surgem todas da mesma cor. A única ajuda que o utilizador poderá ter nesse caso, é o facto da primeira letra estar em maiúscula. Esta questão depende da forma como a professora introduziu as palavras na aplicação.

Este jogo funciona de uma forma muito idêntica com o jogo anterior, pois faz uso de algumas funções comuns a toda a aplicação, mas depende essencialmente da técnica “*drag and drop*”. A técnica consiste em pressionar uma letra e arrastá-la para o retângulo correspondente segundo a ordem definida pela palavra que o aluno pretende construir.

O Jogo Constrói Frases (CP)

Neste jogo, o aluno deve construir frases, de acordo com as palavras que lhe surgem na copa. Para tal, deve arrastar palavras de uma determinada posição, para a posição certa e pela ordem correta para que estas mesmas palavras formem a frase. Este jogo faz uso das mesmas funções do anterior, sendo que as principais diferenças são o facto das “Labels” a arrastar serem constituídas por várias letras e não só por uma e as posições onde as palavras são apresentadas no ecrã, que podemos visualizar na imagem 4.11.

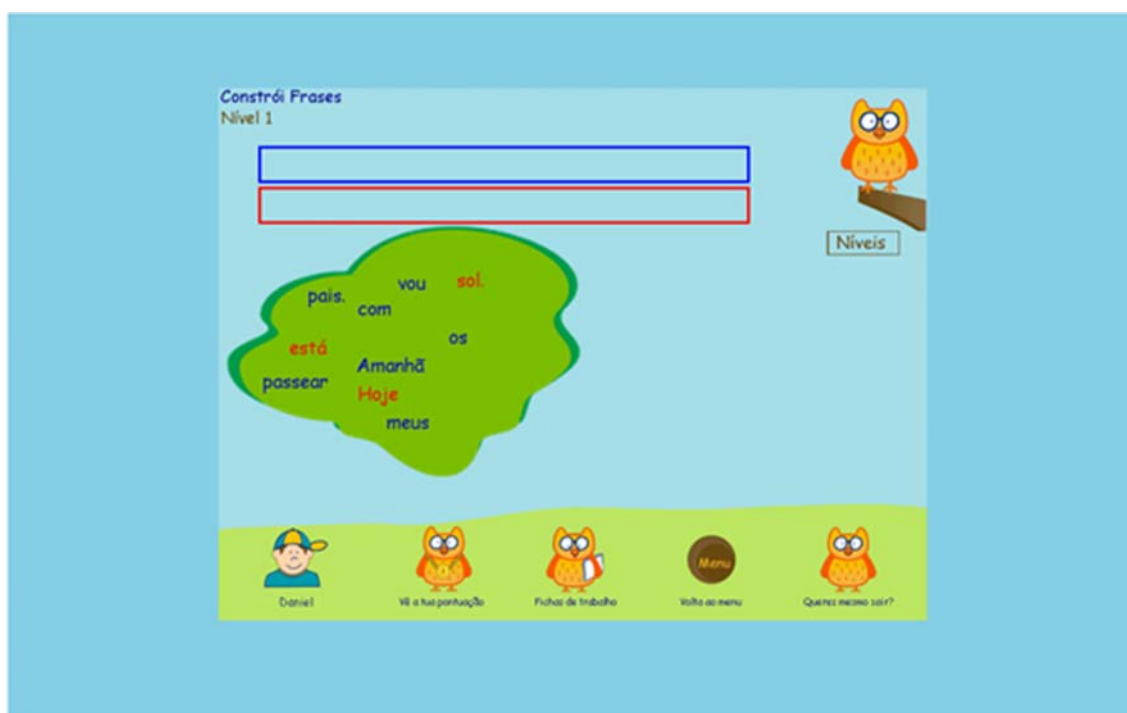


Imagem 4.11 Ecrã do jogo do Software “Palavras e Palavras”

É de referir que este jogo é constituído por dois níveis de dificuldade. No primeiro nível, as frases estão escritas com cores diferentes (vermelho e azul, cores primárias). No último nível, as palavras são apresentadas todas da mesma cor o que faz com que o nível de dificuldade aumente.

Fichas de trabalho do Software

Esta atividade permite ao aluno imprimir fichas de exercícios que são inseridas através da versão do professor. Estas fichas estão organizadas segundo vários temas (ex: Língua Portuguesa – complementos de verbo, complemento circunstancial, complementos de nome, verbos, entre

outros). Ao imprimir uma ficha, o aluno pode então marcá-la com um “check”. Desta forma o aluno e o professor verificam as fichas que ainda lhe faltam imprimir, para que as possa realizar. Para imprimir as fichas de trabalho, basta “clique” no botão da impressora. Este é um botão gráfico, porque contém uma imagem que remete a um tópico que o aluno irá encontrar após a ativação, (Ribeiro N. , 2007).

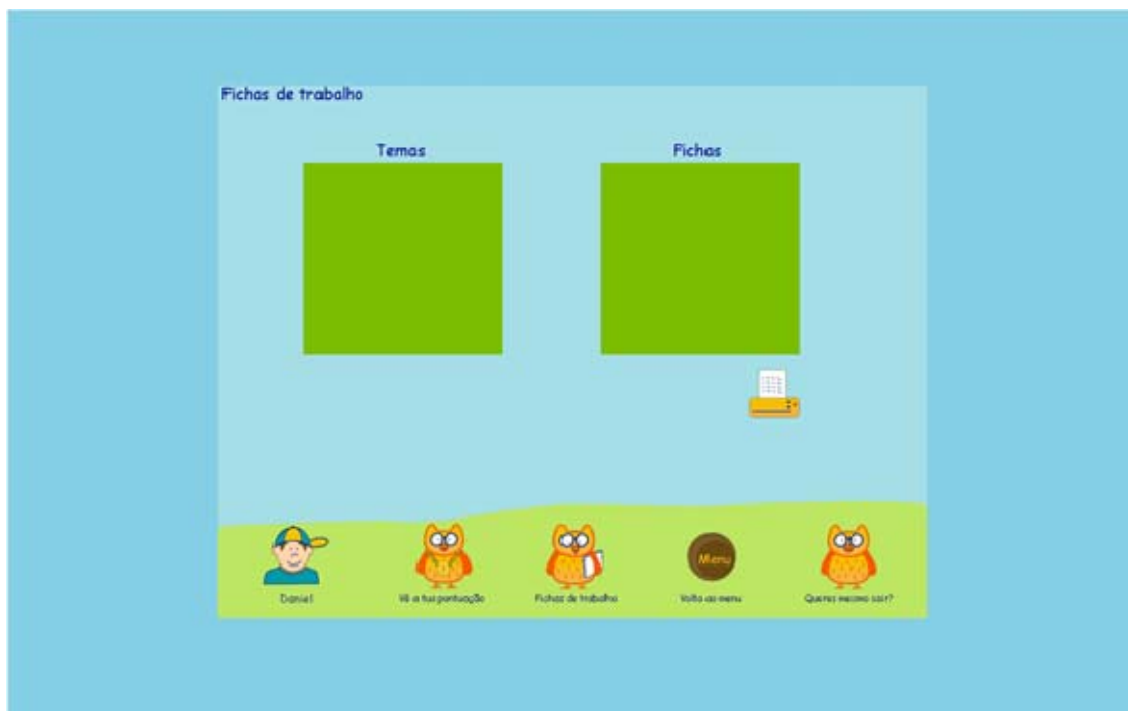


Imagem 4.12 Ecrã Fichas de Trabalho do Software “Palavras e Palavras”

A imagem 4.12 representa o ecrã das “Fichas de Trabalho”, que é composto por duas listas: a lista dos temas e a lista das fichas. Quando o aluno seleciona um tema, a lista de fichas atualiza-se automaticamente para disponibilizar a informação em conformidade.

Pontuação do aluno

Através do ecrã o aluno pode verificar o número de respostas certas e erradas que obteve durante os jogos que realizou. O resultado das respostas certas e erradas que o aluno dá em cada jogo, surgem nas copas dos respetivos jogos. No canto superior esquerdo surge a área de identificação do aluno, ou seja o nome do aluno que tem estado a jogar, podemos ver esta situação na imagem 4.13.

Caso o aluno pretenda voltar a jogar qualquer um dos jogos, basta “clique” no “botão menu”, que se situa na área de navegação do menu.

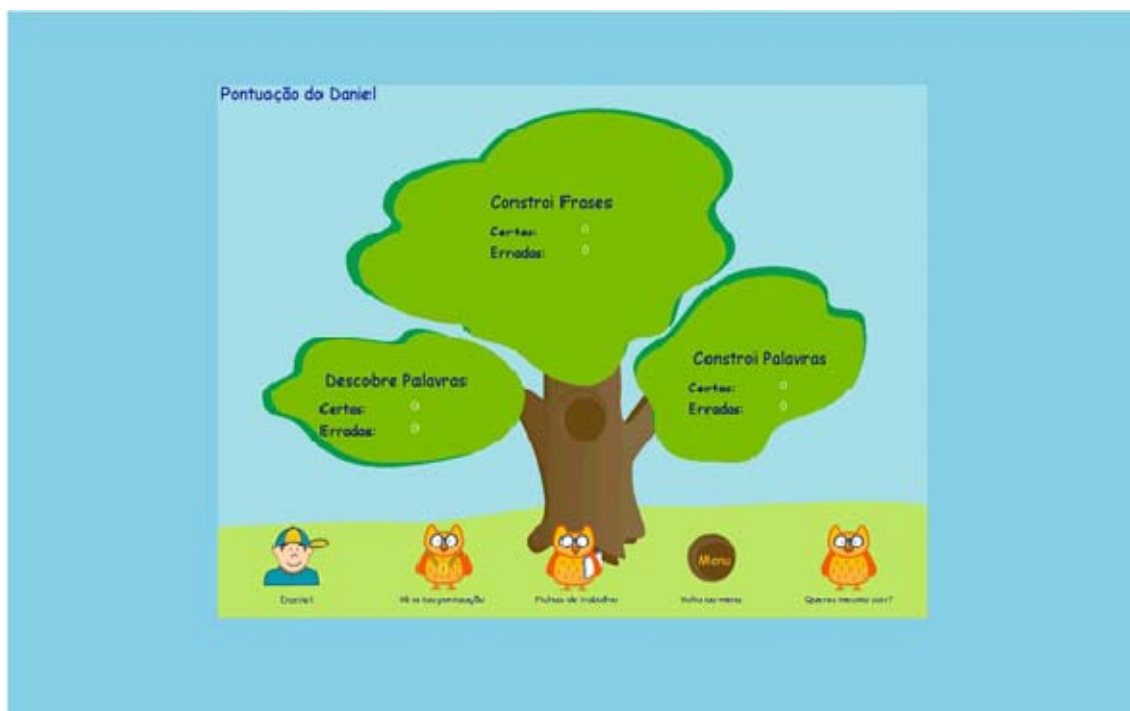


Imagem 4.13 Ecrã da Pontuação do Aluno no Software “Palavras e Palavras”

As contagens de cada jogo são iniciadas cada vez que o aluno está a jogar. Desta forma, o aluno poderá ver a pontuação que alcançou nos jogos que fez num determinado momento. Este processo é tratado de acordo com a ilustração 4.6 que representa uma parte do excerto do código referente às classificações das respostas certas e erradas.

```
public int DP_Certas
{
    get {
        return ((Pontos)this.jogos[this.jogos.Count-1]).DP_Certas;
    }
    set { ((Pontos)this.jogos[this.jogos.Count-1]).DP_Certas = value; }
}
public int DP_Erradas
{
    get {
        return ((Pontos)this.jogos[this.jogos.Count-1]).DP_Erradas;
    }
    set { ((Pontos)this.jogos[this.jogos.Count-1]).DP_Erradas = value; }
}
```

Ilustração 4.6 Excerto do código de contagem de respostas certas e erradas para o Jogo DP do Software “Palavras e Palavras”

No entanto, as pontuações anteriores são armazenadas na aplicação e são tratadas na versão do professor.

4.3.6 Desenvolvimento dos Protótipos de Alta-Fidelidade, para a versão do professor

O *software* “Palavras e Palavras” na versão do professor funciona como gestor de conteúdos do *software* “Palavras e Palavras” da versão para o aluno. Assim, consideramos que este *software* é composto por duas aplicações. É nesta aplicação que se administra os conteúdos lecionados em contexto de sala de aula. Posteriormente, esses conteúdos são disponibilizados na versão do aluno. Desta forma, o *software* tem sempre informação diferente, já que a informação deixa de ser estática para ser dinâmica e adaptável ao aluno. Esta premissa permite que o *software* seja adaptável ao aluno, o que está direcionada para o nosso modelo SIOE.

Na versão do professor o ícone que representa o gestor de conteúdos é um mocho com uma gravata (ver imagem 4.14).



Imagem 4.14 Mocho gestor de conteúdos do *Software* “Palavras e Palavras”

O Jogo Descobre Palavras (DP)

Este ecrã apresenta ao professor as ferramentas necessárias para inserir os dados no jogo DP. Neste ecrã, são mostradas seis janelas, acompanhadas de seis caixas de texto e seis botões para alterar a cor de cada palavra introduzida pelo professor, de acordo com a figura 4.16.

Para introduzir as imagens, basta “clique” em cada janela; ao executar esta ação surge uma caixa de diálogo que permite ao professor escolher uma imagem que tenha no seu computador. Esta imagem é aquela que o aluno irá fazer correspondência com uma outra imagem que seja igual.

Após introduzir os dados nos locais descritos, basta “clique” no botão para salvar as alterações, de acordo com a imagem 4.15.

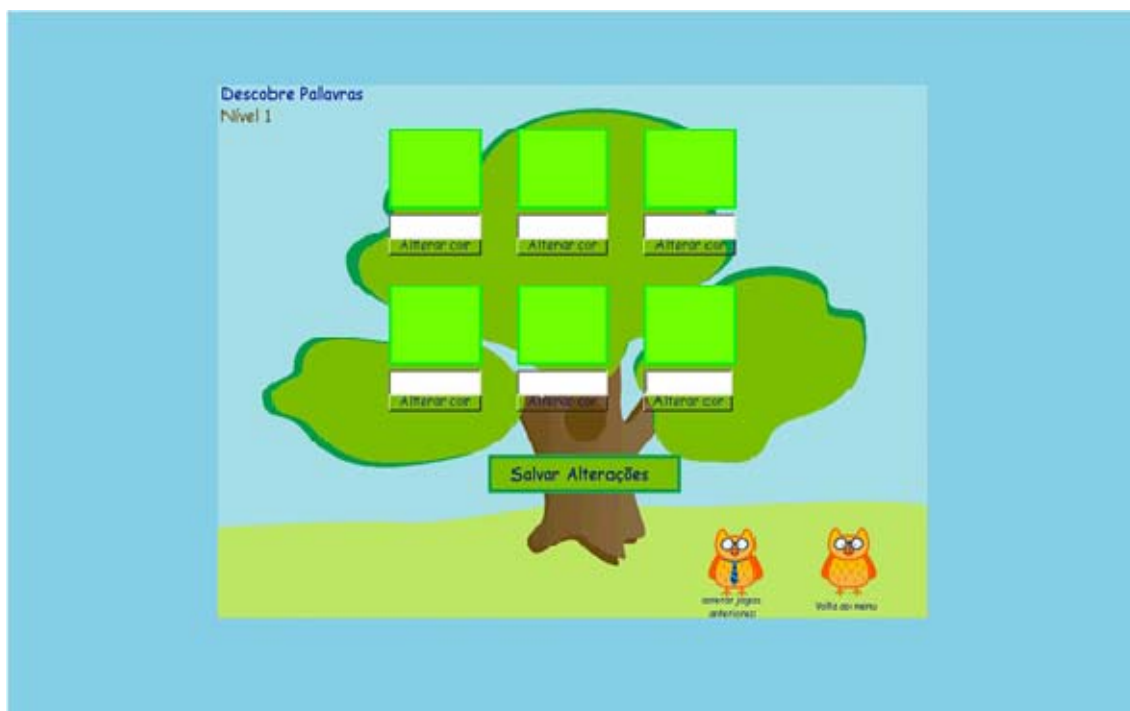


Imagem 4.15 Ecrã da Inserção dos dados do Jogo DP do Software “Palavras e Palavras”

No entanto, uma vez que este jogo tem muitas imagens, houve a necessidade de colocar algumas restrições para não permitir que todas as imagens sejam aceites. Como tal, as imagens não devem ter mais de 60kbytes. Para além desta situação tivemos de ter atenção com o tamanho (altura e largura) da imagem introduzida, isto é, tivemos que reduzir a imagem a um tamanho máximo caso esta o exceda, para que a imagem cumpra o tamanho da janela que o jogo apresenta, de acordo com o código na ilustração 4.7.

Os jogos estão organizados através da primeira letra de cada palavra. Como é que a aplicação trata esta questão? Quando o professor insere um novo jogo, a aplicação lê a primeira letra da primeira palavra e pressupõe que todas as outras palavras começam também com essa letra. O jogo apenas pressupõe tal ocorrência, assim dá liberdade ao professor de poder introduzir um jogo da forma que entender.

Como o professor pode inserir vários jogos para os diferentes níveis de dificuldade do DP, há ainda a necessidade de poder alterar os mesmos e até removê-los. Para tal, o professor tem de “clique” no “mocho de gravata”, para que lhe surja a lista de todos os jogos anteriormente

adicionados, atendendo à letra inicial e ao nível a que estes correspondem. Para remover, basta seleccionar o jogo e pressionar a tecla “DELETE” do seu teclado.

No caso de pretender alterar os dados de um determinado jogo, basta seleccioná-lo e “clicar” o botão “Continuar”. Após essa ação surge o ecrã igual ao ecrã anterior com as seis janelas. A diferença é que este ecrã já tem os campos preenchidos com os dados que foram introduzidos.

```
private void picJanela_Click(object sender, System.EventArgs e)
{
    System.Windows.Forms.PictureBox pictureBox = (System.Windows.Forms.PictureBox)sender;
    this.openFileDialog1.Filter = "Image Files | *.jpg;*.gif;*.png;*.bmp;|All Files | *.*";
    this.openFileDialog1.ShowDialog();
    if (!this.openFileDialog1.FileName.Equals("")) {
        FileInfo fileInfo = new FileInfo(this.openFileDialog1.FileName.ToString());
        if (fileInfo.Length <= 61440)
        {
            Bitmap img = new Bitmap(this.openFileDialog1.FileName.ToString());
            if (img.Width >= pictureBox.Width || img.Height >= pictureBox.Height)
            {
                Image tempImage = (Image)img;
                Size fitImageSize = comum.getScaledImageDimensions(img.Width, img.Height, pictureBox.Width-10,
                pictureBox.Height-10);
                Bitmap imgOutput = new Bitmap(tempImage, fitImageSize.Width, fitImageSize.Height);
                pictureBox.Image = imgOutput;
            }
            else {
                pictureBox.Image = img;
            }
            pictureBox.SizeMode = System.Windows.Forms.PictureBoxSizeMode.CenterImage;
        }
        else {
            MyMsgBox.MyMsgBox msg = new MyMsgBox.MyMsgBox();
            msg.Caption = "Erro";
            msg.Font = new System.Drawing.Font("Comic Sans MS", 9.75F);
            msg.Text = "Escolha uma imagem que ocupe menos espaço em disco, ou reduza o número de cores da
            imagem.\nLimite de tamanho máximo: 60kb!";
            msg.Icon = MyMsgBox.MyMsgBoxIcon.Exclamation;
            msg.Show();
        }
    }
}
```

Ilustração 4.7 Excerto do código referente à inserção de imagens no Jogo DP do *Software* “Palavras e Palavras”

O Jogo Constrói Palavras

O funcionamento deste ecrã é idêntico ao descrito no ecrã do jogo CP e por sua vez análogo ao ecrã seguinte do jogo CF. O “mocho de gravata” tem as mesmas funções, a identificação do jogo e do nível continua a surgir no canto superior esquerdo, a quantidade de caixas de texto e de imagens é menor do que no jogo DP.

O jogo apresenta 3 níveis (ver figura 4.17) e em qualquer um deles, o jogo apresenta quatro caixas de texto, que consoante o nível selecionado são de cores diferentes ou não. No primeiro nível, são acompanhadas por uma caixa de imagem que diz “inserir uma imagem”. Nos restantes níveis essa situação não se verifica, de acordo com a figura 4.17. Nas caixas de texto o professor não pode escolher as cores, tem de utilizar as que estão estipuladas para este jogo. Sendo elas as cores primárias: verde, vermelho, azul e amarelo, como se pode constatar na imagem 4.16.

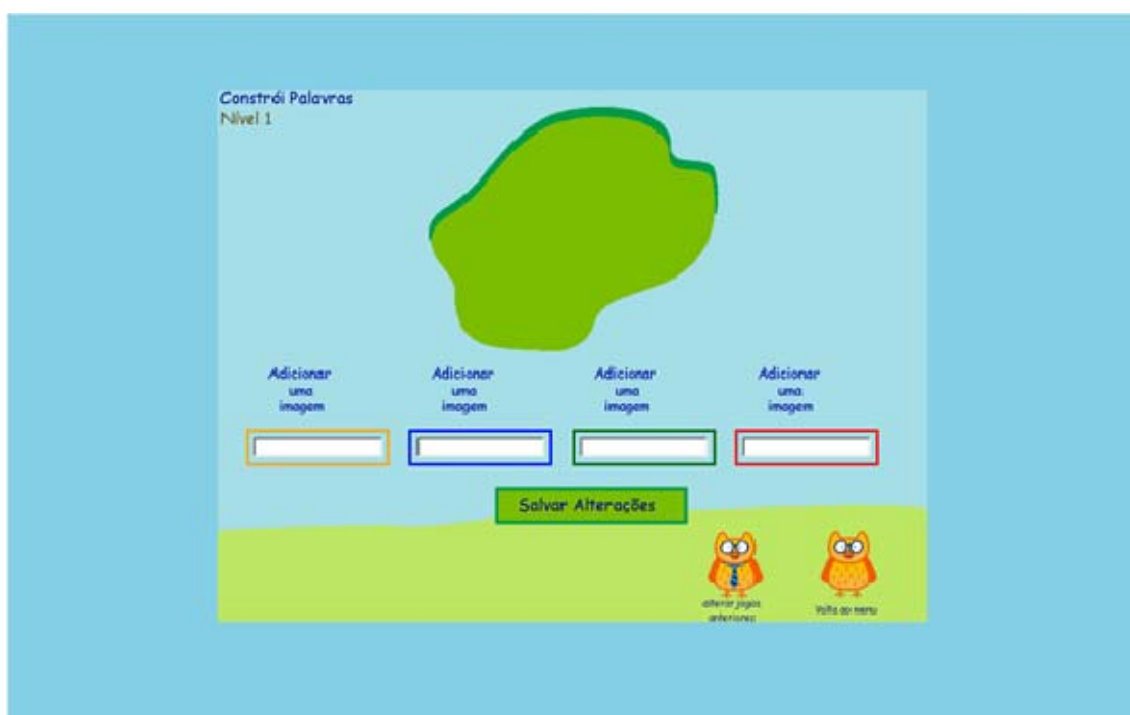


Imagem 4.16 Ecrã de inserção de dados para o jogo CF do *Software “Palavras e Palavras”*

Em relação às opções de inserção, alteração e remoção de conteúdos neste jogo funciona da mesma forma que o jogo DP. Apresentamos uma parte do código deste jogo, respeitante à inserção de imagens, ver ilustração 4.8.

```

private void Pic_Click(object sender, System.EventArgs e) {
    System.Windows.Forms.PictureBox pic = (System.Windows.Forms.PictureBox)sender;

    this.openFileDialog1.Filter = "Image Files|.jpg|.gif|.png|.bmp;|All Files|*.*";
    this.openFileDialog1.ShowDialog();
    if (!this.openFileDialog1.FileName.Equals(""))
    {
        FileInfo finfo = new FileInfo(this.openFileDialog1.FileName.ToString());
        if(finfo.Length <= 61440)
        {
            pic.Image = new Bitmap(this.openFileDialog1.FileName.ToString());
            pic.SizeMode = System.Windows.Forms.PictureBoxSizeMode.CenterImage;

        }
        else
        {
            MyMsgBox.MyMsgBox msg = new MyMsgBox.MyMsgBox();
            msg.Caption = "Erro";
            msg.Font = new System.Drawing.Font("Comic Sans MS", 9.75F)
            msg.Text = "Escolha uma imagem que ocupe menos espaço em disco, ou
            reduza o número de cores da imagem.\n Limite: de tamanho máximo 60kb!";
            msg.Icon = MyMsgBox.MyMsgBoxIcon.Exclamation;
            msg.Show();

        }
    }
}
}

```

Ilustração 4.8 Código respeitante à inserção no Jogo CP do *Software* "Palavras e Palavras"

O Jogo Constrói Frases

O jogo "Constrói Frases" disponibiliza ao professor 2 caixas de texto para introdução de duas frases.

O funcionamento deste ecrã funciona da mesma forma dos ecrãs descritos nos pontos anteriores deste relatório, fazendo uso de alguns métodos comuns a estes três jogos. De acordo com os níveis deste jogo, as frases a inserir podem ter cores diferentes ou não. No entanto, o professor/educador não pode escolher as cores, estas estão pré-definidas. No nível 1 as frases são de cor vermelha e azul (ver imagem 4.17 e a figura 4.18), e para o nível 2 ambas as frases utilizam a cor preta.

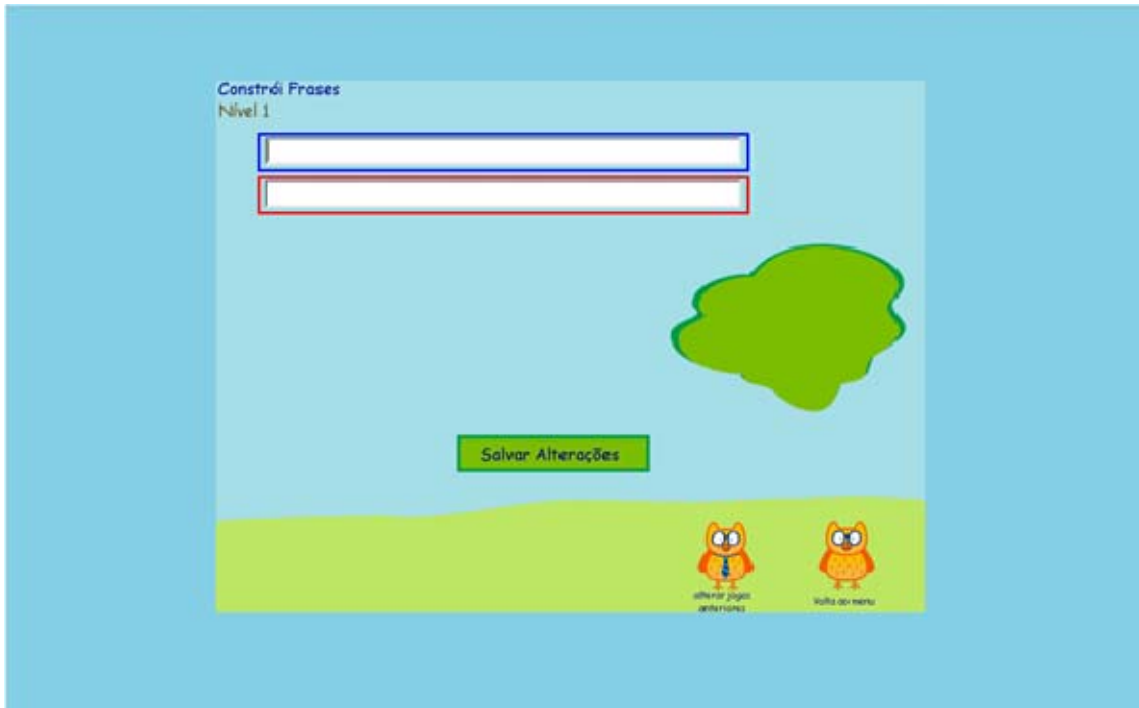


Imagem 4.17 Ecrã do jogo CF do *Software* “Palavras e Palavras”, versão do Professor

Fichas de trabalho

Através deste ecrã o professor insere as fichas de trabalho. Em primeiro lugar tem de inserir os vários temas relacionadas com a língua portuguesa, para tal, tem de existir uma zona reservada para essa ação. Assim como outra zona para inserir as fichas de trabalho. Portanto, o ecrã apresenta 2 listas distintas, como se pode observar na imagem 4.18.

Após inserir o tema o professor pode associar várias fichas de trabalho e/ou testes de avaliação a cada tema. Estes documentos devem ser anteriormente escritos num processador de texto, como por exemplo o *OpenOffice*, ou o *Microsoft Word*. Assim, a aplicação fica preparada para a aprendizagem da leitura e da escrita da língua portuguesa, através dos jogos DP, CP e CF e também através das fichas de trabalho que os alunos posteriormente podem imprimir para responder às questões propostas pelo docente.

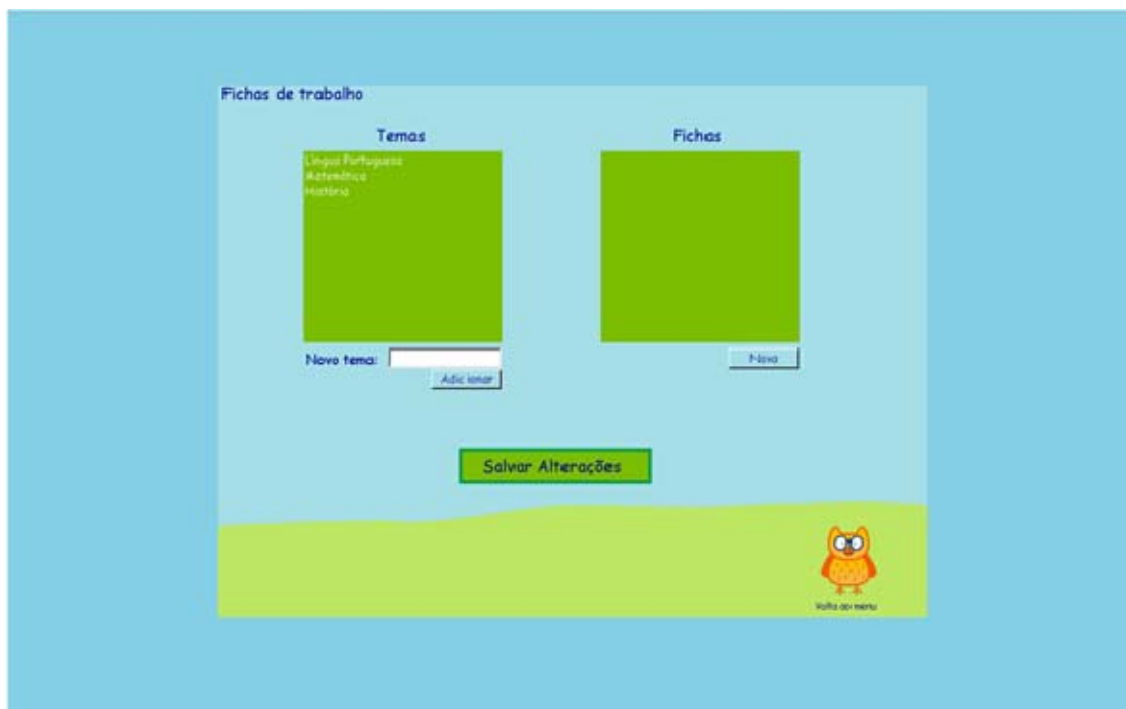


Imagem 4.18 Ecrã da Inserção das Fichas de Trabalho do *Software* “Palavras e Palavras” versão do professor (Rodrigues & Grilo, *CienciaPt - A Educação, Ciência, Tecnologia e Inovação em Portugal - Revista*, 2007)

Pontuação dos alunos

O professor responsável pela gestão de conteúdos pode ver a pontuação de todos os alunos com perfil inserido na aplicação destinada ao aluno. Este ecrã é extremamente importante, uma vez que, para além de permitir a visualização das pontuações de qualquer aluno e não apenas do selecionado anteriormente, permite ainda ver e analisar sob a forma de um gráfico evolutivo as pontuações que o aluno tem obtido, desde o seu primeiro jogo até ao último. Devemos salientar que o professor pode ver em simultâneo as linhas de resultados obtidos nos 3 jogos.

O professor também tem a possibilidade de verificar a pontuação e a evolução do aluno, para tal é necessário selecionar um intervalo de datas ou uma data específica. Portanto o ecrã apresenta 2 calendários que permitem ao utilizador selecionar as datas (ver imagem 4.19). Uma vez selecionado o instante temporal, o gráfico apresenta os resultados conseguidos pelo aluno (anteriormente selecionado) no espaço de tempo escolhido.

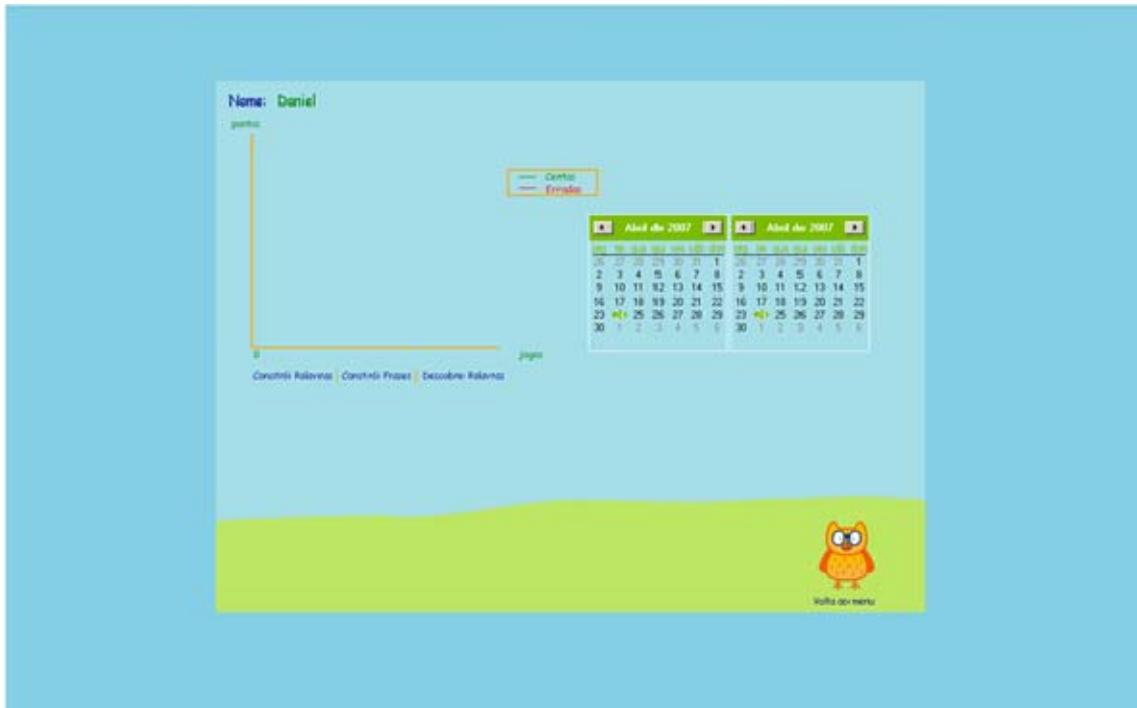


Imagem 4.19 Ecrã da Pontuação do Software “Palavras e Palavras” versão do professor

4.4 Tecnologias Utilizadas no Desenvolvimento do Software “Palavras e Palavras”

De acordo com modelo IAOE e o método MIAOE, estamos na zona do processo UID – Definição dos objetos, que tem como base eleger as tecnologias a utilizar, como por exemplo a linguagem de programação a ser escolhida, como se pode verificar na figura 3.5 e na tabela 3.4.

É de salientar que perante o modelo em cascata estamos na zona de “*Implementation and unit testing*” e na fase de implementação do modelo Híbrido.

No desenvolvimento do Software “Palavras e Palavras” foram utilizadas várias tecnologias para desenvolver as diferentes fases do projeto. A lista seguinte descreve as tecnologias usadas:

Plataforma. NET (ex: *Visual Studio. Net 2003*);

Add-Ins para o *Visual Studio. NET 2003* (ex: *Initialize Component Optimizer*);

Animações gráficas (ex: *Macromedia Flash MX 2004*);

Animações auditivas (ex: Gravador de áudio do Windows XP);

Editores de imagem (ex: *Paint Shop Pro 9*, *Corel Draw 10*, *Paint*);

Setup de instalação (ex: *Inno Setup v5.1.6*);

Linguagens de programação utilizadas: *C Sharp (C#)*, *Flash*, *JavaScript*.

O software “Palavras e Palavras” está desenvolvido sobre a linguagem de programação *C Sharp (C#)*, o símbolo # refere-se ao sinal musical (sustenido). A escolha desta linguagem de programação deve-se a vários fatores, entre os quais: a linguagem funcionar sobre a plataforma .NET o que a torna uma linguagem bastante poderosa, graças as numerosas bibliotecas que esta plataforma possui. Destina-se tanto ao desenvolvimento de aplicações simples como ao desenvolvimento de aplicações complexas para ambientes Windows.

A aplicação compreende múltiplos formulários com dimensões fixas de 800x600, para permitir que o Software “Palavras e Palavras” consiga ser visualizado em ecrãs que suportem resoluções pequenas. Como a resolução não é sempre igual em todos os computadores (pode ser maior), foi necessário acrescentar um formulário em que o tamanho é dinâmico consoante a resolução do computador em que está a correr o *software*. Como tal, o espaço que não for completado pelos formulários de 800x600, será ocupado por este.

Cada formulário em C# é exibido na barra de tarefas do Windows, por isso foi necessário deixar somente o ecrã inicial visível, logo, tivemos de esconder os outros formulários através da opção *ShowInTaskbar*. Como tal, o *software* só ocupa um espaço na barra de tarefas do Windows.

A aplicação utiliza muitas funções nos diferentes ecrãs, nesse sentido, houve uma grande preocupação em dividir as classes e funções comuns em diferentes soluções que posteriormente são associadas a outras soluções através de referências.

Hoje, existe uma oferta de vastíssimas soluções para instalar aplicações. Estas soluções querem-se o mais simples e abrangente possível para que qualquer utilizador consiga seguir todo o processo de instalação sem dificuldade, de forma que este possa escolher opções relacionadas com a instalação, nomeadamente a diretoria onde ficarão armazenados os

ficheiros, os atalhos criados, entre outros. A escolha da solução para a instalação recaiu sobre o *Inno setup* que é um projeto livre e *Open Source*.

O *setup* desta aplicação tem integrado a *framework 1.1* da Microsoft uma vez que esta é precisa para a execução do projeto em causa. Optamos pela escolha desta versão porque poder ser instalada em várias versões do Windows, visto que, as escolas em que foi testado o *software* têm computadores em Windows 98, Windows ME e Windows XP. O *setup* tem ainda integrado o *software* para instalação do *Flash Player MX* caso o computador do professor (utilizador) não tenha tal *software* instalado.

Quando é executado o *setup*, o utilizador tem a possibilidade de escolher duas linguagens (português e inglês) que serão utilizadas para comunicar com o mesmo no decorrer do processo de instalação como se constata na figura 4.29.

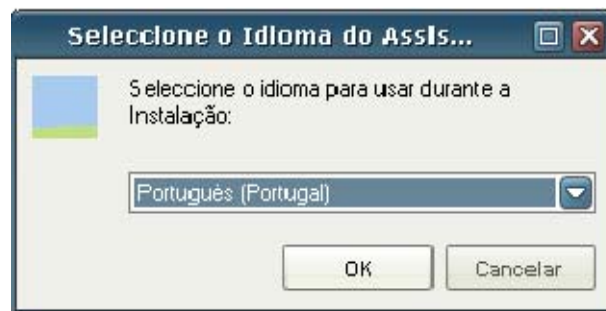


Figura 4.29 Execução do *setup* da nossa aplicação

A documentação de apoio a este *software* é disponibilizada na diretoria onde este se instala, está organizada numa outra diretoria denominada “Docs”. O formato desta documentação é HTML, visto que, oferece ao professor uma forma interativa de aceder à informação que pretende, o que lhe permite selecionar os temas e visualizar a informação de forma organizada, simples e eficaz.

Conclusões

Este capítulo foca todos os aspetos que tivemos de ter conta para a fase de implementação. Foi extremamente trabalhoso porque todas as nossas ações tinham de estar de acordo com o nosso modelo IAOE e com o método proposto MIAOE. Tivemos que conjugar todas as metodologias adotadas (referidas no capítulo 3) de uma forma interativa e dinâmica, nunca podendo esquecer que a população de estudo é muito específica com característica muito próprias e nem sempre delineadas.

Antes de mais o papel da memória e o papel do córtex visual foram elementos fundamentais na criação da interface gráfica. Visto termos tido em conta os aspetos cognitivos no *design* dos ícones, pois eles estão concebidos de modo a que os alunos e os professores facilmente percebem o seu significado e os consigam distinguir uns dos outros.

A interface do *Software* “Palavras e Palavras” foi criada atendendo às necessidades dos alunos e dos professores/educadores. Assim, está concebida para ser utilizada de uma forma intuitiva, simples e eficaz. Permite que o aluno e o professor/educador navegue pela aplicação por meio de acesso a botões ou sob a forma de opções do menu. Consideramos que esta interface é fácil de usar e de compreender, sendo conjuntamente motivadora o que implica no aluno um maior envolvimento cognitivo na exploração dos conteúdos didáticos disponibilizados pelo professor/educador. Esta interface assume uma natureza multimodal.

Não consideramos a interface somente a nível de *design* gráfico, também sintetiza o ambiente de aprendizagem em contexto de sala de aula. Como tal, reflete as características dos alunos, situações de aprendizagem e de avaliação.

A interatividade existente entre o “mocho” e o aluno, entre o “mocho e o professor/educador a nível de concretização de uma atividade e a nível de navegação orientam o seu desempenho a nível cognitivo e de destreza. Como tal, evitamos sobrecarregar os ecrãs

com demasiada informação. Optamos por colocar a informação mais relevante nas zonas mais destacadas, ou seja na área do corpo e nos quadrantes próximos desta área.

Em termos de usabilidade e interatividade o *Software* “Palavras e Palavras” apresenta uma interface ordenada, limpa e clara. A informação apresenta uma linguagem clara e simples, as instruções são transparentes, o que permite ao aluno e ao professor/educador saber que atitude tomar em qualquer situação com que se depare.

A utilização de imagens e de determinadas cores está de acordo com as teorias de aprendizagem e mais uma vez a preponderância do córtex visual nestas situações.

A utilização de “*menu*” no *Software* “Palavras e Palavras” constitui uma técnica de interação que permite ao aluno e ao professor/educador (aos utilizadores) utilizar uma lista de opções, que lhe proporciona indicações visuais de forma ordenada, tendo em conta que a capacidade dos alunos e dos professores (do ser humano) de recordar a informação é inferior à sua capacidade de reconhecer a informação a partir de indicações visuais. Visto que, a utilização de sinais visuais complementam e potenciam a perceção de novos conceitos, de acordo com a psicologia cognitiva cada conceito assimilado é transformado em saber real que serve de base para a aquisição de novos conceitos e que estes dependem bastante mais do que se já se aprendeu do que da complexidade do tema em si.

O trabalho de “*campo*” realizado permitiu seguir a estrutura por nós definida em relação às etapas de desenvolvimento de *software* de acordo com os modelos adaptados.

Testes e Resultados

“Os utilizadores têm um potencial ilimitado para interpretar de forma inesperada os elementos de uma interface e para concretizar as suas tarefas de um modo diferente do imaginado” (Nielsen, 1993)

A fase de teste no ciclo de vida do *software* é um ponto crucial para o sucesso do mesmo. Tem como objetivo encontrar falhas nos sistemas para que possam ser corrigidos atempadamente. Os testes devem ser efetuados durante o desenvolvimento do *software*, *“para que quanto mais cedo for descoberta a falha menor é o custo decorrente da sua correção”* (Myers, 2004). Podemos mesmo dizer que os testes estão relacionados com a qualidade. Como evoca Craig e Jaskiel *“Testing is a concurrent lifecycle process of engineering, using and maintaining testware in order to measure and improve the quality of the software being tested”* (Craig & Jaskiel, 2002) .

Como tal, nesta fase os modelos de qualidade (McCall e Boehm), As Oito Regras de Ouro de Shneiderman, os 7 Princípios de Norman, Os Oito Critérios de Scapin & Bastien e os Quadros (II, III, IV) de Costa, adotados por nós, desempenharam um papel extremamente crucial, assim como as normas ISO (ver capítulos 2 e 3). Associados à conjugação do Processo UID e do Modelo HCI, especificamente na zona de intersecção do ambiente das tarefas com o ambiente da máquina, foi-nos possível realizar os testes ao *Software* *“Palavras e Palavras”* com clareza e objetividade dirigidos para o nosso objetivo final a *“interatividade e adaptabilidade para como o utilizador”*; obviamente que a usabilidade está intrínseca nestes objetivos.

Logo, foi necessário incorporar os testes praticamente desde a sua fase inicial, o que aponta para o modelo em V (ver capítulo 2). Esta versão sofreu bastantes alterações oferecendo mais e melhores ferramentas ao aluno e ao professor. Mais uma vez, estamos perante o modelo em

Cascata e do modelo de Prototipagem. O que implicou voltar a determinados processos UID do MIAOE, essencialmente os processos “UID – Objetos Visuais de Desenho” e “UID – Interfaces Específicos para o Utilizador”. Também tivemos em conta o modelo de MacCall e o modelo Boehm, de acordo com o capítulo 2.

Foram realizados testes de sistema e testes de aceitação. Os testes de sistema tiveram de verificar se a aplicação em si interage corretamente com o aluno e com o professor, o que inclui: inserção, alteração e eliminação de alunos; o funcionamento dos jogos (arrastar palavras ou letras, clicks, etc); a inserção dos conteúdos dos jogos, etc. Obviamente que estão incluídos os testes à interface, à usabilidade, à segurança, à Norma ISO/IEC 9126. Os testes de aceitação verificaram se a aplicação atendia aos objetivos e exigências propostas por nós (ver figura 5.1).

Porém, devemos referir que os aspetos da usabilidade relacionados com o conteúdo e com as funcionalidade do *Software* “Palavras e Palavras” têm preferência em relação aos aspetos estéticos que nada acrescentam em relação ao conteúdo.

Realçamos que o *Software* “Palavras e Palavras” possui características específicas (dependem do tipo de aluno) que foram consideradas no momento de realização dos testes. Desta forma, os professores tiveram um papel ativo na participação deste estudo, uma vez que foi efetuada uma fase de estudo de viabilidade de um *software*, sendo a participação dos professores a chave para esse estudo. Foram definidos objetivos bem concretos do que teria de se implementar; uma fase de desenho onde todos participantes tiveram em conta algumas das regras mais importantes de usabilidade e uma fase de protótipo de alta-fidelidade onde foi possível verificar a receptividade por parte dos alunos e dos professores.

Referimos que em determinado momento a ficha de avaliação de *software* educacional - parte III de Rodrigues, foi melhorada, visto termos constatado que os resultados obtidos não abrangiam na totalidade os objetivos propostos para o nosso projeto. As melhorias efetuadas na ficha permitiram termos a noção exata da qualidade do nosso *software*. Denominamos esta parte da ficha como: Nova Taxonomia de Avaliação de *Software* Educacional/Educativo. No entanto, apresentamos ambos os testes e os resultados dos mesmos da parte III da ficha de avaliação.

Salientamos que os testes abrangeram uma avaliação quantitativa e qualitativa. Os dados gerados qualitativamente são de natureza descritiva. Os dados gerados quantitativamente são

de natureza numérica, numa escala de 0-5 valores. Para a obtenção dos dados foi utilizada a técnica de observação direta e indireta. A observação direta permitiu capturarmos os comportamentos por si só e no momento em que ocorreram, portanto não houve a necessidade da intermediação de um documento ou testemunho, é uma observação do tipo etnológico na qual estivemos sempre presentes. Enquanto na observação indireta foi utilizado o inquérito por questionário, de forma estruturada e diretiva em que o elemento básico foram as questões relacionadas com modelo MIAOE e o método MIAOE. Como se pode verificar através da figura 5.1.



Figura 5.1 Metodologia dos testes de aceitação do Software “Palavras e Palavras”

A avaliação do Software “Palavras e Palavras” foi realizado nas 3 escolas abrangidas com este estudo e indicadas no capítulo 1.

5.1 Testes e Resultados Obtidos a Nível Quantitativo

“A provação vem, não só para testar o nosso valor, mas para o aumentar; o carvalho não é apenas testado, mas enrijecido pelas tempestades.” (Cowman, 1999) .

Nesta fase de teste o software “Palavras e Palavras” foi submetido a 2 avaliações, que tiveram como objetivo questionar a aplicação através de casos de teste e de observações para avaliar as respostas obtidas de acordo com a ficha de avaliação parte III de Rodrigues (Rodrigues E. , A Contribution for the Evaluation of Educational Interactive Software, 2002) ver apêndice A e ver Apêndice B.

Como a ficha de avaliação (parte III) de Rodrigues apresenta características qualitativas e quantitativas, avaliámo-las em separado. Na avaliação quantitativa foi calculada a média dos resultados obtidos por item. Enquanto na avaliação qualitativa analisámos os aspetos pedagógicos através da quantidade de respostas “sim” e “não”.

Na aplicação relacionada com o aluno, aspetos como as cores o tamanho e o tipo de letra, o arrastar das palavras, o clicar nas imagens foram analisados e testados pelos mesmos. Estes testes foram efetuados algumas vezes através da observação direta outras vezes através da ficha de avaliação de Rodrigues (Rodrigues E. , A Contribution for the Evaluation of Educational Interactive Software, 2002).

Aos professores coube testar o *software* a todos os níveis, ou seja, testaram a versão alunos e a versão professor.

5.1.1 Testes e Resultados Obtidos Através da Ficha de Avaliação de Software Educacional Parte III de Rodrigues

A avaliação Ficha de Avaliação parte III de Rodrigues (Rodrigues E. , A Contribution for the Evaluation of Educational Interactive Software, 2002) foi testada pelas professoras e pelos alunos no decorrer do projeto. Houve partes da ficha III que foram a avaliar mais do que uma vez, tais como, a interface, a interatividade, ferramentas de exploração e aspetos pedagógicos. A necessidade de efetuar estas avaliações deve-se à metodologia ASD (por nós adotada, ver capítulo 3) que tem como princípio elementar as pessoas.

Em relação Ficha de Avaliação de Software Educacional de Rodrigues (Rodrigues E. , A Contribution for the Evaluation of Educational Interactive Software, 2002) e em termos quantitativos, analisámos os resultados em termos de “Ajuda”, de “Interatividade”- a nível do

som (discurso oral); da imagem (estática e animada); da participação e controlo do aluno; aspetos pedagógicos; das ferramentas de exploração.

Resultados da Interatividade

O gráfico 4.1 apresenta os resultados do ponto “Interatividade” com os seus itens, a nível quantitativo. É de referir que o item 2- “Estrutura da comunicação”, apresentou uma resposta direta, como tal não aparece no gráfico, a resposta obtida é a seguinte: “comunicação ativa e experimentação”. Com este tipo de resposta, ficámos sem saber valorizar a estrutura da comunicação. A situação foi resolvida com alterações nos critérios de avaliação, como se verá no subponto – Resultados da versão 2 da ficha de avaliação.

Em relação à interatividade as tabelas 5.1, 5.2 e a tabela 5.3 apresentam as cotações das 5 professoras, assim como a média em relação a cada um dos itens: feedback, grau de participação do aluno e o grau de controlo do aluno. É de referir que a pontuação está compreendida entre 0-5 valores, sendo 0 não existente e 5 muito bom, de acordo com o apêndice A.

1. Feedback	
Professora A	5
Professora B	5
Professora C	5
Professora Apoio D	5
Professora Apoio E	5
Média	5

Tabela 5.1 Resultado do Feedback no *Software* “Palavras e Palavras”

3. Grau de Participação do Aluno	
Professora A	5
Professora B	5
Professora C	5
Professora Apoio D	4
Professora Apoio E	5
Média	4,8

Tabela 5.2 Resultado do Grau de participação do aluno no *Software* “Palavras e Palavras”

4.Grau de Controlo do Aluno	
Professora A	4
Professora B	5
Professora C	4
Professora Apoio D	4
Professora Apoio E	5
Média	4,4

Tabela 5.3 Resultado do Grau de controlo do aluno no *Software* “Palavras e Palavras”

Em relação ao item “ feedback” as respostas atingiram o nível máximo que é de 5 (muito bom), no entanto, não é um resultado esclarecedor, porque não nos indica a área específica de feedback que estamos a observar ou a analisar. Como tal, houve que alterar este item para obter resultados mais específicos que validassem os objetivos do projeto. Como se pode ver no ponto resultados da interatividade após as alterações à ficha de avaliação de Rodrigues.

O gráfico 5.1 apresenta a média dos 3 itens que compõem a interatividade em termos quantitativos.

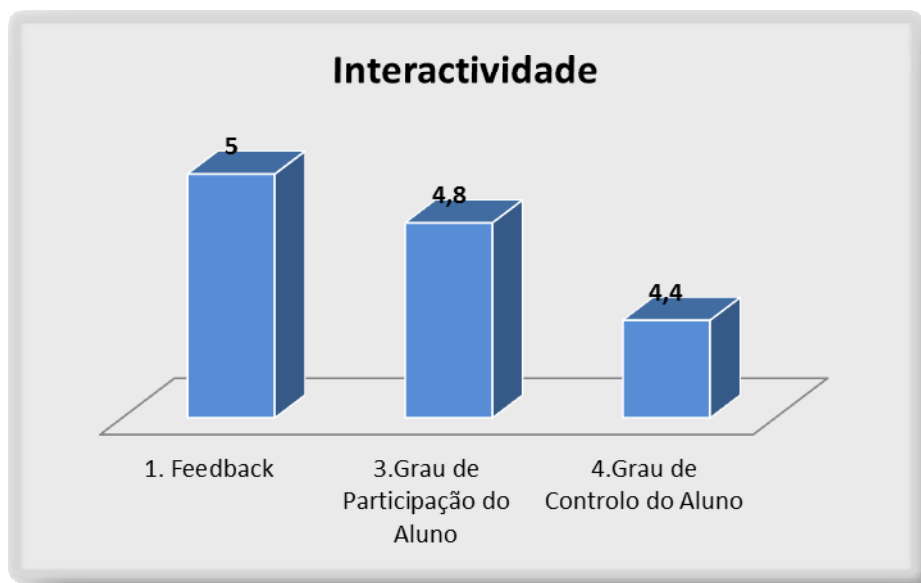


Gráfico 5.1 Média dos Resultados da Interatividade do Software “Palavras e Palavras”

Resultados de Materiais de Apoio

Os materiais de apoio a que nos referidos estão inseridos no tipo de “Ajuda” no *software* “Palavras e Palavras” inclui os itens mensuráveis – “Clareza de instruções” e “quantidade de informação”. As tabelas 5.4 e 5.5 apresentam os resultados obtidos individualmente pelas avaliadoras e a média dos mesmos.

Clareza de Instruções	
Professora A	3
Professora B	2
Professora C	3
Professora Apoio D	2
Professora Apoio E	3
Média	2,6

Tabela 5.4 Resultados Materiais de Apoio na Clareza de Instruções no *Software* “Palavras e Palavras”

Quantidade de Informação	
Professora A	2
Professora B	2
Professora C	2
Professora Apoio D	2
Professora Apoio E	2
Média	2

Tabela 5.5 Resultados Materiais de Apoio na Quantidade de Informação no *Software* “Palavras e Palavras”

O gráfico 5.2 apresenta a média obtida através dos resultados das tabelas 5.4 e 5.5.

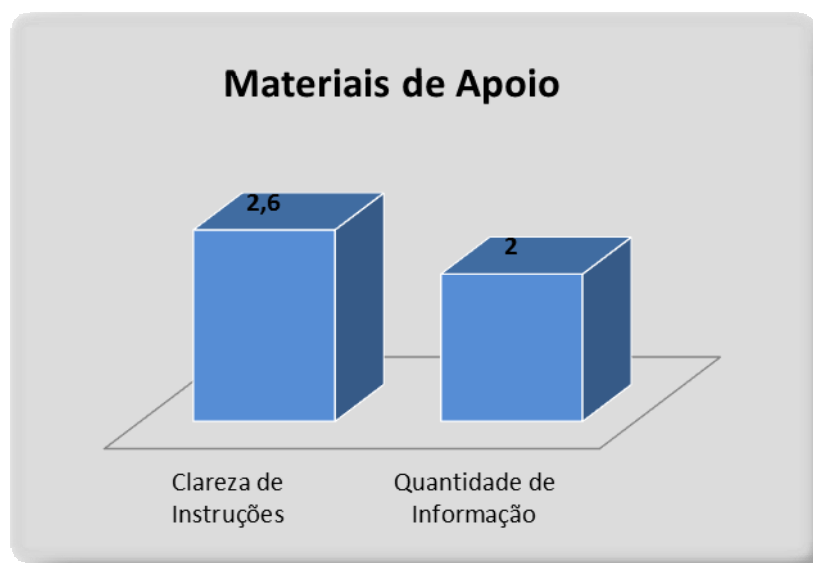


Gráfico 5.2 Média dos itens dos Materiais de Apoio no Software “Palavras e Palavras”

As respostas ao item 3 são de nível qualitativo, como tal ver o resultado no ponto 5.2.2.

Aspetos Pedagógicos

As respostas aos itens 1 e 2 deste ponto não são avaliadas, pois dependem das características do aluno e dos seus objetivos de aprendizagem.

Os itens 6, 8, e 9 da ficha de avaliação de Rodrigues só permitem avaliar estas respostas a nível qualitativo (ver ponto 5.2.2), enquanto os restantes itens deste ponto permitem uma avaliação quantitativa.

As tabelas 5.6, 5.7, 5.8 e 5.9 mostram os resultados obtidos na avaliação do *Software* “Palavras e Palavras” a nível quantitativo.

Visuais	
Professora A	3
Professora B	3
Professora C	3
Professora Apoio D	3
Professora Apoio E	3
Média	3

Tabela 5.6 Resultado da avaliação dos aspetos visuais em relação aos aspetos pedagógicos do *Software* “Palavras e Palavras”

Interactividade e motivação em relação ao aluno	
Professora A	5
Professora B	5
Professora C	4
Professora Apoio D	5
Professora Apoio E	5
Média	4,8

Tabela 5.7 Resultado da avaliação da interatividade e motivação em relação ao aluno nos aspetos pedagógicos do *Software* “Palavras e Palavras”

Estimula a criatividade	
Professora A	3
Professora B	3
Professora C	3
Professora Apoio D	3
Professora Apoio E	2
Média	2,8

Tabela 5.8 Resultado do item estimula a criatividade em relação aos aspetos pedagógicos do *Software* “Palavras e Palavras”

Resposta exige reflexão por parte do aluno	
Professora A	5
Professora B	5
Professora C	5
Professora Apoio D	5
Professora Apoio E	5
Média	5

Tabela 5.9 Resultado do item resposta exige reflexão por parte do aluno em relação aos aspetos pedagógicos do *Software* “Palavras e Palavras”

A totalidade destes resultados é apresentado pela média obtida através do gráfico 5.3, como se pode verificar.

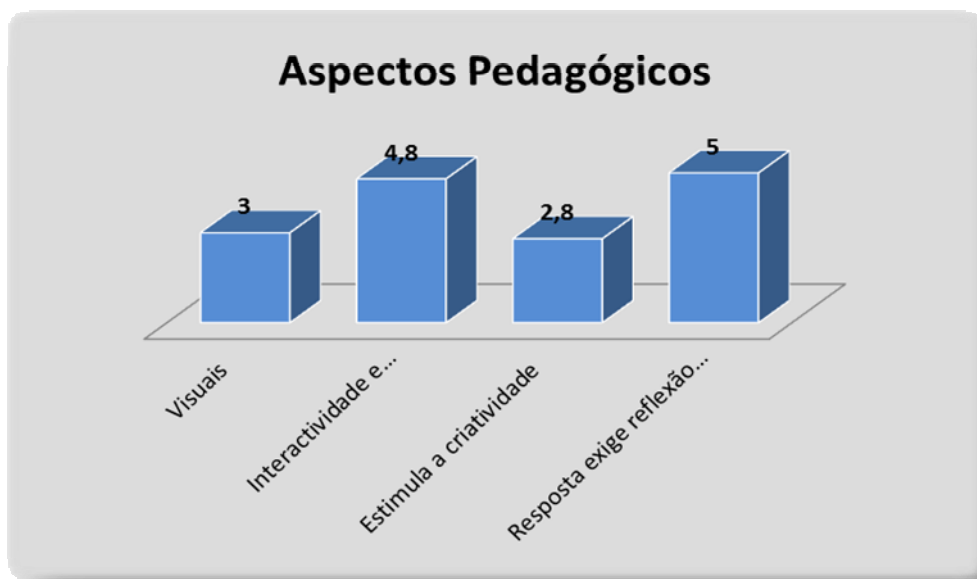


Gráfico 5.3 Média obtida na avaliação dos aspetos pedagógicos do *Software* “Palavras e Palavras”

5.1.2 Resultados Obtidos Através da Nova Taxionomia de Avaliação de Software Educacional (Parte III)

As alterações à ficha de avaliação situam-se entre o ponto de interatividade e o ponto ferramentas de exploração e no ponto referente aos aspetos pedagógicos, de acordo com o apêndice B.

É de realçar que as alterações foram realizadas, somente, nos itens a nível quantitativo.

Apresentamos somente os resultados que correspondem às alterações referidas. Como tal, apresentamos o item 1 relativo à interatividade.

Resultados em relação à Interatividade

As funções do texto (ver tabela 4.11) estão relacionadas com a estrutura da comunicação, seja ela ativa, passiva e experimentação ou criativa. Dos 4 itens existentes nesta função associamos somente 2 dos itens (Função Apelativa e Função Informativa) à interatividade, de acordo com as tabelas 5.10 e 5.11. Os outros 2 itens estão associados à usabilidade. O item usabilidade foi acrescentado na nova Taxonomia de Avaliação de *Software* Educacional.

Salientamos que a pontuação atribuída está compreendida entre 0-5 valores, sendo o 0 o valor mais baixo (que corresponde à inexistência do item), 5 o valor mais alto que corresponde ao muito bom.

2. Funções do Texto	
Função Apelativa	
Professora A	4
Professora B	4
Professora C	4
Professora Apoio D	3
Professora Apoio E	4
Média	3.8

Tabela 5.10 Resultado do item Interatividade na Função do Texto – Função Apelativa no *Software “Palavras e Palavras”*

2. Funções do Texto	
Função Informativa	
Professora A	4
Professora B	4
Professora C	5
Professora Apoio D	5
Professora Apoio E	4
Média	4,4

Tabela 5.11 Resultado do item interatividade na Função do Texto – Função Informativa no *Software “Palavras e Palavras”*

O resultado destes dois itens é traduzido através do gráfico 5.4

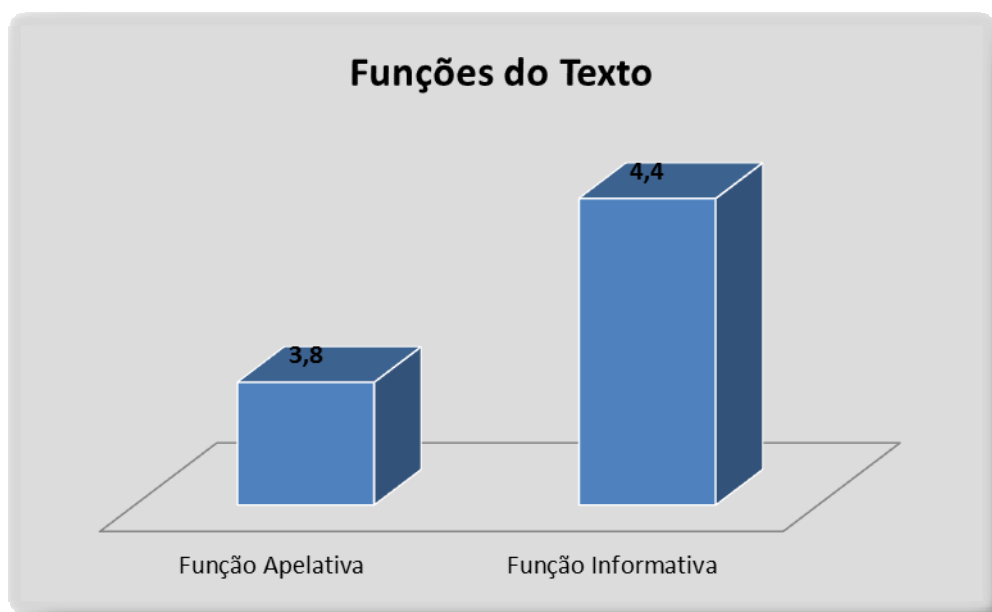


Gráfico 5.4 Resultado das funções do Texto no Software “Palavras e Palavras”

Com este resultado concluímos que a função apelativa está próxima do bom, enquanto a função informativa ultrapassa o bom, mas não chega ao muito bom.

Funções do Som

As funções do som adotadas estão descritas na tabela 4.15. A tabela 5.12 expõe a média obtida pela função informativa em relação à função som no *Software* “Palavras e Palavras”, sendo ela de 5 valores que corresponde ao valor máximo.

2. Funções do Som	
Função Informativa	
Professora A	5
Professora B	5
Professora C	5
Professora Apoio D	5
Professora Apoio E	5
Média	5

Tabela 5.12 Média obtida pela Função Informativa no *Software* “Palavras e Palavras”

A tabela 5.13 apresenta a média obtida pela função apelativa, tendo sido de 5 valores, o que corresponde ao valor máximo.

2. Funções do Som	
Função Apelativa	
Professora A	5
Professora B	5
Professora C	5
Professora Apoio D	5
Professora Apoio E	5
Média	5

Tabela 5.13 Média obtida pela Função Apelativa no Software “Palavras e Palavras”

A tabela 5.14 apresenta a média da função poética no Software “Palavras e Palavras”, sendo ela de muito bom.

2. Funções do Som	
Função Poética	
Professora A	5
Professora B	5
Professora C	5
Professora Apoio D	5
Professora Apoio E	5
Média	5

Tabela 5.14 Média da Função Poética no Software “Palavras e Palavras”

Apresentamos o último resultado relacionado com o som, assim a tabela 5.15 descreve os valores obtidos pela função indagadora. A média obtida está próxima do muito bom.

2. Funções do Som	
Função Indagadora	
Professora A	4
Professora B	5
Professora C	5
Professora Apoio D	4
Professora Apoio E	5
Média	4,6

Tabela 5.15 Média da Função Indagadora no Software “Palavras e Palavras”

O gráfico 5.5 apresenta o conjunto das médias obtidas em cada um dos itens, correspondentes à função som.

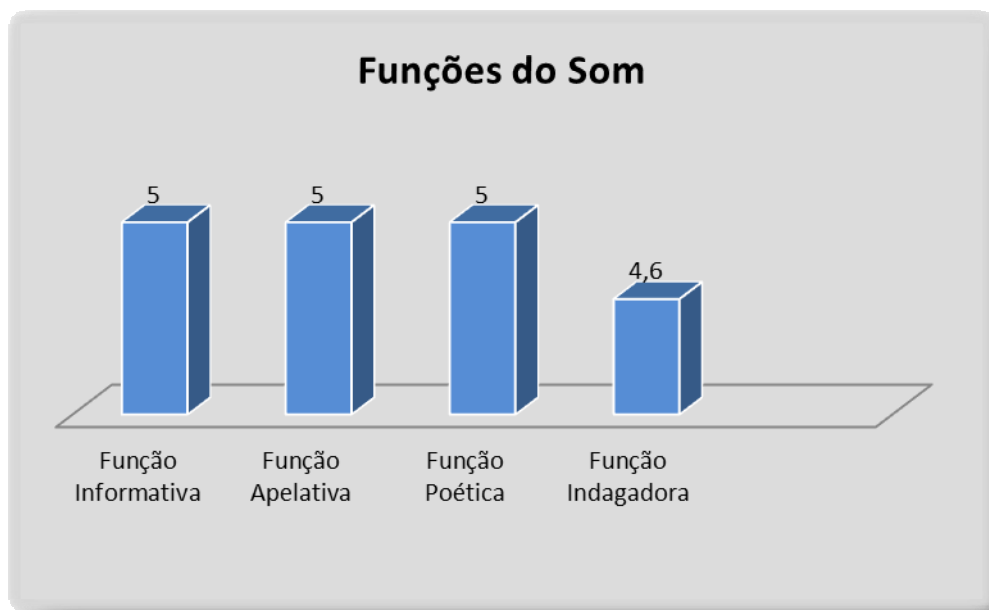


Gráfico 5.5 Resultado obtido pela Função do Som no Software “Palavras e Palavras”

Funções da Imagem

Como foi referido no capítulo 4, as imagens têm um papel muito importante no desenvolvimento de *software*. Como tal, a imagem desempenha determinadas funções nesse processo. Portanto é necessário efetuar uma avaliação a essas funções para que se possam melhorar os aspetos referentes às mesmas.

Salientemos que as funções das imagens foram analisadas em 2 perspetivas, sendo a primeira analisada pela imagem em si própria e o seu papel em relação à interatividade. A segunda perspetiva é analisada nas funções didáticas da imagem no processo ensino aprendizagem, logo é avaliada fase à interatividade e aos aspetos pedagógicos (ver este ponto).

A tabela 5.16 apresenta o resultado obtido na avaliação, para a função absoluto. Enquanto a tabela 5.17 descreve os resultados para a avaliação na função de colocação estratégica e a tabela 5.18 descreve os resultados para a função de reação do utilizador. A descrição destas funções está na tabela 4.13.

2. Funções da Imagem	
Função Absoluto	
Professora A	5
Professora B	5
Professora C	5
Professora Apoio D	5
Professora Apoio E	5
Média	5

Tabela 5.16 Média da Função Absoluto no *Software* “Palavras e Palavras”

2. Funções da Imagem	
Função de Colocação Estratégica	
Professora A	4
Professora B	5
Professora C	5
Professora Apoio D	5
Professora Apoio E	5
Média	4,8

Tabela 5.17 Média da Função de Colocação Estratégica no *Software* “Palavras e Palavras”

2. Funções da Imagem	
Função de Reação do Utilizador	
Professora A	4
Professora B	5
Professora C	5
Professora Apoio D	5
Professora Apoio E	5
Média	4,8

Tabela 5.18 Média da Função de Reação do Utilizador no *Software* “Palavras e Palavras”

O gráfico 5.6 representa os resultados obtidos pelas funções da imagem consideradas no *Software* “Palavras e Palavras”.

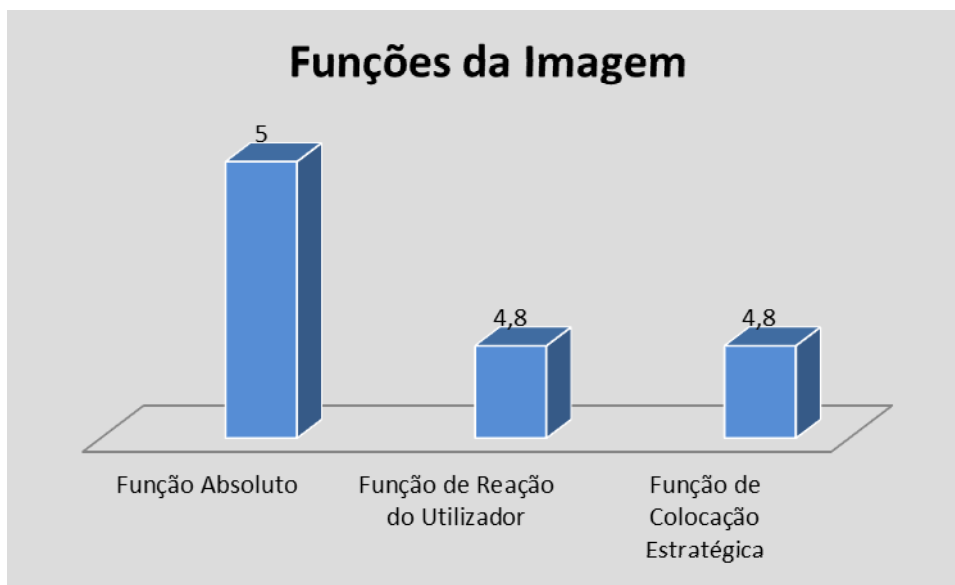


Gráfico 5.6 Resultado obtido por Funções da Imagem no *Software* “Palavras e Palavras”

Ferramentas de Exploração e Usabilidade

Este item de avaliação descreve a usabilidade com maior detalhe. Para tal, houve de inserir um conjunto de perguntas novas direcionadas exclusivamente para a usabilidade. Apresentamos somente o resultado aferido por essas perguntas, visto que as respostas às restantes perguntas foram iguais às obtidas com o questionário anterior. Optámos por colocar em apêndices as tabelas que evocam os resultados deste item e mostrar, neste local, somente os gráficos. Referimos que estabelecemos uma valorização que vai desde - 1 (muito mau), 5 (muito bom) para cada uma destas perguntas. Assim mantemos o mesmo tipo de avaliação.

Usabilidade

A avaliação da usabilidade no *Software* “Palavras e Palavras” foi analisada na ótica das ferramentas de exploração e nos aspetos pedagógicos. Em relação ao item ferramentas de exploração, foram criadas três novas perguntas, sendo elas: o equilíbrio visual é mantido em toda a aplicação; o ponto focal é mantido durante toda aplicação; a unicidade versus harmonia existe em toda a aplicação (ver tabela 4.13). Estas perguntas foram criadas pelo facto de termos de relacionar a interatividade com os princípios de composição, que influenciam a facilidade de encontrar a informação durante toda a aplicação.

A situação referida no parágrafo anterior está de acordo com o modelo de Boehm (ver figura 2.14) essencialmente na área de testes que engloba: a autodestruição, a estrutura, a comunicabilidade e a acessibilidade, e que estão englobados no nosso modelo IAQE e no método MIAQE.

A tabela com os resultados dessa avaliação encontra-se no apêndice C, mas a média de cada um desses itens é evidenciada no gráfico 5.7.

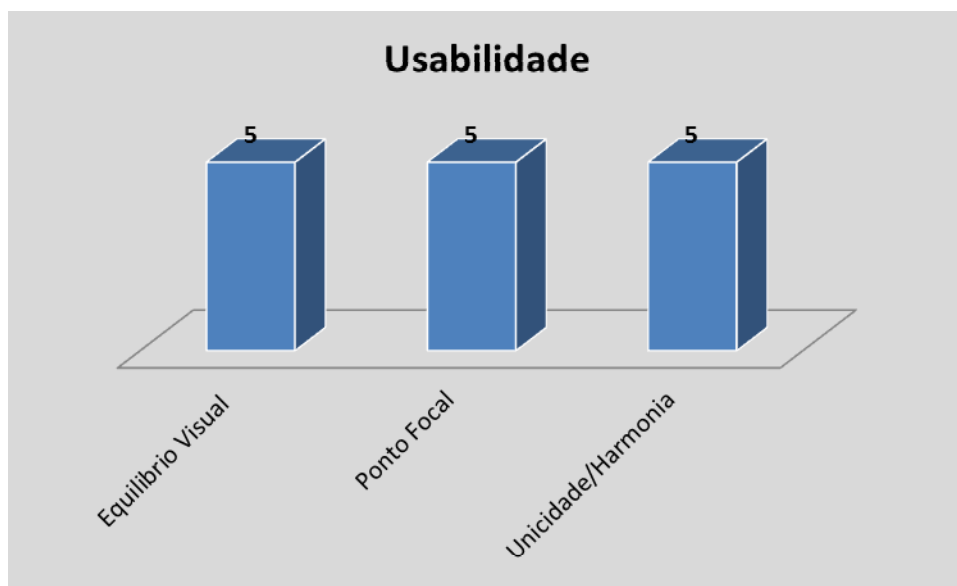


Gráfico 5.7 Média obtida em cada item da usabilidade no *Software* "Palavras e Palavras"

Aspetos Pedagógicos

Em relação aos aspetos pedagógicos foram analisados os seguintes pontos: Funções Didáticas do Som (Função Poética); Funções Didáticas da Imagem (Catalisadora de Experiências, Informativa, Estética e Identificação/Associação). As respostas a cada uma destas funções encontram-se no apêndice D.

No entanto, apresentamos a média de cada uma dessas funções através do gráfico 5.8.

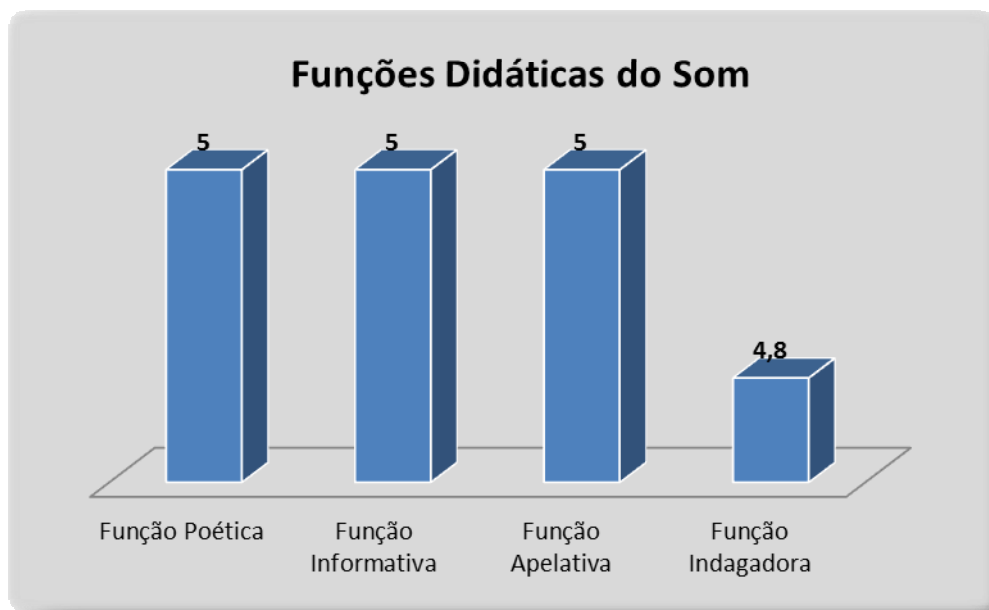


Gráfico 5.8 Média obtida em cada item das Funções didáticas do Som no Software “Palavras e Palavras”

5.2 Testes e Resultados obtidos a Nível Qualitativo

“Qualquer instrumento de avaliação, seja ele um teste, uma prova, uma escala, uma ficha ou uma grelha de entrevista, pode definir-se como um conjunto de itens, questões ou situações mais ou menos organizado e relacionado com um certo domínio a avaliar” (Freire & Almeida, 2008)

Optámos por efetuar dois tipos de avaliação: a avaliação heurística (ver capítulo 2) de acordo com o anexo L e a ficha de avaliação parte III de Rodrigues (Rodrigues E. , A Contribution for the Evaluation of Educational Interactive Software, 2002) (ver apêndice A). Não foi necessário efetuar a avaliação a nível da Nova Taxonomia de Avaliação de Software Educacional (Parte III), porque não houve alterações a nível qualitativo.

Decidimos situar a avaliação Heurística a nível qualitativo, visto que não é possível quantificar esta avaliação. Visto que é baseada num método de inspeção especializado de interfaces suportado nas 10 heurísticas revistas de Nielsen, nas quais são percorridas as diferentes interfaces (parte aluno e parte professor) que são confrontadas com as referidas heurísticas. Esta avaliação só se foca nas interfaces da aplicação e na sua importância na

interação pessoa-computador e não em outras dimensões do desenvolvimento da aplicação. Na utilização deste método não são referidos os potenciais pontos fortes da usabilidade da aplicação, mas somente os potenciais problemas.

Logo, uma interface ao “passar” nesta avaliação simboliza que está de acordo com as premissas das 10 heurísticas, enquanto o “não passar” traduz que, em algum momento, uma ou mais dessas premissas não está a ser validada. Nessa altura, são sugeridos um conjunto de conselhos e indicações ou propostas de alteração da interface de forma a contemplar os parâmetros da avaliação heurística.

5.2.1 Resultados da Avaliação Heurística

Para a avaliação heurística o avaliador Bruno (Bruno, s/data) teve em conta as 10 heurísticas de Nielsen (de recordar):

- Visibilidade do Estado do Sistema
 - O sistema deve manter sempre o utilizador informado sobre o que está a acontecer na interface através do fornecimento de feedback apropriado dentro de tempos razoáveis.
- Correspondência entre o Sistema e o Mundo Real
 - O sistema deve utilizar a linguagem do utilizador, usando palavras, frases e conceitos familiares (ao invés de usar termos relacionados com o software). O sistema deve seguir as convenções do mundo real, fazendo a informação aparecer na ordem lógica e natural, devendo ser feito uso de metáforas universais sempre que possível.
- Liberdade e Controle do Sistema pelo Utilizador
 - Os utilizadores geralmente cometem enganos ao escolher opções do sistema. Neste caso, precisam de recuperar desses estados indesejados, sem ter que passar por diálogos extensos. Recursos recomendáveis para estas situações são por exemplo, as opções de desfazer e refazer.
- Consistência e Aderência a Padrões
 - Os utilizadores não devem ser confundidos perante opções de interface que não são universais naquele tipo de diálogos. Recomenda-se o uso de um

conjunto de padrões/convenções que sejam facilmente identificáveis pelos utilizadores.

- Prevenção de Erros
 - O projeto de interação deve privilegiar a prevenção de erros. Nielsen afirma que um projeto cuidadoso que previne o aparecimento de erros é melhor do que o fornecimento de boas mensagens de erros.
- Reconhecimento em vez de memorização
 - O projeto deve focar no reconhecimento e impedir que o utilizador tenha que memorizar aspetos relativos à interação. Isto pode ser conseguido tornando-se os objetos, ações e opções visíveis. O utilizador não deve ser obrigado a relembrar uma parte do diálogo para poder dar sequência à sua interação. Instruções para o uso do sistema devem ser visíveis e facilmente recuperáveis.
- Flexibilidade e Eficiência na Utilização
 - O sistema deve fornecer aceleradores de tarefas que podem oferecer maior velocidade de interação para os utilizadores mais experimentados. Além deste recurso, deve-se proporcionar ao utilizador mecanismos para criar atalhos para ações frequentes.
- Desenho de Ecrã Estético e Minimalista
 - Os diálogos não devem conter informação irrelevante ou raramente necessária. Qualquer unidade extra de informação num diálogo compete com unidades relevantes, diminuindo a sua visibilidade relativa.
- Auxílio no Reconhecimento, Diagnóstico e Recuperação de Erros
 - As mensagens de erro devem ser expressas em linguagem natural, sem códigos e de maneira clara. O teor da mensagem deve indicar o problema com precisão e sugerir uma solução construtiva.
- Ajuda e Documentação
 - Permitir que qualquer informação seja facilmente pesquisada e tenha como foco principal a tarefa do utilizador; Deve conter listas concretas de passos a executar; Deve ser concisa.

Todos os ecrãs foram avaliados desde H2-1 a H2-10. O resultado desta avaliação encontra-se no apêndice B. Após a avaliação foram efetuadas as alterações indicadas.

5.2.2 Resultados Obtidos Através da Ficha de Avaliação de Software Educacional Parte III de Rodrigues

“A investigação qualitativa envolve pegar nos objectos e acontecimentos e levá-los ao instrumento sensível da sua mente de modo a discernir o seu valor como dados” (Bogdan & Biklen, 1994)

A nível qualitativo analisámos os aspetos pedagógicos, a interface e alguns itens das ferramentas de exploração através das respostas “sim” e “não”.

Interface

As respostas dadas pelos avaliadores para o item 1 foram as seguintes:

- Cor agradável – sim;
- Som – sim;
- Vídeo – não;
- Texto – sim;
- Imagens – não.

As respostas ao item 2 foram as seguintes:

- Imagens significativas e claras – sim;
- Botões e links – sim.

Ferramentas de Exploração

As respostas ao item 1 do ponto ferramentas de exploração, foram sempre de “sim”, como se constata na tabela 5.19.

Pergunta	Tipo de Resposta – Sim/Não
Existências de botões de retorno à ação anterior	Sim
Informação geral para a manipulação de <i>software</i>	Sim

Tabela 5.19 Resposta ao item 1 do *Software “Palavras e Palavras”*

A resposta ao item 3 – “Inclui apoio à utilização pedagógica do programa”. Foi sempre de “sim” (documento descritivo de utilização da aplicação). Enquanto a resposta ao item 4 – “*Outros materiais de apoio*” foi sempre de “não”.

Materiais de Apoio

A resposta ao item 3 – “*Inclui apoio à utilização pedagógica do programa?*” – “*Sim. Documento em html*”.

A resposta ao item 4 – “*Outros materiais de apoio?*” – “*Não*”

Aspetos Pedagógicos

Em relação ao item 3 – “*Adaptabilidade à problemática e à capacidade do aluno (sim/não Descreva)*”

- Visuais – “sim”;
- Auditivos – “não”;
- Motores – “não”;
- Cognitivos – “sim”;
- Comunicação/linguagem – “sim”.

Observações: Adaptabilidade à problemática visual – utilização de teclados aumentativos. Em relação aos itens cognitivos e comunicação/linguagem – a introdução dos dados no *software* é de acordo com as características do aluno.

O item 6 – “*Permite uma aprendizagem permite uma resposta auto- didata? (sim/não)*” – obteve como resposta – “*sim, porque permite ao aluno consolidar sozinho determinados conteúdos recebidos em contexto de sala de aula*”.

Em relação ao item 8 – “*Permite avaliar a evolução do aluno*”. A resposta é “*sim*” e “*porque permite verificar as respostas dos alunos em diversos períodos de tempo*”.

No item 9 – “*O software permite mudar de um passo mais avançado para um passo mais simples com o objetivo de superar as dificuldades no processo de aprendizagem*”. Obteve sempre a resposta “*sim*” e “*porque o software apresenta 3 jogos diferentes (Constrói Palavras, Constrói Frases e Descobre Palavras) cada um deles com níveis de dificuldade diferentes*”.

O item 10 – “*É possível utilizá-lo em grupo*”. A resposta foi sempre de “*Não*”.

Conclusões

O enfoque deste capítulo, como já foi mencionado centra-se descrição das ferramentas utilizadas na realização dos testes e na análise dos resultados do *Software* “Palavras e Palavras”. Para tal, foi necessário ter em conta o modelo IAOE e a sua metodologia destacados no capítulo 3, assim como todo o capítulo 4.

Com vista a testar a qualidade foram analisados os aspetos interativos, de usabilidade e pedagógicos. Logo, foram analisadas em detalhe as associações entre a imagem, texto, som, texto versus som, imagem versus movimento e som. Da análise dos resultados não há uma prevalência significativa destas associações em detrimento de outras. Pois cada uma destas associações tem o seu papel no desenvolvimento de *software* orientado para o processo de ensino.

A relação entre as associações referidas no parágrafo anterior, são mais dinâmicas ou mais passivas, podemos mesmo dizer que é o caso do menu da área de navegação que apresenta imagens estáticas o que torna a comunicação mais passiva. Para colmatar este facto de forma a tornar o menu mais comunicativo e apelativo, utilizam-se ícones (por exemplo, o mocho), o contraste das cores e o som. Os ícones estão associados a uma linguagem, que alude a uma

tarefa a desenvolver e o passar do cursor do rato sobre os ícones permite ouvir a mensagem associada. Despertando no aluno a curiosidade de saber o que está para além desse ícone. O que torna o menu da área de navegação mais apelativo e comunicativo.

Realçamos que no início do 1º ano do 1º ciclo os alunos ainda não sabem ler! Mas surgem nos ecrãs do *Software* “Palavras e Palavras” imagens com texto, porém estas não surgem isoladas, estão sempre acompanhadas de som, de forma a corrigir tal situação. Visto que a imagem (ícones) associada ao movimento e som torna a comunicação ativa, desperta mais a atenção do aluno, levando-o a interagir praticamente em simultâneo, criando uma interatividade entre o aluno e o *Software* “Palavras e Palavras”, como se pode constatar através dos resultados obtidos durante os testes. Visto que nos aspetos pedagógicos as funções didáticas do som alcançaram praticamente o valor máximo, desta forma aumenta a inteligibilidade da comunicação no processo ensino-aprendizagem.

Os resultados da interatividade e da usabilidade são extremamente positivos, o que nos leva a concluir que promovem a autonomia e orientando o desempenho do aluno. Porém na interatividade às funções do texto (na função apelativa) devem ser melhoradas, visto a sua média ser de 3,8 para 5 valores. O resultado das funções do som e das funções da imagem foram extremamente gratificantes, já que atingiram praticamente o topo da avaliação.

Através desta avaliação também provamos que o local de colocação da informação, o ponto focal, a unicidade/harmonia das interfaces e a metodologia de desenvolvimento do modelo IAOE e do método MIAOE são fatores essenciais para a qualidade do *Software* “Palavras e Palavras”, aumentando assim o seu grau de usabilidade.

O facto de o aluno ser apoiado no seu desempenho durante a exploração dos jogos do pelo *feedback* “do mocho” ao saber de imediato se executou ou não de forma correta determinada tarefa e ao ser estimulado para continuar, criou uma empatia pelo *software*. Aqui, as funções do som revelaram-se um elemento primordial no desenvolvimento de *software* educacional/educativo, como se pode ver pelo gráfico 5.5.

Os testes a nível qualitativo foram analisados pelo número de respostas “sim” e “não”. Contabilizando esse número, verificamos que na totalidade as respostas “sim” foram superior às respostas “não”.

Conclusões e Trabalho Futuro

“Nenhum homem vos pode revelar nada que não repouse já meio adormecido na manhã do vosso conhecimento” (Gibran, 2009).

Neste capítulo, apresentamos as conclusões finais com base nas conclusões de todos os capítulos, na observação direta e indireta, na revisão da literatura e em toda a investigação realizada. Fazemos também algumas considerações de recomendações que poderão servir de base a investigações futuras.

6.1 Conclusões

Este estudo não teve como princípio medir o estilo de aprendizagem de cada aluno, no entanto, foi tido em conta como ponto de partida para a criação do modelo IA, do modelo IAOE e do método MIAOE, visto que o uso das novas tecnologias no ensino pode ser usado como um meio de resposta às diversidades de aprendizagem.

A diferença entre o modelo IA e o modelo IAOE reside no fato de o modelo IAOE estar diretamente focado para o ensino enquanto o modelo IA pode abranger diversas áreas.

Como tal, o estudo às diferentes teorias de aprendizagem e às teorias cognitivas permitiu-nos perceber e entender a complexidade do processo de ensino-aprendizagem. De acordo com Piaget, a criança na faixa etária compreendida entre os 7-11 possui a noção de reversibilidade é capaz de numerar, classificar, selecionar e ordenar. A sua flexibilidade de pensamento permite um sem número de aprendizagens. Quando uma criança inicia a escolaridade, os processos de aprendizagem básicos como a leitura e a escrita são primordiais para a passagem a níveis

superiores de aprendizagem. Para tal, tem de existir uma interação entre o professor e o aluno, para que este último passe de um estado de conhecimento para outro estado mais avançado. Por vezes, é necessário explicar a mesma matéria de diversas maneiras, até que o aluno a entenda, interprete e possa interferir no conhecimento que está a adquirir. Visto que todos os alunos são excecionalmente diferentes, compete ao professor levar o aluno para além da memorização de factos e dados, de forma a adquirir e promover novos conhecimentos.

Ao considerar que o conhecimento dos alunos é o equivalente a um “copo meio cheio” e ávido de ser enchido, compete ao professor executar essa tarefa. Para tal, o professor socorre-se de um conjunto de teorias e modelos de aprendizagem e de ferramentas que o ajudam de forma intencional e planeada.

Portanto, o professor tem perante ele um conjunto de teorias e modelos de aprendizagem, das quais adotamos algumas para o nosso estudo. Todas elas dão um contributo no processo ensino-aprendizagem, não sendo possível dissociá-las ou eleger uma como a primordial. Para algumas situações poderá ser melhor utilizar as teorias behavioristas para outras o sócio construtivismo de Vygotsky, as teorias cognitivas, a teoria comportamental, a teoria Behaviour de Watson, Skinner e de Bandura ou/e a teoria de inteligências múltiplas. Tudo depende do aluno em causa, das suas capacidades, das suas motivações e do que se pretende ensinar. Como tal, não podemos afirmar que a aprendizagem se faz somente por um único processo, nem é uma receita de uma metodologia. Deverá ser encarada como uma panóplia de elementos metodológicos que procuram encontrar diversas perspetivas e técnicas para o processo ensino-aprendizagem. Deverá o professor ser um ousado aventureiro neste tão digno processo.

Para que a aprendizagem seja um processo contínuo é necessário que um conjunto processos e factos se mantenham interligados como uma rede neurológica, onde a memória, o córtex visual, área de Broca e a área de Wernicke são elementos cruciais no processo ensino-aprendizagem.

Associar todas estas áreas do cérebro, os modelos e teorias de aprendizagem a ferramentas educativas/educacionais tecnológicas foi uma tarefa extremamente difícil. Pois todo o processo educativo deve ser individualizado. Individualizar uma ferramenta de apoio como um *software* foi um desafio. Adaptar um dispositivo como um rato, teclado e um ecrã, por intermédio dos quais o aluno interage com o computador, pode não ser complicado, mas a adaptação do

software acarreta um grau de dificuldade diferente. Tem de ser analisado em dois sentidos distintos: a da interface e dos conteúdos.

É através da interface que existe um grau de interação mais efetiva entre o aluno e o computador, portanto houve uma particular atenção a toda a conceção e desenvolvimento do *Software* “Palavras e Palavras”, pois o seu objetivo principal é provar a viabilidade e eficácia do modelo IAOE, que contempla teorias de aprendizagem, teorias e modelos de qualidade de desenvolvimento de *software* e teorias e modelos ergonómicos de *software*.

Assim, toda a interface do nosso *software* leva o aluno e o professor a um envolvimento constante nas atividades apresentadas. Esse envolvimento é promovido pela interatividade e usabilidade presentes no *software* (de acordo com os resultados apresentados). É impossível dissociar a interatividade e a usabilidade; de certa forma formam um “corpo”. A interatividade entre o aluno e a aplicação, entre o professor e a aplicação é intrínseca à prática pedagógica bem-sucedida. No nosso estudo, consideramos a interatividade como uma arte, porque requer um detalhe e preciosismo de várias áreas e habilidades, que vai desde o desenho gráfico passando pelas linguagens de programação até à inserção dos conteúdos didáticos, em contexto de sala de aula. De tal forma, que dá ao aluno a capacidade de explorar toda a aplicação do *software* ao seu próprio ritmo. No que respeita ao professor oferece-lhe o cargo de administração dos conteúdos da aplicação, de forma individualizada e autónoma. Desta forma, estabelece com estes atores uma comunicação ativa.

Em termos de usabilidade, a “ajuda” representa um dos seus patamares, é específica de cada situação, está destinada à navegação na aplicação e na concretização das atividades propostas. Consideramos a cor e as imagens, o movimento e o som como uma “ajuda”. A cor e o contraste da cor demonstraram ser muito importantes pois permitiram estruturar os assuntos nos ecrãs e orientar o aluno e o professor na exploração e na retenção de determinada informação. As imagens como o próprio nome indica, representam diretamente um objeto conhecido pelos alunos, sendo considerada um veículo de comunicação com o aluno e com o professor. A nosso ver e de acordo com Lopes (Lopes, 1991) “ (...) a comunicação pedagógica com a imagem – é uma comunicação intencional, (...) ” pois permite aos alunos e aos professores identificar a mensagem contida na imagem, logo estimula a retenção e compreensão dos conteúdos. Para tal, há uma coerência na colocação dos grafismos nos ecrãs ao longo de toda a aplicação.

O som e a narração foram muito importantes para este nível etário, porque visaram explicar os elementos existentes na interface e a forma do aluno e do professor se interrelacionarem com a aplicação. Mais uma vez estamos perante a interatividade da aplicação. Mais referimos, que nas situações em que utilizamos o discurso oral e o texto (inserir aluno, apagar, níveis, etc.) para apoiar o aluno e o professor a ajudá-los nas tomadas de decisão, foi uma mais-valia a nível de destreza e eficácia apresentada pelos atores (aluno e professor). Desta forma a nossa interface disponibiliza a ajuda sempre que necessário, pelo que foi concebida de forma intuitiva e eficaz, o aluno e o professor navegam por todo o *software* mediante o acesso a botões, áreas interativas e sob o menu de opções.

A associação de movimento e do som à imagem originou uma animação, que atrai a atenção do aluno para determinada zona do ecrã, motivando-o e reforçando a sua aprendizagem. Acreditamos, que a nossa interface é fácil de usar, de compreender e motivar. Motiva o aluno quando erra e quando acerta nas questões a resolver, portanto explora o coeficiente emocional do aluno para um envolvimento cognitivo, passando assim para níveis superiores de dificuldade de conteúdos.

Não foi só através da interatividade e usabilidade que verificamos a qualidade do nosso *software*. As atividades são outro elemento essencial, pois valida a qualidade pedagógica do *software*. Estas apresentam diferentes graus de dificuldades o que incentivou os alunos a melhorarem o seu desempenho visto-lhes ser atribuída uma classificação, o que lhes fomenta o desejo de pontuar pelo máximo possível.

Salientemos que o *software* não tem inserido os conteúdos didáticos, estes são introduzidos pelo professor em contexto de sala de aula e de acordo com as dificuldades ou necessidades de cada aluno, o que o torna adaptável. Portanto, à primeira vista e em sentido figurativo o *software* é meramente um esqueleto, no qual o professor introduz todos os conteúdos necessários à promoção de um específico conhecimento num determinado aluno.

Podemos dizer que os objetivos propostos foram realizados: sistematizar um conjunto de atividades relacionadas com o processo de análise ergonómica das tarefas, quando esta está implícita no desenvolvimento de *software* interativo e adaptável para o ensino; desenvolver uma metodologia interativa e adaptável para o ensino; desenvolver um modelo interativo adaptável no processo ensino aprendizagem; adaptar o referido modelo ao utilizador qualquer

que seja ele; testar a capacidade evolutiva do aluno face à metodologia desenvolvida; testar a capacidade evolutiva do aluno face à evolução do modelo, ou seja à medida que o modelo evolui o aluno deve aprender conceitos mais complexos.

No entanto é de referir que tivemos imensas dificuldades neste estudo pois houve muitas mudanças em termos de conteúdos pedagógicos, novos professores em sala de aula, alunos com necessidades específicas, falta de professores para os alunos do ensino especial, etc. Estas condicionantes implicaram mudanças em determinados processos, perante as quais tínhamos a sensação de voltarmos atrás constantemente. Mas, tudo isto foi ultrapassado da melhor forma. A prova está nos resultados obtidos e no interesse dos professores e dos alunos em continuarem a utilizar o *software*. Esperamos que, brevemente, venha a ser utilizado em mais escolas do 1º ciclo.

Salientamos que o modelo IAOE e o método MIAOE é reflexo de uma boa gestão entre a qualidade do produto e a qualidade do processo de desenvolvimento. Acreditámos que o nosso modelo responda a essa necessidade, já que permite que o modelo seja interativo, didático e adaptável, a qualquer aluno e adequado aos pressupostos pedagógicos.

6.2 Trabalho Futuro

Ao termos realizado um estudo em duas áreas tão distintas, sendo uma delas a área do conhecimento humano, permanece a ideia que nem tudo foi dito e muito ficou por conceber. De certeza absoluta algo ficou por realizar! Principalmente durante o ano seguinte após os testes foram-nos surgindo algumas questões que entendemos partilhar pela sua pertinência para futuros estudos.

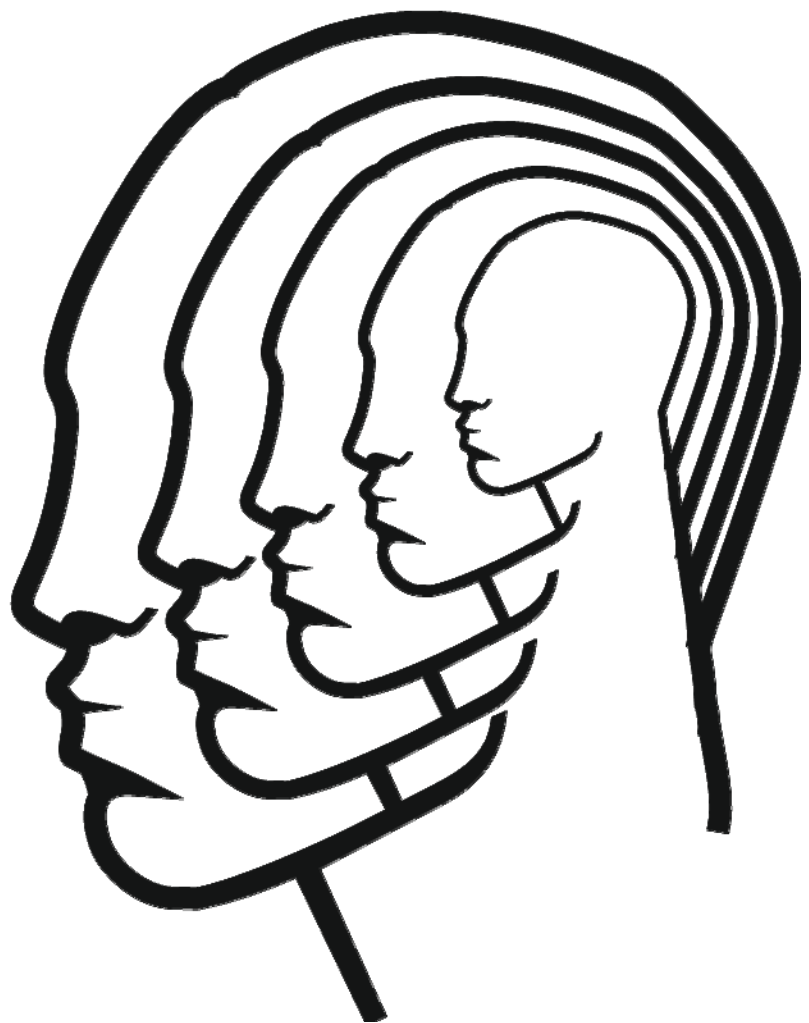
Esses estudos que podemos indicar não colocam em causa a modelo MIA nem o método MIAOE, mas sim como os podemos aplicar de forma mais correta. Tendo como ponto de partida o *Software* “Palavras e Palavras”, e ao pensar expandi-lo para outras línguas. Que alterações teriam de existir para proporcionar essa mais-valia? Que alterações teriam de se

realizar para que a atual aplicação possa ser executada noutras resoluções, de forma eficiente, com maior eficácia e com a mesma qualidade?

Ainda que seja possível adicionar conteúdos em outras línguas, neste momento não é possível alterar os botões e sons explicativos para outras línguas, nem adicionar outros temas de estudo em contexto de sala de aula.

Como tal sugerimos um conjunto de recomendações para trabalhos futuros:

- A adequação do *software* para outras línguas;
- A adequação do *software* para outros temas;
- Acrescentar mais ferramentas que abranjam mais alunos e professores no exercício das suas atividades.
- Estudos comparativos do desempenho dos alunos em relação à forma tradicional de ensino e à forma interativa, durante 1º ciclo do ensino.



Bibliografia

International standards for HCI and usability. (2003). Retrieved 2006, from Usability Net:
http://www.usabilitynet.org/tools/r_international.htm

Abrunhosa, M. A., & Leitão, M. (2009). *Psicologia B*. Lisboa: ASA.

Afonso, R. W. (2004). *Análise da Integração de múltiplos formatos no software educativo multimédia*. Universidade do Minho - Instituto de Educação e Psicologia, Braga.

Alcino, S. (1998, Julho). *Folha do Alcino - Formação de Professores Texto de Apoio - Avaliar ?*
Retrieved 2010, from Folha do Alcino - Formação de Professores:
<http://www.prof2000.pt/users/folhalcino/formar/outros/avaliar.htm>

Alonso, C. M., & Gallego, D. J. (2000). *Aprendizaje y ordenador*. Madrid: Dykinson.

Altet, M. (2000). *Análise das Práticas dos Professores e das Situações Pedagógicas*. (J. F. Cláudio, Trans.) Porto, Porto, Portugal: Porto Editora,Lda.

Amantes, A., & Borges, O. (s.d.). *The Use of SOLO'S TaxonomyY As Metodologic Tool In Eduacional Research*. Retrieved 2008, from
<http://www.fae.ufmg.br/abrapec/viempec/CR2/p678.pdf>

Amorim, T. (2009, Julho 16). *RAREFECUNDO | Thiago Amorim*. Retrieved Outubro 2009, from RAREFECUNDO | Thiago Amorim Tecnologias,Sociedade, (in)Justiça, Utilitários, Filosofia, Sistemas de Informação(SI) etc.: <http://rarefecundo.com/2009/07/16/micro-introducao-interface-homem-computador-hci-ihc/>

- Anderson, L. W., & Krathmohl, D. R. (2001). *A Taxonomy for Learning, Teaching and Assessing*. New York: Addison Wesley Longman.
- Antunes, C. (2005, 01 05). (R. Barros, Ed.) Retrieved 2010, from Revista Educação: <http://revistaeducacao.uol.com.br/textos.asp?codigo=10624>
- Antunes, C. (2005, 01 05). (R. Barros, Ed.) Retrieved 2010, from Revista Educação: <http://revistaeducacao.uol.com.br/textos.asp?codigo=10624>
- Antunes, C. (2008, Setembro 26). Desenvolvendo o carácter a partir de valores e virtudes. *Canal de neadunincor*. (C. d. neadunincor, Interviewer) Brasil.
- Arnheim, R. (2002). *Arte y Percepción Visual*. Espanha: Editorial Alianza - Suave.
- Arruda, S. F. (1998). *Frases Célebres Notáveis* (2ª ed.). (L. N. S.A., Ed.) São Paulo, Brasil: Paym Gráfica Editora Ltda.
- Association, I. S. (1990). *IEEE Find standards - 610.12-1990 - IEEE Standard Glossary of Software Engineering Terminology*. Retrieved 2007, from IEEE Advancing Technology for Humanity : <http://standards.ieee.org>
- Ayres, D. A. (2009, Junho 31). *BOLETÍN REDEM*. Retrieved Setembro 20, 2009, from <http://www.redem.org/boletin/boletin310709f.php>.
- B.Dahl, C. (2008). Using the SOLO Taxonomy to Analyze Competence Progression of University Science Curricula. *Higher Education*, 531-549.
- B.Shneiderman. (1998). *Designing The User Interface Strategies for Effective Human-Computer Interaction*. Addison-Wesley.
- Baldas, M., Campos, S., & Coutinho, J. (2006). *Departamento de Informática*. Retrieved from Pontifícia Universidade Católica do Rio de Janeiro: <http://www.inf.puc-rio.br/~inf1379/DFD-Regras.pdf>
- Bandura, A. (1977). *Social Learning Theory*. Pearson.

- Bauersfeld, H. (1998). *Interaction, Construction and Knowledge: Alternative perspectives for mathematics education*. Retrieved 2009, from <http://webcache.googleusercontent.com/search?q=cache:35SOy4aYRIgl:fordis.es.e.ips.pt>
- Berrocso, J. (2001). *Facultad de Formación del Profesorado*. Retrieved 2009, from Universidade de Extremadura: <http://www.unex.es/didactica/TecnologiaEducativa/PDF/Imagen.pdf>
- Bertoldi, S. (n.d.).
- Bertoldi, S. (1999). <http://www.inf.ufsc.br/~edla/orientacoes/sergio.pdf>. Retrieved 2005, 2008, 2009
- Bevan, N. (1995). Measuring Usability as Quality of Use. *Software Quality Journal*, 4, 115-130.
- Bevan, N. (1999). Quality in Use: Meeting User Needs for Quality. *Journal of System and Software*.
- Bevan, N., Claridge, N., M. Maguire, & M. Athousaki. (2002). Specifying and evaluating usability requirements using the Common Industry Format. *IFIP 17th World Computer Congress, Montreal, Canada 25-30* (pp. 133-148). Canada: Kluwer Academic Publishers.
- Biggs, J. (1992). *Bulletin of the Hong Kong Psychological Society*. Retrieved Janeiro 2008, from University of Hong Kong: <http://teaching.polyu.edu.hk/datafiles/R42.pdf>
- Biggs, J. (2003). Maidenhead: Open University Press.
- Biggs, J. (2003). *Teaching for Quality Learning at University*. Buckingham: Open University Press.
- Biggs, J. B., & Collis, K. F. (1982). *Evaluating the Quality of Learning: The SOLO Taxonomy Structure of the Observe Learning Outcome*. London: Academic Press.
- Biggs, J., & Collis, K. (1982). *Evaluating the Quality of Learning: The SOLO Taxonomy*. New York: Academic Press.
- Boehm, B. (2000). *Spiral Development Experience, Principles and Refinement, Software*. Engineering Institute. Pittsburgh: Carnegie Mellon University.

- Boehm, B., Brown, J., Kaspar, H., Lipow, M., Machleod, G., & Merritt, M. (1978). *Characteristics of Software Quality*. Elsevier Science Ltd.
- Bogdan, R., & Biklen, S. (1994). *Avaliação Qualitativa em Educação*. Porto: Porto Editora.
- Booch, G. (1991). *Object Oriented Design: With Applications*. California U.S.A: The Benjamin/Cummings Publishing Company.
- Boyle, T. (1997). *Design for Multimedia Learning*. London: Prentice Hall.
- Bronowski, J. (1976). *The Ascent of Man*. Little Brown and Company.
- Bruner, J. (2008). *Actos de Significado*. Edições 70.
- Bruno, L. C. (s/data). Avaliação Heurística ao Software "Palavras e Palavras". Beja, Baixo Alentejo, Portugal.
- Cabral, A. (1990). A Teoria do Jogo, "pedagogia-nº16". *Editorial Notícias*.
- Campos, G. H. (1994). *Metodologia para Avaliação da Qualidade de Software Educacional. Directrizes para Desenvolvedores e Usuários. Tese (Doutoramento em Engenharia)*. Rio de Janeiro: COPPE/UFRJ.
- Casas, S. (1987). *Didáctica del Video*. Barcelona: Editorial Alta Fulla.
- Castela, R. (2004, Outubro). *Acesso Rápido à Pasta dos professores*. Retrieved 2010, from Faculdade Assis Gurgacz: <http://www.fag.edu.br/>
- Castro, E. (2005). Memória e Aprendizagem - Aquisição e Retenção de Saberes. *Mestrado Educação Tecnologia Educativa*. Braga, Portugal. Retrieved 2010, from https://docs.google.com/viewer?a=v&q=cache:Jy8CtlX129kJ:elisacarvalho.no.sapo.pt/pdf/psicologia.pdf+o+c%C3%A9rebro+de+uma+crian%C3%A7a+consegue+recordar+12+imagens&hl=pt-PT&pid=bl&srcid=ADGEESjOurqw1fB5d9yl8z9otVI7SfO2f3tepj3j4LFeL0obn_v1YjNVaYPSa9RuXvj-x

- Cebrian, M. (s/data). *La incorporación de la Informática a la enseñanza*. Retrieved Janeiro 2012, from Grupo de Tecnología Educativa: <http://tecnologiaedu.us.es/cursos/29/html/cursos/tema8/presenta.htm>
- Chagas, I., Bettencourt, T., & Marques, L. (sem data). Retrieved Maio de 2010, from <http://lsm.dei.uc.pt/ribie/docfiles/txt20037292255A%20Internet%20e%20a%20constru%C3%A7%C3%A3o.pdf>
- Coelho, P. (1990). *O Diário de um Mago*. Lisboa: Pergaminho, Lda.
- Collins, J., Hammond, M., & Wellington, J. J. (1997). *Teaching and Learning With Multimedia*. London: Routledge.
- Corrêa, F. P. (2003). *Carga Mental em Ergonomia; Universidade Federal de Santa Catarina; Dissertação de Mestrado*. universidade Federal de Santa Catarina, Santa Catarina Brasil.
- Costa, F. A. (1999). Contributos para um Modelo de Avaliação de Produtos Multimédia Centrado na Participação dos Professores. *1º Simpósio Ibérico de Informática Educativa*. Aveiro.
- Costa, F. A. (1999). Contributos para um Modelo de Avaliação de Produtos Multimédia Centrado na Participação dos Professores. *1º Simpósio Ibérico de Informática Educativa*. Aveiro.
- Cowan, N. (2001, Fevereiro). *Cambridge Journals*. Retrieved 2008, from Cambridge Journals: <http://journals.cambridge.org/>
- Cowman, L. (1999). *Mananciais no Deserto* (2 ed.). Betânia.
- Craig, R. D., & Jaskiel, S. P. (2002). *Systematic Software Testing*. London: Artech House Publishers.
- Craig, R., & Jaskiel, S. (2002). *Systematic Software Testing*. Boston: Artech House Publishers.
- Cross, C. (1990). *Le Design des Didactiles: Guide pratique pour la Conception de Scénarios Pédagogiques Interactifs*. Paris, france: ACL, Editions.
- Csikszentmihalyi, M., & Csikszentmihaly, I. S. (1998). *Optimal Experience Psychological Studies of Consciousness*. Cambridge, United Kingdom: Cambridge University Press.

- Cury, A. (2009). *Filhos Brilhantes Alunos Fascinantes*. (Pergaminho, Ed.)
- Cybis, W. d. (1997). *Qualidade do software na interação com o usuário: uma abordagem ergonômica*. Florianópolis: LabUtil.
- Dahlstedt, A. (2003). Retrieved 2007, from University of Skovde:
http://www.ida.his.se/ida/kurser/informationssystem_engineering/kursmaterial/forelasni
- DeMarco, T. (1978). *Structured Analysis and System Specification*. Yourdon INC.
- Desconhecido. (2007, 04 25). *News of the Wild*. Retrieved 2009, from simbiotica.org, a Rede Simbiótica de Biologia e Conservação da Natureza:
<http://yourwebapps.com/WebApps/mail-list-archive.cgi?list=65673;newsletter=1014>
- Desconhecido. (n.d.). *engenharia de software*. Retrieved 2010, from Slideshare:
<http://www.slideshare.net/CursoSENAC/engenharia-de-software>
- Desconhecido. (S/data). *Instituto Português da Qualidade*. Retrieved Março 2010 , from
<http://www.ipq.pt/custompage.aspx>
- Desconhecido. (sem data). *Universidade de Aveiro*. Retrieved Janeiro 2010, from
<http://www.doc.ua.pt/PagelImage.aspx?id=5459>
- Diéguez, J. L. (1995). *Los Lenguajes en los medios: lenguaje verbal y lenguaje icónica*. Alcoy: Editorial Marfil.
- Dillon, A. (1993). *University of Texas School of Information Publications*. Retrieved 2009, from University of Texas School of Information:
<http://www.ischool.utexas.edu/~adillon/BookChapters/collaborative%20writing.htm>
- Dix et all, 1. (1998). *Human Computer Interaction*. Prentice Hall Europe.
- Dix, A., Finlay, J., D.Abowd, G., & Beale, R. (2004). *Human-Computer Interaction* (3ª Edição ed.). England: Pearson Education Limited.
- Dondis, D. A. (2007). *Sintaxe da Linguagem Visual*. São Paulo, Brasil: Martins Fontes.

- Drews, C. (2007, Novembro 22). *PsicologiaRG*. Retrieved 2012, from <http://psicologiarg.blogspot.pt/2007/11/algumas-coisas-para-as-quais-o-crebro.html>
- Dronkers, N. (2000, Setembro 28). A relação entre a gratuidade da afasia de Broca e área de Broca. *Behavioral and Brain Sciences*. *Cambridge Journals*, 23, 30-31.
- Duarte, F. J. (2002, Julho). Retrieved 2009, from Universidade do Minho: <http://repositorium.sdum.uminho.pt/bitstream/1822/365/1/Disserta%c3%a7%c3%a3o%20de%20Mestrado%20-%20Francisco%20Duarte.pdf>
- E.Rodrigues, J.Moreno, & D.Grilo. (2009). New Partices of Learning Adjustable Software. In A. M. A. Vilas-Méndez (Ed.), *V International Conference on Multimedia and ICT in Education (m-ICTE2009)*. I, pp. 417-420. Lisboa, Portugal: Formatex, Badajoz, Espanha.
- Eason. (1991, Junho). *Informaworld contents*. Retrieved 2007, from *linformaworld*: <http://www.informaworld.com/smpp/content~content=a779205913&db=all>
- Eason, K. (1991, Junho). *informaworld contents*. Retrieved 2007, from *informaworld*: <http://www.informaworld.com/smpp/content~content=a779205913&db=all>
- Eason, K. D. (1991). Ergonomic perspective on advances in human-computer interaction. *Ergonomics*, (pp. 721-741).
- Editora, P. (2003). *infopédia*. Retrieved Novembro 2009, from <http://www.infopedia.pt/lingua-portuguesa/normas>
- Einstein, A. (s/data). *pensador.info/autor*. Retrieved Maio 2010, from *pensador.info*: http://www.pensador.info/autor/Albert_Einstein
- Eysenck, M. W., & Keane, M. T. (2005). *Cognitive Psychology - A Student's Handbook*. London: Psychology Press.
- Faber, P. (2011, Janeiro). *LexiCon research group*. Retrieved 2011, from *LexiCon research group*: <http://lexicon.ugr.es/pdf/faber2011.pdf>
- Faria, C. (1994, Maio). A Liberdade Humana de Aprender e Ensinar. (J. L. Ramalho, Ed.) *Ler Educação*, pp. 93-111.

- Fialho, F., & Santos, N. d. (1995). *Manual de Análise Ergonómica*.
- Figueiredo, A. D., & Silva, P. C. (1994). Base para o desenvolvimento estruturado de programas educativos. *II Congresso Ibero-americano de Informática na Educação* (pp. 120,121,122). Fundação Calouste Gulbenkian - Lisboa, Portugal: biblioteca Nacional.
- Fino, C. (2001). Vygotsky e a Zona de Desenvolvimento Proximal (ZDP): três implicações pedagógicas;. *Revista Portuguesa de Educação*., 14, 2, pp. 273-291.
- Fino, C. N. (1998). Um Software Educativo que suporte uma construção de conhecimento em interação (com pares e professores). *3º Simpósio de Investigação e Desenvolvimento de Software Educativo, edição em CD-Rom*. Universidade de Évora.
- Fino, C. N. (2003).
- Fino, C. N. (2003). Avaliar Software "Educativo". *III Conferência Internacional de Tecnologias de Informação e Comunicação na Educação*, (pp. 689-694). Braga, Universidade do Minho.
- Freire, T., & Almeida, L. S. (2008). *Metodologia da Investigação em Psicologia e Educação*. Braga: Psiquilibrios.
- Fromm, E. (1983). *Fear Of Freedom*. TAYLOR & FRANCIS LTD.
- Fuentes, C. (2003). *GABO MEMÓRIAS DA MEMÓRIA*. Lisboa: Publicações Dom Quixote, Lda.
- G. Jones, M., & Okey, J. R. (1999). User Interface Design Guidelines for Computer-Based Learning Environments. *Journal of Visual Literacy*, 19, 65-84.
- Gálvis, A. (1997). Micromundos Lúdicos Interactivos:Aspectos Criticos en su Diseno y Desarrollo. *Informática Educativa*, 10.2, pp. 191-204.
- Gama, M. C. (1998). *A teoria das Inteligências múltiplas e suas implicações para a educação*. Retrieved 2002, from Web Server em homemdemello.com.br: <http://www.homemdemello.com.br/psicologia/intelmult.html>

- Gamez, L. (1998). *LabIUtil Laboratório de Utilizabilidade*. Retrieved Novembro 2004, from Laboratório da Utizabilidade da Informática (LabIUtil): <http://www.labiutil.inf.ufsc.br/estilo/Ticese.htm>
- Gane, C., & Sarson, T. (1987). *Análise Estruturada de Sistemas*. Livros Técnicos e Científicos Editora S.A.
- Gardner, H. (1995). *Inteligências Múltiplas*. Porto Alegre, Brasil: Artes Médicas.
- Gentner, D., & Stevens, A. L. (1983). *Mental Models*. 365Brodney: Lawrence Erlbaum Associates, Inc.
- Gibran, K. (2009). *O Profeta*. Pergaminho.
- Goleman, D. (2002). *Inteligência Emocional* (11ª edição ed.). Lisboa: Temas e Debates - Actividades Editoriais, Lda.
- Gracioso, A. C. (s/data). Retrieved Janeiro 2010, from <http://webcache.googleusercontent.com/>
- Grandjean, E. (1998). *Manual de Ergonomia :Adaptando o trabalho ao homem*. Porto Alegre.
- Grandjean, E. (1998). *Manual de Ergonómia: Adaptando o trabalho ao Homem*. (J. P. Stein, Trans.) Porto Alegre, Brasil: Artes Médicas Sul LTDA.
- Green, C. D. (s/data). *PSYCHO-RESSOURCES, Le Bottin des Professionnels de la Psychologie*. Retrieved 2008, from Moteur de Recherche: <http://psychclassics.yorku.ca/Watson/views.htm>
- Griève. (2006). *Neuropsicologia em Terapia Ocupacional*. Brasil: Santos.
- Griève, J. (2000). *Neuropsicologia para Terapeutas Ocupacionais: evaluación de la percepción y cognición*. Madrid: Médica Panamericana.
- Grilo, D. (s/data).
- Guerra, M. A. (1994). *Imagen y Educacion*. Salamanca: Cervantes.
- Henriques, A. C. (2007). *Jogar e Compreender*. Instituto Piaget.

- Herbert, J., & Price, A. (1995). Métodos para Avaliação da Qualidade de Software. *XIV Congresso da Sociedade Brasileira de Computação*. Porto Alegre.
- Hiratsuka, T. P. (1996). *Contribuições da Ergonomia e do Design na Concepção de Interfaces Multimédia*. Dissertação de Mestrado em Engenharia de Produção, Florianópolis - Universidade Universal de Santa Catarina.
- Holland, J., Holyoak, K., Nisbett, R., & Thagard, P. .. (1986). *Induction: Processes of Inference, Learning and Discovery*. Cambridge: MA: MIT Press.
- Horton, W. (1994). *O Livro do Ícone*. Berkeley.
- Hulme, C., Roodenrys, S., Brown, G., & Mercer, R. (1995). *Journals Advanced Search*. Retrieved 2008, from Journals of the British Psychological Society: <http://www.bpsjournals.co.uk/>
- Hyvärinen, L. (s/Data). *Avaliação do Processamento da Informação Visual - perspectiva educativa*. Retrieved 2012, from Lea-Test Ltd: <http://www.lea-test.fi/>
- I.S.I.D. (1999). *Interface Systems International Design*. Retrieved 2001, from Interface Systems international: <http://www.isii.com/>
- Iida, I. (2005). *Ergonomia: Projecto e produção*. São Paulo: Edgard Blucher.
- Institute, P. M. (2004). *A Guide to the Management Body of Knowledge*. Estados Unidos: Project Management Institute.
- ISID. (1999). *Interface Systems International Design*. Retrieved 2001, from Interface Systems international: <http://www.isii.com/>
- J.A.McCall, P. (1977). *Factors in Software Quality* . Nova York: Griffits Air Base.
- J.Biggs, C. (2007). *Teaching for Quality Learning at University*. Maidenhead: Open University Press.
- Jakobson, R. (2003). *Linguística e Comunicação*. Diversos.
- James, W. (1890). *The Principles of Psychology* (Vol. 1). New York: Henry Holt.

- Jarufe, M. S. (2004). *Calaméo publicações*. Retrieved 2009, from Calaméo: <http://www.calameo.com/books/0001035471f4c18893523>
- John Biggs, C. T. (2007). *Teaching for Quality Learning at University*. University of New South Wales, Australia: Open University Press.
- Johnson-Laird. (1983). *Mental Models*. Cambridge: Harvard University Press.
- Johnson-Lari. (1983). *Mental Models*. Cambridge: Harvard University Press.
- Kelly, M. (n.d.). *Bloom's Taxonomy in the Classroom*. Retrieved 2008, from <http://712educators.about.com/od/testconstruction/p/bloomstaxonomy.htm>.
- kuhn, T. (1996). *A Estrutura das Revoluções Científicas*. (A. Contín, Trans.) Chicago: Universidade de Chicago.
- Laville, A. (1997). *Ergonomia*. São Paulo, Brasil: Editora Pedagógica e Universitária,Lda.
- Levine, M. W., & Shefner, J. (2000). *Fundamentals of Sensation and Perception*. New York, United States: Oxford University Press Inc.
- Lévy, P. (1994). *As tecnologias da Inteligência*. Lisboa: Instituto Piaget.
- Lida, I. (2005). *Ergonomia Projeto e Producao*. Edgard Blucher.
- Lida, I. (2005). *Ergonomia: Projecto e produção*. São Paulo: Edgard Blucher.
- Lopes, I. M. (1991). *A imagem na Comunicação Pedagógica: Estudo Exploratório numa população do ensino secundário*. Coimbra: Faculdade de Psicologia e Ciências da Educação da Universidade.
- MacCall, J., Richards, P., & Walters, G. (1997). *Factors in Software Quality*. Springfield.
- MacFarland, R. (1995). *Technological Horizons in Education Journal*. Retrieved 2009, from The Journal - Transforming Education Through Technology: <http://thejournal.com>
- Malone, T. (1981). Towards a theory of intrinsically motivating instruction. *Cognitive Science*, 333-369.

- Manns, T., & Coleman, M. (1988). *Software Quality Assurance (Computer Science Series)*. Palgrave Macmillan.
- Marcelino, M., & Mendes, T. (1994). *Estratégias e ferramentas para a construção de programas educativos de simulação*.
- Marois, R., & Todd, J. J. (2011). *Publications*. Retrieved 2011, from Human Information Processing Laboratory at Vanderbilt University: <http://www.psy.vanderbilt.edu>
- Martins, G. V. (2003). "*Roleta Matemática*" Um módulo da aplicação "A Magia dos Números" para o ensino do Mínimo Múltiplo Comum e Máximo Divisor Comum. Dissertação no âmbito do mestrado em Educação Multimédia, Faculdade de Ciências do Porto, Porto.
- Marzano, J. R., & Kendall, J. S. (2000). *Designing a New Taxonomy of Education Objectives*. Thousand Oaks, CA 91320: Corwin Press.
- Marzano, J. R., & Kendall, J. S. (2000). *Designing a New Taxonomy of Education Objectives*. Thousand Oaks, CA 91320: Corwin Press.
- Marzano, R. J., & Kendall, J. S. (2008). *Designing & Assessing Educational Objectives, Applying the New Taxonomy*. Thousand Oaks, California 91320: Corwin Press.
- Marzano, R. J., & Kendall, J. S. (2008). *Designing & Assessing Educational Objectives, Applying the New Taxonomy*. Thousand Oaks, California 91320: Corwin Press.
- Mayer, R. E. (2001). *Multi-Media Learning*. California: Cambridge University Press.
- Mehta, R., & Zhu, R. (2009). *Science Journals*. Retrieved 2010, from Science: <http://www.sciencemag.org/content/suppl/2009/02/05/1169144.DC1/Mehta.SOM.pdf>
- Merriam-Webster*. (n.d.). Retrieved 2011, from Merriam-Webster Online's Dictionary: <http://www.merriam-webster.com/dictionary/context>
- Miguel, A. (2006). *Gestão de Projectos de Software*. Lisboa: FCA - Editora de Informática.

- Miller, G. A. (1956, Maio). *Clássicos da História da Psicologia Miller*. (T. C. York University, Ed.)
Retrieved Março 2006, from Clássicos da História da Psicologia:
<http://psychclassics.yorku.ca/Miller/>
- Minnken, I., Stenseth, B., & VaVik, L. (1988). *Educational Software*. Norway: Ultima Gruppen A/S.
- Moderno, A. (1992). A Comunicação Audiovisual no Processo Didático. No Ensino e na Formação Profissional. Departamento de Didáctica e Tecnologia Educativa da Universidade de Aveiro.
- Molinari, L. (s/data). *slideshare.net*. Retrieved 2010, from SLIDESSHARE.NET:
<http://www.slideshare.net/lm7/testesfuncionaisdesoftware>
- Monet, D. (1997). *O Multimedia*. Instituto Piaget.
- Monet, D. (1997). *O Multimedia*. Instituto Piaget.
- Moraes, A. (s/data). *Ergonomia*. Retrieved 2009, from Serviços de Páginas Pessoais da PUC - Rio:
<http://wwwusers.rdc.puc-rio.br/moraergo/ergonomi.htm>
- Moreira, M. A. (2001, 07 25). *Modelos Mentais*. Retrieved 2008, from
<http://www.if.ufrgs.br/public/ensino/N3/moreira.htm>
- Morville, P., & Rosenfeld, L. (2006). *Information Architecture for the World Wide Web* (3ª ed.). O'Reilly Media.
- Myers, G. J. (2004). *The Art of Software Testing*. New Jersey: John Wiley & Sons, Inc.
- Nascimento, M. I. (2007). *Revista e-Curriculum*. Retrieved Novembro 2009, from
<http://redalyc.uaemex.mx/redalyc/pdf/766/76620208.pdf>
- Neto, A. C. (s/data). *Editorial Engenharia de Software*. (M. A. Araújo, & E. O. Spínola, Eds.)
Retrieved 2010, from DevMedia:
<http://www.google.pt/imgres?imgurl=http://www.devmedia.com.br/imagens/engsoft/artigo7/image03.jpg&imgrefurl=http://www.devmedia.com.br/articles/viewcomp.asp%3>

Fcomp%3D8035&h=300&w=566&sz=19&tbnid=6oSZ7gRgcRIHGM:&tbnh=71&tbnw=134
&prev=/images%3Fq%3Dmodelo%2

Neto, M. B. (2002). *Análise do comportamento: Behaviorismo radical, análise experimental do comportamento e análise aplicada do comportamento*. Retrieved 2008, from Portal da Psicologia: <http://www.psicologado.com/site/escolas/comportamental/diferenca-entre-a-psicologia-comportamental-e-a-analise-do-comportamento>

Nielsen, J. (1993). *Usability Engineering*. Boston: Academic Press.

Norman, D. A. (1990). *The Design Everyday Things*. Doubleday Business.

Norman, D. A. (1991). Cognitive artifacts. *Designing interaction* (pp. 17- 38). New York: Cambridge University Press.

Norman, D. A., & Draper, S. W. (1986). *User Centered System Design: New Perspectives*. Broadway: Lawrence Erlbaum Associates, Inc. Publishers.

Oliveira, C. C., Costa, J. W., & Moreira, M. (2001). *Ambientes Informatizados de aprendizagem: produção e avaliação de software educativo*. São Paulo: Papirus.

Palincsar, A. S., Brown, A. L., & Campione, J. C. (1993). First-Grade Dialogues for Knowledge Acquisition and Use. In N. M. Ellice A. Forman (Ed.), *Dynamics in Childrens' Development* (pp. 43-57). New York: Oxford University Press.

Paula, M. G., Barbosa, S. D., & Lucena, C. J. (2005, Junho). *PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO Biblioteca*. Retrieved Fevereiro 2010, from PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO: ftp://ftp.inf.puc-rio.br/pub/docs/techreports/05_23_paula.pdf

Paula, M. G., Barbosa, S. D., & Lucena, C. J. (2005, Junho). *PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO Biblioteca*. Retrieved Fevereiro 2010, from PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO: ftp://ftp.inf.puc-rio.br/pub/docs/techreports/05_23_paula.pdf

- Pedro, F., & Rodrigues, E. (2009, Setembro). *Análise de Sistemas de Informação 2009/2010*. Retrieved 2009, from Escola Superior de Gestão e Tecnologias de Beja: <http://estig.e-learning.ipbeja.pt/course/view.php?id=587>
- Piaget, J. (1973). *A epistemologia Genética*. . Editora Vozes.
- Piaget, J. (1997). *A Psicologia da Criança*. Lisboa: Edições ASA.
- Platão. (s/data). *Apologia de Sócrates , o primeiro discurso 21d*. Grécia.
- Prates, R. O., & Barbosa, S. D. (2010, Março 29). *FOSWIKI*. Retrieved Abril 2010, from Wiki DCC/UFBA:
http://wiki.dcc.ufba.br/pub/Aside/ProjetoPibicCarina/JAI2007_PratesBarbosa_EngSem.pdf
- Preece, J., Rogers, Y. R., Sharp, H., Benyon, D., Holland, S., & Carey, T. (1994). *Human-Computer Interaction*. Workingham: Addison-Wesley Publishing.
- Pressman. (1992). *Software Engineering*. Mcgraw-Hill.
- Pressman, R. (2000). *Software Engineering A Practitioner's Approach*. London: McGraw-Hill.
- Pressman, R. (2005). *Software Engineering A Practitioner's Approach* (6ª Edição ed.). New York: McGraw-Hill.
- Pressman, R. S. (2005). *Software Engineering A Practitioner's Approach* (6ª ed.). New York: McGraw-Hill.
- R.S. Pressman. (1997). *Development Strategies and Project Management, The Computer Science and Engineering Handbook*.
- Rafael, T. (2010, Fevereiro 25). *Modelo de análise sistémica do contexto no design de sistemas interactivos centrados nos utilizadores*. Retrieved Março 28, 2010, from Trocasmetinonline[09]: <http://teresarafael.wordpress.com>

- Ramos, J. (1998). *A criação e utilização de micromundos de aprendizagem como estratégia de integração do computador no currículo do ensino secundário. Dissertação de doutoramento.* Universidade de Évora.
- Reeves, T. (1994). *Dimensões Pedagógicas Eficazes de Sistemas Interactivos de Aprendizagem.* Mimeo.
- Reeves, T. (1994). *Systematic Evaluation Procedures for Interactive Multimedia for Education and Training. Multimedia computing: preparing for the 21 st century .* Harrisburg, PA: Idea Group.
- Regojo, C. (2011, Janeiro 19). *Coaching Sistémico e Constelações Organizacionais.* Retrieved Janeiro 2011, from Associação dos estudantes da faculdade de Ciências de Lisboa - AEFL: <http://aefcl.fc.ul.pt/>
- Reis, G. A. (2007). *Centrado a Arquitectura de Informação no usuário.* Retrieved 2010, from Arquitectura de Informação, usabilidade e Webdesign: http://www.guilhermo.com/mestrado/Guilhermo_Reis-Centrando_a_Arquitetura_de_Informacao_no_usuario.pdf
- Ribeiro, C. R. (2006). *RobôCarochinha: Um Estudo Qualitativo sobre a Robótica Educativa no 1º ciclo do Ensino Básico.* Mestrado em Educação, ecnologia Educativa, Instituto de Educação e Psicologia, Universidade do Minho, Braga.
- Ribeiro, N. (2007). *Multimédia e Tecnologias Interactivas.* Lisboa: FCA - Editora Informática.
- Rivlin, C., Lewis, R., & Davies-Cooper, R. (1990). *Guidelines for screen design.* Blackwell Scientific Publications.
- Robert J. Marzano, J. S. (2000). *Designing a New Taxonomy of Education Objectives.* . Thousand Oaks.
- Robert J. Marzano, J. S. (2007). *The New Taxonomy of Education Objectives.* Corwin Press.
- Robert J. Marzano, J. S. (2007). *The New Taxonomy of Education Objectives.* Thousand Oaks: Corwin Press.

- Rodrigues, E. (2002). A Contribution for the Evaluation of Educational Interactive Software. *Information Society and Education: Monitoring a Revolution Proceedings of ICTE2002. I*, pp. 6-10. Badajoz: Junta de Extremadura Consejería de Educacion Ciencia y Tecnologia.
- Rodrigues, E. (2002). A Contribution for the Evaluation of Educational Interactive Software. *Information Society and Education: Monitoring a Revolution Proceedings of ICTE2002* (pp. 6-11). Badajoz: Junta de Extremadura Consejería de Educacion Ciencia y Tecnologia.
- Rodrigues, E. (2009, Outubro 2). *Elearning - Análise de Sistemas de Informação 2009-2010*. Retrieved 2009, from Escola Superior de Tecnologias e Gestão de Beja: <http://estig.e-learning.ipbeja.pt/course/>
- Rodrigues, E. S. (2002). A Contribution for the Evaluation of Educational Interactive Software. *Information Society and Education: Monitoring a Revolution Proceedings of ICTE2002. I*, pp. 6-10. Badajoz: Junta de Extremadura Consejería de Educacion Ciencia y Tecnologia.
- Rodrigues, E. S., & Moreno, J. (2006). An Approach for Interactive Educational Software in a Classroom Setting. In A. M. A. Vilas-Méndez (Ed.), *IV International Conference on Multimedia and Information and Communication Technologies* (pp. 762-767). Sevilha: Publicado por Formatex, Badajoz.
- Rodrigues, E., & Brito, I. (2009). *Bases de dados 1*. Beja.
- Rodrigues, E., & Grilo, D. (2007). *CienciaPt - A Educação, Ciência, Tecnologia e Inovação em Portugal - Revista*. Retrieved from *CienciaPt - A Educação, Ciência, Tecnologia e Inovação em Portugal*: <http://www.cienciaportugal.net/Revista/20070503novtec.pdf>
- Rodrigues, E., Grilo, D., & Rodrigues, C. (n.d.).
- Rousseau, J. J. (1966). *Émile ou De L'Éducation*. Paris: GF Flammarion.
- Rousseau, J. J. (n.d.). *Emílio ou a Educação*.
- Ruskin, J. (2007, Dezembro 24). *docs.google*. (Docs.google, Ed.) Retrieved Junho 2010, from O Portal Qualidade:
<http://docs.google.com/viewer?a=v&q=cache:FiwksO1Z0S0J:www.mbc.org.br/mbc/uplo ads/biblioteca/1198675018.3566A.pdf>

- Sagrada, B. (1963). *A Bíblia Sagrada*. (A. P. Figueiredo, Trans.) Lisboa: Vulgata latina.
- Santiago, R. M. (2000, Setembro-Dezembro). *Revista Ibero-Americana de Educación/ de Educação, La Ciencia y la Cultura*. Retrieved 2009, from Revista Ibero-Americana de Educación.
- Santos, J. B. (2000). *O Averso da Maldição do Gênesis A saga de quem não tem trabalho*. São Paulo Brasil: Annablume Editora Comunicação.
- Santos, M. (2003). *Orientações para o Design da Interface de Software Educativo Multimédia. Tese de Doutoramento*. Salamanca: Universidade de Salamanca.
- Santos, O. A. (s/data). *Clube de Gerenciamento de Projectos Artigos*. Retrieved 2010, from Clube de Gerenciamento de Projectos: http://www.clubegp.com.br/artigo/artigo_gerenciaagil.pdf
- Scapin, D., & Bastien, C. (1993). Ergonomic Criteria for the Évaluation of Human-Computer Interfaces. *INRA - Institut National de Recherche en Informatique et en Automatique, Rocquencourt, France*.
- Shneiderman, B. (1989). Reflections on authoring, editing, and managing hypertext. *The society of text: hypertext, hypermedia, and the social construction of information*. USA: MIT Press Cambridge. Retrieved 2009, from <http://dl.acm.org>
- Shneiderman, B. (1998). *Designing The User Interface Strategies for Effective Human-Computer Interaction*. Addison-Wesley.
- Silva, A., & Videira, C. (2005). *UML - Metodologias e Ferramentas CASE*.
- Silva, P. C., & Figueiredo, A. D. (1994). Base para o desenvolvimento estruturado de programas educativos. *II Congresso Ibero-americano de Informática na Educação* (pp. 117-123). Fundação Calouste Gulbenkian - Lisboa, Portugal: biblioteca Nacional.
- Software, I. 9. (n.d.).
- Sommerville, I. (2001). *Software Engineering* (6ª Edição ed.). Addison-Wesley.

- Sommerville, I., & Sawyer, P. (1997). *Requirements Engineering: A good practice guide* (1 ed.). New York: John Wiley & Sons.
- Sousa, P. J. (2008). Retrieved 2010, from http://paginas.fe.up.pt/~pjsousa/fulltext/Paulo_Sousa_EBAI_2008.pdf
- Squires, D., & McDougall, A. (2003). *Choosing and Using Educational Software: A teachers' guide*. RoutledgeFalmer.
- Stemler, L. K. (1997). Retrieved 2009, from Université de Montréal: http://www.medvet.umontreal.ca/techno/eta6785/articles/multimedia_design.pdf
- Subbotsky, E. (1998). *Hanover College Departamento de Psicologia*. Retrieved from Faculdade de Hanover-Hanover India: <http://psych.hanover.edu/vygotsky/subbot.html>
- Szabo, M., & Kanuka, H. (1999). *Issues for the Journal of Educational Multimedia and Hypermedia*. (8. 2.-4. Journal of Educational Multimedia and Hypermedia, Ed.) Retrieved 2009, from EdITLib, the Digital Library for Education & Information Technology: <http://editlib.org/p/9540>
- Tait, T. F. (1998). Retrieved 2010, from ABEPRO Associação Brasileira de Engenharia de Produção : www.abepro.org.br/biblioteca/ENEGEP1998
- Tamashiro, A. (2010, Março). *Qualidade e Testes de Software*. Retrieved Abril 2010, from <http://qabrazil.blogspot.com/>
- Taylor, R. P. (1980). *CITE Journal - Contemporary Issues in Technology and Teacher Education*. (G. Bull, Ed.) Retrieved 2009, from CITE Journal - Contemporary Issues in Technology and Teacher Education: <http://www.citejournal.org/vol3/iss2/seminal/article1.cfm>
- Teixeira, P. A. (2006, Dezembro 5). *Universidade do Minho RepositóriUM*. Retrieved 2009, from Universidade do Minho: <http://hdl.handle.net/1822/6406>
- Telles, M. (2005, Agosto). <http://www.dynamiclab.com/moodle/mod/forum/discuss.php?d=436>. Retrieved Outubro 2006, from DynamicLab Gazette: <http://www.dynamiclab.com/moodle/mod/forum/discuss.php?d=436>

- Thorndike, E. L. (1912). *Education, a first book*. New York: The Macmillan Company.
- Tomás, M. R. (2009). Retrieved 2010, from http://run.unl.pt/bitstream/10362/2003/1/WPSeries_09_2009Tomas.pdf
- Usabilidade, O. M. (2001). *As Normas Internacionais de HCI e Usabilidade*. Retrieved 2009, from Usability Net: http://www.usabilitynet.org/tools/r_international.htm
- Vieira. (1999). *Avaliação de Software Educativo: Reflexões para uma análise Criteriosa*.
- Vieira, F. M. (2001). *Avaliação de Software Educativo: Reflexões para uma Análise Criteriosa*. Retrieved 10 22, 2006, from <http://www.edutecnet.com.br/Textos/Alia/MISC/edmagali2.htm>
- Villafañe, J. (2006). *Introducción a la teoría de la imagen*. Madrid: Ediciones Pirámide.
- Vygotsky, L. S. (2007). *A formação social da mente*. Brasil: MARTINS MARTINS FONTES.
- Wasson, B. (1996). Instruction Planning and Contemporary Theories of Learning: Is this a Self-Contradiction? *European Conference on Artificial Intelligence in Education Yildiz, R. Atkins Evaluating Multimedia applications in Computers and Education* (pp. 133-139). Istambul: P.Brna, A.Paiva &J.Self (Eds.).
- Wasson, B. (1996). Instructional Planning and Contemporary Theories of Learning: Is this a Self-Contradiction? *European Conference on Artificial Intelligence in Education*. P.Brna, A. Paiva & J.Self(Eds).
- Wasson, B. (n.d.). Instructional Planning and Contemporary Theories of Learning: Is this a Self-Contradiction?
- Watson, J. (1913). *Psychology as the behaviorist views it*. *Psychological Review*.
- Webber, C., Boff, E., & Bono, F. (2009). Retrieved 2010, from <http://docs.google.com/view>
- Weiner, B. (1990). History of motivational research in education. *Journal of Educational Psychology*, 616-620.

- Whitfield, D., & Langford, J. (2001). *The Library*. Retrieved Janeiro 2010, from Institute Of Ergonomics & Human Factors: <http://www.ergonomics.org.uk/what-ergonomics>
- Wilson, L. O. (2006). <http://www.uwsp.edu/education/lwilson/curric/newtaxonomy.htm>.
- Wisner, A. (1994). *A Inteligência no Trabalho, Textos seleccionados de ergonomia*. São Paulo: Editora da UNESP.
- Woleck, A. (2002). Retrieved Julho 2009, from Instituto Catarinense de pós-Graduação: <http://www.ea.ufrgs.br/graduacao/disciplinas/adm01156/CONCEITOSDETRABALHOEMPREGO.pdf>
- Yourdon, E. (1989). *Modern Structured Analysis*. New Jersey, U.S.A: Prentice-Hall.
- Yourdon, E. (1989). *Modern Structured Analysis*. Prentice-Hall, Inc.
- Zhang, P. (2004). Retrieved 2006, from EISEVIER: http://www.elsevier.com/wps/find/homepage.cws_home

Apêndices

APÊNDICE A

IDENTIFICAÇÃO DO SOFTWARE

Nome

Língua:

Versão:

Temas de exploração:

Data de edição:

Nível de ensino:

Desenvolvido por:

CARACTERÍSTICAS DO HARDWARE

Equipamento Requerido:

CD-ROM (S/N) Placa Gráfica (S/N)

Periféricos:

Memória RAM: (Mb) Capacidade de disco: (Mb)

REQUISITOS TÉCNICOS

1. *Procedimento de instalação:*

2. *Indicação do ambiente (Windows/MAC) ou Sistema Operativo*

INTERFACE

1. *Formas de apresentação da informação (S/N)*

Cor agradável Som Vídeo Animação
Texto Imagens

OBSERVAÇÕES:

2. *Informação clara sobre a manipulação da aplicação (S/N):*

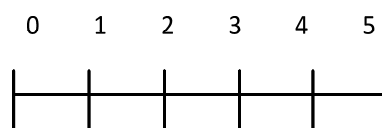
Imagens significativas e claras Botões e links

OBSERVAÇÕES:

INTERACTIVIDADE

1. *Tipo de feedback . Requer decisões e/ou respostas da parte do utilizador, aumentando assim, o grau de participação? (Sim/Não. Descreva)*

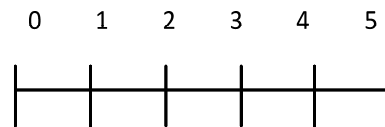
Estabelece-se uma valorização que vai desde o -1 (muito mau), 5 (muito bom) e 0, se não está representada



2. Estrutura da comunicação. (Comunicação passiva, activa, experimentação, criativa)

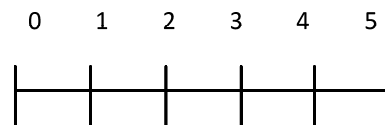
3. Grau de participação e controlo por parte do utilizador:

Estabelece-se uma valorização que vai desde o -1 (muito mau), 5 (muito bom) e 0, se não está representada.



4. Grau de controlo por parte do utilizador:

Estabelece-se uma ligação que vai desde o - 1 (muito difícil), 5 (muito fácil)



FERRAMENTAS DE EXPLORAÇÃO

1. Formas possíveis de fornecer ajuda, com o objectivo de diminuir a possibilidade do utilizador se desorientar. Exemplo: existência de botões de retorno a ação anterior, informação geral para a manipulação do software, etc.

2. Indique os mecanismos para orientar na manipulação da aplicação (Navegação). Utilização de botões, menus e/ou links

3. Registos de notas e apontamentos (S/N)

OBSERVAÇÕES:

MATERIAIS DE APOIO

1. Clareza de instruções

1 2 3

confusas claras

2. Quantidade de informação

1 2 3

escassa suficiente abundante

3. Inclui apoio à utilização pedagógica do programa?

Sim Não

Se sim, qual?

4. Outros materiais de apoio:

ASPECTOS PEDAGÓGICOS

1. Alunos abrangidos/alvo (Colocar as características importantes para definir os alunos)

2. Objetivos da aprendizagem.

3. Adaptabilidade à problemática e à capacidade do aluno. (Sim/Não Descreva)

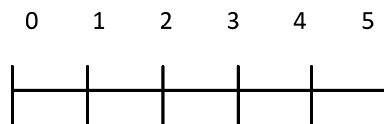
Visuais Auditivos Motor

Cognitivo Comunicação/linguagem

Observações:

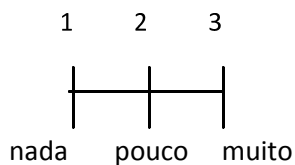
4. Interatividade e motivação em relação ao aluno. (Sim/Não Descreva)

Estabelece-se uma valorização que vai desde o -1 (muito mau), 5 (muito bom) e 0, se não está representada



5. Estimula a criatividade

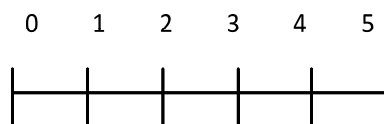
Estabelece-se uma valorização que vai desde o -1 (nada), 3 (muito).



6. Permite uma aprendizagem autodidática. (Sim/Não Descreva)

7. A resposta exige reflexão por parte do aluno. (Sim/Não Descreva)

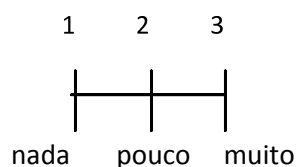
Estabelece-se uma valorização que vai desde o -1 (muito mau), 5 (muito bom) e 0, se não está representada.



8. Permite avaliar a evolução do aluno. (Sim/Não Descreva)

9. Software permite mudar de um passo mais avançado para um passo mais simples com o objetivo de superar as dificuldades no processo de aprendizagem. (Sim/Não Descreva)

Estabelece-se uma valorização que vai desde o -1 (nada), 3 (muito).



10. É possível utilizá-lo em grupos. (Sim/Não)

DESCRIÇÃO SUCINTA DA APLICAÇÃO

Título da ferramenta a avaliar

Atividade	Descrição	Dispositivos de Acesso	Objectivos

APÊNDICE B

IDENTIFICAÇÃO DO SOFTWARE

Nome

Língua:

Versão:

Temas de exploração:

Data de edição:

Nível de ensino:

Desenvolvido por:

CARACTERÍSTICAS DO HARDWARE

Equipamento Requerido:

CD-ROM (S/N)

Placa Gráfica (S/N)

Periféricos:

Memória RAM:

(Mb)

Capacidade de disco:

(Mb)

REQUISITOS TÉCNICOS

1. Procedimento de instalação:

2. Indicação do ambiente(Windows/MAC) ou Sistema Operativo

INTERFACE

1. Formas de apresentação da informação (S/N)

Cor agradável Som Vídeo Animação
Texto Imagens

OBSERVAÇÕES:

2. Informação clara sobre a manipulação da aplicação (S/N):

Imagens significativas e claras Imagens estáticas

Imagens Dinâmicas Botões e links

OBSERVAÇÕES:

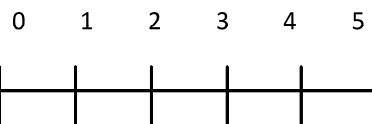
INTERATIVIDADE

1. Tipo de feedback

Funções do texto

Estabelece-se uma valorização que vai desde - 1 (muito mau), 5 (muito bom) e 0 se não está representada.

Função Apelativa



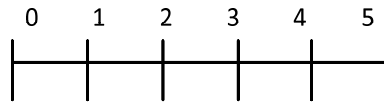
Função Interativa



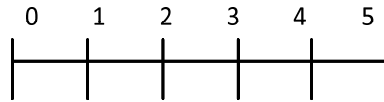
Funções do Som

Estabelece-se uma valorização que vai desde - 1 (muito mau), 5 (muito bom) e 0 se não está representada.

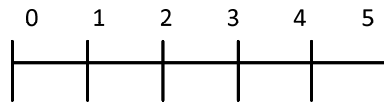
Função Informativa (existe uma informação dos aspetos mais importantes)



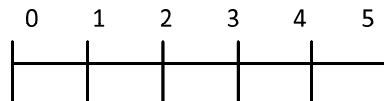
Função Apelativa (persuadir o aluno a adotar uma determinada ação)



Poética (o software se apoia na mensagem, nos sons para transmitir sugestões e sentimentos)



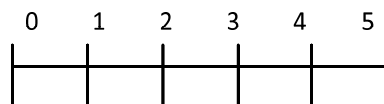
Indagadora (o aluno tem de responder a uma questão formulada, que pode ser dada pela ação clicar)



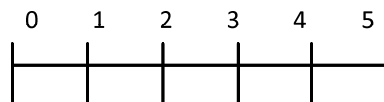
Funções da Imagem

Estabelece-se uma valorização que vai desde - 1 (muito mau), 5 (muito bom) e 0 se não está representada.

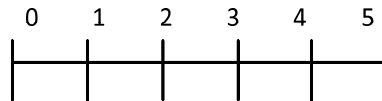
Absoluto (Convencer; Informar)



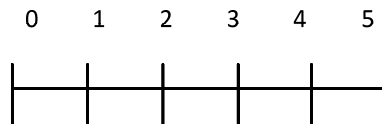
De colocação Estratégica (Focar a atenção; Encaminhar; Realçar a informação)



De reação do Utilizador (Deslocar para outra parte do programa)



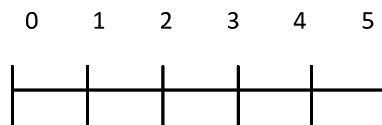
Estabelece-se uma valorização que vai desde -1 (muito mau), 5 (muito bom) e 0, se não está representada.



2. Estrutura da comunicação. (Comunicação passiva, ativa, experimentação, criativa)

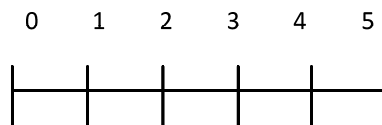
3. Grau de participação e controlo por parte do utilizador:

Estabelece-se uma valorização que vai desde - 1 (muito mau), 5 (muito bom) e 0 se não está representada.



4. Grau de controlo por parte do utilizador:

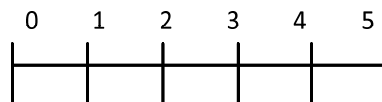
Estabelece-se uma ligação que vai desde - 1 (muito difícil), 5 (muito fácil)



FERRAMENTAS DE EXPLORAÇÃO E USABILIDADE

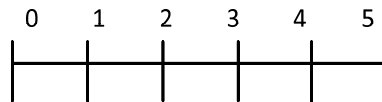
1. O equilíbrio visual é mantido em toda a aplicação.

Estabelece-se uma valorização que vai desde - 1 (muito mau), 5 (muito bom) e 0 se não está representada.



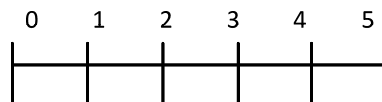
2. O ponto focal é mantido durante toda aplicação.

Estabelece-se uma valorização que vai desde - 1 (muito mau), 5 (muito bom) e 0 se não está representada.



3. A unicidade versus harmonia existe em toda a aplicação

Estabelece-se uma valorização que vai desde - 1 (muito mau), 5 (muito bom) e 0 se não está representada.



4. Formas possíveis de fornecer ajuda, com o objetivo de diminuir a possibilidade do utilizador se desorientar. Exemplo: existência de botões de retorno a ação anterior, informação geral para a manipulação do software, etc.

5. Indique os mecanismos para orientar na manipulação da aplicação (Navegação). Utilização de botões, ícones, menus e/ou links.

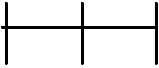
6. Registos de notas e apontamentos (S/N)

OBSERVAÇÕES:

MATERIAIS DE APOIO

7. Clareza de instruções

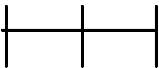
1 2 3



confusas claras

8. Quantidade de informação

1 2 3



escassa suficiente abundante

9. Inclui apoio à utilização pedagógica do programa?

Sim Não

Se sim, qual?

10. Outros materiais de apoio:

ASPECTOS PEDAGÓGICOS

1. *Alunos abrangidos/alvo (Colocar as características importantes para definir os alunos)*

2. *Objetivos da aprendizagem.*

3. *Adaptabilidade à problemática e à capacidade do aluno. (Sim/Não Descreva)*

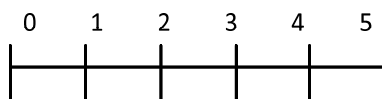
Visuais Auditivos Motor

Cognitivo Comunicação/linguagem

Observações:

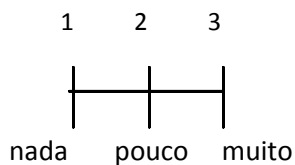
4. *Interatividade e motivação em relação ao aluno. (Sim/Não Descreva)*

Estabelece-se uma valorização que vai desde - 1 (muito mau), 5 (muito bom) e 0 se não está representada.



5. *Estimula a criatividade*

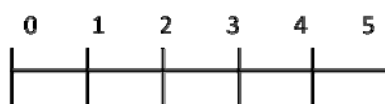
Estabelece-se uma valorização que vai desde o -1 (nada), 3 (muito).



6. *Permite uma aprendizagem autodidática. (Sim/Não Descreva)*

7. A resposta exige reflexão por parte do aluno. (Sim/Não Descreva)

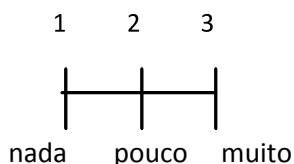
Estabelece-se uma valorização que vai desde - 1 (muito mau), 5 (muito bom) e 0 se não está representada.



8. Permite avaliar a evolução do aluno. (Sim/Não Descreva)

9. Software permite mudar de um passo mais avançado para um passo mais simples com o objetivo de superar as dificuldades no processo de aprendizagem. (Sim/Não Descreva)

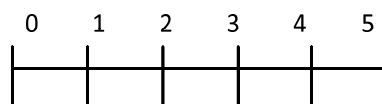
Estabelece-se uma valorização que vai desde - 1 (nada) até ao 3 (muito)



10. Funções Didáticas do Som (a nível pedagógico)

Estabelece-se uma valorização que vai desde - 1 (muito mau), 5 (muito bom) e 0 se não está representada.

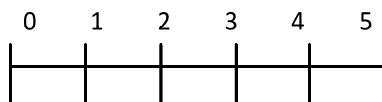
Função Poética (a aplicação apoia-se nos sons para transmitir sugestões e sentimentos)



11. Funções Didáticas da Imagem (a nível pedagógico)

Estabelece-se uma valorização que vai desde - 1 (muito mau), 5 (muito bom) e 0 se não está representada.

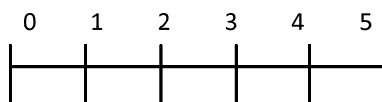
Catalisadora de Experiências (a imagem traduz a realidade de forma a facilitar a sua compreensão)



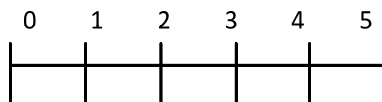
Informativa (a imagem assume o papel didático)



Estética (a imagem é usada para dar equilíbrio e cor a um espaço)



Identificação/Associação (a imagem é utilizada para reconhecer características noutra imagem)



12. É possível utilizá-lo em grupos. (Sim/Não)

DESCRIÇÃO SUCINTA DA APLICAÇÃO

Título da ferramenta a avaliar

Atividade	Descrição	Dispositivos de Acesso	Objetivos

--	--	--	--

Grelha de Avaliação de Software [.....]

APÊNDICE C

Função Didática do Som	
Função Informativa	
Professora A	5
Professora B	5
Professora C	5
Professora Apoio D	5
Professora Apoio E	5
Média	5

Função Didática do Som	
Função Poética	
Professora A	5
Professora B	5
Professora C	5
Professora Apoio D	5
Professora Apoio E	5
Média	5

Função Didática do Som	
Função Indagadora	
Professora A	4
Professora B	5
Professora C	5
Professora Apoio D	5
Professora Apoio E	5
Média	4,8

Função Didática do Som	
Função Apelativa	
Professora A	5
Professora B	5
Professora C	5
Professora Apoio D	5
Professora Apoio E	5
Média	5

Resultado das Funções Didáticas do Som no Software "Palavras e Palavras"

Anexos

Anexo A - As Inteligências Múltiplas, adaptado (Martins, 2003)

Inteligência	Descrição	Relação c/ Outras	Exemplos Pessoais	Habilidades	Agentes
LINGÜÍSTICA (Hemisfério esquerdo, Vocabulário: lobo frontal acima do lobo temporal, Linguagem: lobo temporal)	Capacidade de processar rapidamente mensagens linguísticas, de ordenar palavras e dar sentido lúcido às mensagens.	Relaciona-se com todas as demais, particularmente com a lógica-matemática e a cinestésica corporal.	Shakespeare, Dante, Cervantes, Guimarães Rosa, Clarisse, inspetor, escritores radialistas, advogados e principalmente poetas.	Descrever, Narrar, Observar, Comparar, Relatar, Avaliar, Concluir, Sintetizar.	Pais, Avós, Professores, Amigos.
LÓGICO-MATEMÁTICA (Lobos frontais e parietais esquerdos)	Facilidade para o cálculo e para a percepção da geometria espacial. Prazer específico em resolver problemas embutidos em palavras cruzadas, charadas ou problemas lógicos como jogos de gamão e xadrez.	Inteligência linguística, espacial, cinestésica corporal, e principalmente a musical.	Euclides, Pitágoras, Newton, Bertrand, Russell, Einstein, engenheiros, arquitetos e mestre-de-obras.	Enumerar, Seriar, Deduzir, Medir, Comparar, Concluir, Provar.	Pais Professores treinados.
ESPACIAL (Hemisfério direito)	Capacidade de perceber formas e objetos mesmo quando apresentados em ângulos não usuais, capacidade de perceber o mundo visual com precisão de efetuar transformações sobre as percepções de imaginar movimento ou deslocamento interno entre as partes de uma configuração, de recriar aspectos da experiência visual e de perceber as direções no espaço concreto e abstrato.	Com todas as demais especialmente a linguística, a musical e a cinestésica corporal.	Isaac Assimov, Karl Marx, Picasso, Darwin, Dalton, escritores de ficção, exploradores, geógrafos, marinheiros.	Localizar no espaço, Localizar no tempo, observar, comparar, Deduzir, Relatar, Combinar, Transferir.	Pais, Professores Alfabetizados Linguísticos Cartográficos.
CORPORAL CINESTÉSICA (Hemisfério esquerdo)	Capacidade de usar o próprio corpo de maneira diferenciada e hábil para propósitos expressivos. Capacidade de trabalhar com objetos, tanto os que envolvem motricidade específica quanto os que exploram uso integral do corpo.	Principalmente as inteligências linguísticas, espacial e pictóricas.	Nijinsky, Nureyev, Pelé, Eusébio Garrincha, Magic Johnson, mímicos, bailarinos, atletas e também concertistas, cirurgiões e muitos outros.	Comparar, Medir, Relatar, Transferir, Demonstrar, Interagir, Sintetizar, Interpretar, Classificar.	Instrutores de dança e desportos, Pais, Professores.

Anexo A1 - (continuação - As Inteligências Múltiplas, adaptado (Martins, 2003))

Inteligência	Descrição	Relação c/ Outras	Exemplos Pessoais	Habilidades	Agentes
PICTÓRICA (<i>Hemisfério direito</i>)	Capacidade de expressão por traço, desenho ou caricatura. Sensibilidade para dar movimento e beleza a desenhos e pinturas, autonomia para captar e retransmitir as cores da natureza, movimentar-se com facilidade em diferentes níveis da computação gráfica.	Inteligência linguística, espacial, cinestésica corporal, mas principalmente a musical.	Giotto Botticelli, Rafael, Leonardo da Vinci, Miguel Ângelo, Portinari, Tarsila do Amaral, Bill Anderson e cartoonistas, pintores, ilustradores, especialistas em computação gráfica.	Observar, Refletir, Reproduzir, Transferir, Criticar, Concluir.	Pais, Professores preparados.
NATURALISTA (<i>Hemisfério direito, presumivelmente</i>)	Atração pelo mundo natural e sensibilidade em relação a ele, capacidade de identificação da linguagem natural e capacidade de êxtase diante da paisagem humanizada ou não.	Com todas as demais, especificamente com as inteligências linguística, musical e espacial.	Darwin, Humboldt, La Condamine, Mendel, Ruschi, Noel Nutels, Villas-Boas, Burle Marx, naturalistas, botânicos, geógrafos, paisagistas.	Relatar, Demonstrar, Selecionar, Levantar, hipótese, Classificar, Rever.	Avós, Pais e Professores.
MUSICAL (<i>Hemisfério direito, lobo frontal</i>)	Facilidade para identificar sons diferentes. Reconhecer sons naturais e, na música, perceber a distinção entre tom, melodia, ritmo, timbre e frequência. Isolar sons em agrupamentos musicais.	Mais intensamente com a lógico-matemática e com as pictóricas e cinestésica corporal.	Beethoven, Chopin, Brahms, Schubert, Tchaikovsky, Carlos Gomes, Villa Lobos, Tom Jobim, Cartola, Caetano Veloso, Paulinho do Viola, compositores, poetas, naturalistas.	Observar, Identificar, Relatar, Reproduzir, Conceituar, Combinar.	Pais, Avós, Professores devidamente sensibilizados.
PESSOAIS <i>Interpessoal e Intrapessoal</i> (<i>Lobos frontais</i>)	Interpessoal: capacidade de perceber e compreender outras pessoas, descobrir as forças que as motivam e sentir grande empatia pelo outro indivíduo. Intrapessoal: capacidade de autoestima, auto motivação, de formação de um modelo coerente e verídico de si mesmo e do uso desse modelo par operacionalizar a construção da felicidade pessoal e social.	As inteligências pessoais interagem e relacionam-se com todas as demais, particularmente com a linguística, a naturalista e a cinestésica corporal.	Proust, Gandhi, Freud, Anne Sullivan, Adler, Joana D'Arc, Martin Luther King, Antônio Conselheiro, Padre Cícero, pessoas reconhecidas como "carismáticas", políticos, líderes religiosos, psicoterapeutas, psicólogos, assistentes sociais.	Interagir, Perceber, Relacionar-se com empatia. Apresentar autoestima e autoconhecimento. Ser ético.	Pais, Psicólogos, Professores devidamente treinados.

Anexo B - Atividades em Contexto de Sala de Aula, (Antunes C. , Revista Educação, 2005)

Inteligências	Algumas Atividades	Materiais de Ensino	Ação Docente
ESPACIAL	Atividades artísticas, apresentações visuais, metáforas, visualização e mapas conceituais. Concursos fotográficos, metáforas por meio de imagens, símbolos gráficos diversos.	Colagens, gráficos, frisas do tempo, mapas, massa de modelagem, argila, lápis de cor, recursos tácteis. Coleção de fotos.	Explorar o uso de linguagens alternativas, solicitar a transferência de textos para desenhos, gráficos, quadros de síntese.
CINESTÉSICA CORPORAL	Teatro, dança, mímica, exercícios de relaxamento, atividades diversas que envolvam o uso do corpo.	Instrumentos de montagem, tampinhas, blocos, equipamentos desportivos, recursos manipuláveis, peças de LEGO. Mapas corporais.	Solicitar o uso de movimentos do corpo para expressar conhecimentos de disciplinas descritivas. Exercícios sobre consciência física. Propostas sobre cozinhar, costurar, jardinar, realidade virtual.
LÓGICO MATEMÁTICA	Desafios, problemas, enigmas, atividades científicas de experimentação, desafios numéricos, pensamentos críticos. Concursos sobre resolução de problemas lógicos, criação de códigos, linguagens de computação.	Calculadoras, ábacos, jogos matemáticos, desafios que explorem a grandeza, proporções, perspectivas. Uso de escalas diversas. Computador e, quando possível, computador fora de uso para desmontagem e análise.	Empenhar-se em desenvolver a capacidade de expressar pensamentos através de gráficos, busca de proporções, médias, grandezas e outros elementos lógicos
NATURALISTA	Aprendizagem rítmica, apresentação de corais abordando temas escolares, seleção e criação de músicas envolvendo os conteúdos disciplinares. Dramatizações e Concertos. Visitas a apresentações musicais. Vinculação de conceitos à música.	Aquários, hortas coletivas, pequenos museus ou coleções naturalistas.	Proposição de desafios que envolvam conhecimento de animais e plantas, transposição de temas para um enfoque naturalista, organização de diários de campo e registros de atividades ao ar livre.
SONORA	Aprendizagem rítmica, apresentação de corais abordando temas escolares, seleção e criação de músicas envolvendo os conteúdos disciplinares. Dramatizações e Concertos. Visitas a apresentações musicais. Vinculação de conceitos à música.	Gravador, coleção de fitas, instrumentos musicais, coleção de CDs. Aparelhos de reprodução sonora.	Sugestão para a criação de paródias, organização de grupos para apresentação de temas escolares com ritmos diversos e uso de fundo musical.

Anexo B1 - Atividades em Contexto de Sala de Aula (cont), (Antunes C. , Revista Educação, 2005)

Inteligências	Algumas Atividades	Materiais de Ensino	Ação Docente
LINGÜÍSTICA	Exposições, debates, organização de telejornais, jornais impressos ou murais, jogos de palavras e atividades que explorem a narração, leitura ou redação. Organização de grupos para debates sobre filmes assistidos, clubes literários, concursos linguísticos	Livros diversos, dicionários de vários tipos, coleção de jornais e revistas, portfólios sobre temas, concurso de redação, trovas e outros	Estímulos para pesquisas bibliográficas, exploração de diferentes habilidades operatórias como sintetizar, analisar, relatar, descrever e outras, desafios sobre interpretação de textos, concursos de manchetes, trovas e poemas para expressar diferentes conteúdos.
INTRAPESSOAL	Orientação individual, exploração de pesquisas sobre a autoestima, aceitação de produções individualizadas, oportunidade de opções para manifestações diferenciadas do conhecimento adquirido	Recursos diversos para autoavaliação, organização de portfólios, diários, materiais diversificados sobre projeto, orientação pessoal de pesquisas	Exploração de atividades que envolvam a significação dos fatos apreendidos no uso diário, ajuda para a contextualização do apreendido no cotidiano vivido.
INTERPESSOAL	Trabalhos em grupo, organização de micro-cooperativas e projetos de apoio comunitário, organização de campanhas filantrópicas, reuniões sociais. Propostas para atividades compartilhadas, exercícios de simulações	Jogos coletivos, relação de atividades sociais, arquivo de projetos de ação comunitária, fontes de apoio a ações coletivas	Estímulo a cooperação, proposta de campanhas diversas
Pictórica	Concurso de fotografias, desenho, caricatura, escultura e/ou outras formas artísticas	Jogos de palavras com elementos pictóricos, desenhos, exercícios que exercitem a montagem e a interpretação, telas e cores	A aula deve ser cativante, promover a sensibilidade para o movimento, para as cores, para a beleza, desenvolver autonomia

Anexo C - Níveis e Objetivos Gerais da Nova Taxonomia, adaptado (Marzano & Kendall, 2008)

Novo Nível Taxonómico	Operação	Objetivos Gerais
NÍVEL 6 SISTEMA DE AUTO CONHECIMENTO	Importância da Examinação	O aluno terá capacidade para identificar a importância da informação no processo mental ou processo psicomotor, para si próprio e a razão que está por detrás desta percepção.
	Eficácia da Examinação	O aluno terá capacidade de identificar se acredita nas suas capacidades de melhorar competências ou entendimento relativo à informação, processo mental ou processo psicomotor e as razões que estão por detrás desta percepção.
	Examinando respostas emocionais	O aluno terá capacidade de identificar as suas respostas (reações) emocionais à informação, processo mental ou processo psicomotor e as razões para estas respostas.
	Motivação Examinadora	O aluno terá capacidade de identificar o seu nível global de motivação para melhorar competências ou entendimento relativo à informação, ao processo mental ou ao processo psicomotor e as razões que estão por detrás desta motivação.
NÍVEL 5 METACOGNIÇÃO	Especificar Objetivos	O aluno terá a capacidade de estabelecer um objetivo relativo à informação, processo mental ou processo psicomotor e um plano para atingir esse objetivo.
	Processo de Monitorização	O aluno terá capacidade de dirigir o progresso no sentido de atingir um objetivo específico relativo à informação, processo mental ou processo psicomotor.
	Clareza da Monitorização	O aluno terá capacidade para determinar até que ponto lhe é clara a informação, o processo mental ou o processo psicomotor.
	Precisão da Monitorização	O aluno terá capacidade para determinar até que ponto é preciso acerca da informação, processo mental ou processo psicomotor.
NÍVEL 4 UTILIZAÇÃO DO CONHECIMENTO	Tomada de decisões	O aluno será capaz de usar a informação, processo mental ou processo psicomotor para tomar decisões em geral ou para tomar decisões acerca do uso da informação, do processo mental ou processo psicomotor.
	Resolução de problemas	O aluno será capaz de usar a informação, o processo mental ou o processo psicomotor para resolver problemas em geral ou para resolver problemas acerca da informação, do processo mental ou processo psicomotor.
	Experimentação	O aluno será capaz de usar a informação, o processo mental ou o processo psicomotor para gerar e testar hipóteses em geral ou gerar e testar hipóteses acerca da informação, processo mental ou processo psicomotor.
	Investigação	O aluno será capaz de usar a informação, o processo mental ou o processo psicomotor para conduzir investigações em geral ou para conduzir investigações acerca da informação, processo mental ou processo psicomotor.

Anexo C1 - Níveis e Objetivos Gerais da Nova Taxonomia, (cont.) adaptado (Marzano & Kendall, 2008)

Novo Nível Taxonómico	Operação	Objetivos Gerais
NÍVEL 3 ANÁLISE	Equiparar	O aluno será de identificar semelhanças e diferenças importantes relativas à informação, ao processo mental ou ao processo psicomotor.
	Classificar	O aluno será capaz de identificar categorias superiores e subordinadas relativas à informação, processamento ou processo psicomotor.
	Analisar erros	O aluno será capaz de identificar erros na apresentação ou uso da informação, processo mental ou processo psicomotor.
	Generalizar	O aluno será capaz de construir novas generalizações ou princípios baseados na informação, processo mental ou processo psicomotor.
	Especificar	O aluno será capaz de identificar consequências lógicas da informação, processo mental ou processo psicomotor.
NÍVEL 2 COMPREENSÃO	Integrar	O aluno será capaz de identificar a estrutura básica da informação, do processo mental ou do processo psicomotor e as características críticas em oposição às não críticas.
	Simbolizar	O aluno será capaz de construir uma representação simbólica e precisa da informação, do processo mental ou do processo psicomotor diferenciando os elementos críticos e não críticos.
NÍVEL 1 RECUPERAÇÃO	Reconhecer	O aluno será capaz de validar afirmações corretas acerca de aspetos da informação, mas não tem necessariamente que compreender a estrutura do conhecimento ou diferenciar componentes críticos ou não críticos.
	Recordar	O aluno será capaz de validar declarações corretas sobre os recursos de informação, mas não necessariamente compreender a estrutura do conhecimento ou diferenciar componentes críticos e não críticos.
	Executar	O aluno será capaz de realizar um procedimento sem erros significativos, mas não tem necessariamente que perceber como e porque é que o procedimento funciona.

Anexo D - Taxonomia SOLO – verbos associados a cada SOLO

SOLO 1	SOLO 2	SOLO 3	SOLO 4	SOLO 5
<i>Pre-Structural Level</i>	<i>Uni-Structural Level</i>	<i>Multi-Structural Level</i>	<i>Relational Level</i>	<i>Extended Abstract Level</i>
Não utilizado para definir os resultados da aprendizagem	Anotar	Calcular	Analisar	Colocar hipóteses
	Citar	Classificar	Aplicar	Compor
	Contar	Descrever	Argumentar	Criar
	Definir	Discutir	Caracterizar	Defender um ponto de vista original
	Encontrar	Distinguir	Comparar	Generalizar
	Enunciar	Enumerar	Concluir	Gerar
	Escrever	Ilustrar	Construir	Inventar
	Fazer			
	Corresponder	Listar	Contrastar	Originar
				Provar a partir de princípios fundamentais
	Identificar	Narrar	Criticar	Refletir
	Imitar	Selecionar	Debater	Resolver a partir de princípios fundamentais
			Defender um ponto de vista	Teorizar
	Lembrar	Separar	Diferenciar	
	Memorizar	Sequenciar	Elaborar planos	
	Nomear	Verificar	Examinar	
	Ordenar		Explicar	
	Recitar		Implementar	
	Reconhecer		Integrar	
			Organizar	
			Parafrasear	
			Prever	
			Relacionar	
			Resolver um problema	
			Resumir (preciso)	
			Rever e reescrever	
			Traduzir	
			Transferir	

Anexo E - Quadro I - Requisitos Técnicos (Costa F. A., 1999)

Grupo I	1. Requisitos Técnicos 1.1 Equipamento Requerido 1.2 Informação Técnica Sobre o Software 1.3 Informação Sobre o Processo de Instalação
----------------	--

Anexo F - Quadro II - Conteúdo e Aspetos Pedagógicos da Aplicação, (Costa F. A., 1999)

Grupo II	2. Conteúdo da Aplicação 2.1 Conteúdo Científico 2.2 Conteúdo Sociocultural, Étnico e Ideológico 2.3 Conteúdo Pedagógico 2.4 Estrutura e Organização da Informação 2.5 Extensão e Densidade da Informação 2.6 Domínio e Nível de Complexidade do Conteúdo 3. Aspetos Pedagógicos 3.1 Público Visado 3.2 Contexto Curricular de Utilização 3.3 Objetivos de Aprendizagem 3.4 Estratégias de Exploração da Informação 3.5 Motivação 3.6 Autonomia na Aprendizagem 3.7 Interação Social 3.8 Formas e Instrumentos de Avaliação
-----------------	--

Anexo G - Quadro III- Interface Gráfica, Interatividade e Ferramentas de Exploração, (Costa F. A., 1999)

Grupo III	<p>4. Interface Gráfica</p> <p>4.1 Zonas de Comunicação 4.2 Formas de representação da Informação</p> <p>5. Interatividade</p> <p>5.1 Estrutura de Comunicação 5.2 Feedback 5.3 Grau de Participação e Controle por parte do Utilizador</p> <p>6. Ferramentas de Exploração</p> <p>6.1 Mecanismo de Ajuda 6.2 Meios de Navegação 6.3 Sistema de Orientação 6.4 Sistema de Pesquisa 6.5 Registo de Notas 6.6 Impressão e Exportação da Informação</p>
------------------	---

Anexo H - Quadro I Usabilidade da Aplicação, (Costa F. A., 1999)

Grupo IV	<p>7. Usabilidade</p> <p>7.1 Necessidade 7.2 Utilidade 7.3 Flexibilidade 7.4 Versatilidade 7.5 Fiabilidade/Solidez 7.6 Facilidade de Aprendizagem 7.7 Valor Atribuído ao Conteúdo 7.8 Satisfação Global 7.9 Documentação de Apoio 7.10 Avaliação Global Enquanto Ferramenta de Aprendizagem</p>
-----------------	--

Anexo I - Origem das normas ISO e/ou IEC (International standards for HCI and usability, 2003)

Título ISO	Descrição
ISO nnn (data)	A data é representada por um número padrão. Desenvolvido por uma comissão da ISO.
ISO xx-xxx	Xx parte de um padrão desenvolvido por uma comissão da ISO
ISO/IEC nnn (data)	Um padrão desenvolvido pela JTC1: uma comissão técnica conjunta da ISO e IEC.
ISO TS nnn (data)	Uma ISO Especificação Técnica: um documento normativo, que poderá ser revisado e publicado como um padrão.
ISO TR nnnn (data)	Uma ISO Technical Report: um documento informativo contendo informações de um tipo diferente do que normalmente publicada num padrão normativo.
ISO nnn ZZ (data)	Um projeto de norma do tipo ZZ disponibilizados no data.

Anexo J - Normas ISO e outras, fonte: (International standards for HCI and usability, 2003)

	Princípios e Recomendações	Especificações
Contexto de Utilização	ISO/IEC 9126-1: Engenharia de Software Qualidade do Produto, Parte I: Qualidade do Modelo	ISO 20282: Usabilidade para todos os produtos
	ISO/IEC 9126-4: Engenharia de Software Qualidade do Produto, Parte 4: Métricas de Qualidade	
	ISSO 9241-11: Orientações sobre Usabilidade	
Interface e Interação	ISO/IEC TR 9126-2: Engenharia de Software Qualidade do Produto - Parte 2 Métricas Externas.	ISO 9241: Requisitos Ergonómicos para escritórios com terminais. Partes 3-9
	ISO/IEC TR 9126-3: Engenharia de Software-Qualidade do Produto – Parte 3 Métricas Internacionais.	ISO/IEC 10741-1: Diálogos Interactivos controle do cursor para a edição de texto.
	ISO 9241: Requisitos Ergonómicos para escritórios com terminais – Partes 10-17.	ISO/IEC 11581: Ícones símbolos e funções.
	ISO 11064: Desenho Ergonómico para Centros de Controle.	ISO 13406: Requisitos Ergonómicos para o trabalho com indicadores visuais baseados em ecrãs planos.
	ISO 14915: <i>Software</i> Ergonómico para Interfaces de Multimédia	ISO/IEC 14754 Interfaces com a utilização de canetas, editores de texto com sistemas pen-based.
	IEC TR 61997: Linhas Orientadoras para interfaces de multimédia	ISO/IEC 18021: Tecnologias de Informação - Interface para ferramentas móveis.
		ISO 18789: Requisitos Ergonómicos e técnicas de medição para visores electrónicos.
Documentação	ISO/IEC 18019 Diretrizes para a elaboração e preparação de documentação de usuário do <i>software</i>	ISO/IEC 15910: Software processo de documentação do usuário
Processo de Desenvolvimento	ISO 13407 Desenho centrado no homem processos para sistemas interactivos	ISO/IEC 14910: Software processo de documentação do usuário
	ISO TR 16982: Métodos de usabilidade de apoio design centrado no ser humano	
Capacidade	ISO TR 18529 : Ergonomia da interacção homem-sistema - descrição dos processos do ciclo de vida humana centrada	
Outros	ISO 9241 – 1 Parte 1: Introdução Geral	
	ISO 9241-2 : Parte 2: Orientação sobre os requisitos de tarefa	
	ISO 10075-1 : Princípios ergonómicos relacionados com a carga de trabalho mental - as condições gerais e definições	
	ISO 16071 DTS : Orientação sobre a acessibilidade para interfaces homem-computador	

Anexo L

Palavras e Palavras.iss – Avaliação Heurística

Pelo Mestre Luís Carlos Bruno

Interface avaliado (#1):

Problema: Quando o utilizador entra na página da aplicação pode ser fornecida uma mensagem de boas vindas que explique sumariamente as principais tarefas que o utilizador pode realizar no sistema.

Heurísticas Violadas: Correspondência entre o Sistema e o Mundo Real e Consistência e Aderência a Padrões.

Solução: Adicionar por voz e/ou texto uma mensagem boas vindas que sintetize as principais tarefas que os utilizadores podem realizar a seguir.

Problema: Quando o utilizador passa sobre uma zona de interação (botão) o cursor passa a ser representado por uma folha, que não constitui uma metáfora reconhecida universalmente neste contexto (este problema é persistente nas interfaces da aplicação).

Heurística Violada: Consistência e Aderência a Padrões

Solução: Utilizar um cursor que seja reconhecidamente universal para este tipo de ações, como por exemplo o cursor com a forma de mão.

Interface avaliado (#2):

Problema: Quando o utilizador seleciona um utilizador e escolhe a opção Apagar, o sistema não pede confirmação da operação, apagando de imediato o utilizador do sistema.

Heurística Violada: Prevenção de Erros.

Solução: A seguir à escolha da opção “Apagar” deve aparecer um diálogo a solicitar a conformação da operação “Apagar”.

Problema: O botão “selecionar” não parece ter nenhuma função associada. Mais parece ser uma etiqueta para escolher uma opção da lista de utilizadores que está por baixo.

Heurística Violada: Consistência e Aderência a Padrões.

Solução: Retirar a moldura da opção “Selecionar” para não parecer um botão e alterar a *label* para “Selecionar:”.

Problema: Existem na interface dois diferentes tipos de botões: os que têm etiquetas de texto e os que só têm representação gráfica (mochos). Estes últimos são de mais difícil percepção por parte dos utilizadores (embora tenham etiqueta de voz), porque não têm explícita a sua função (esta situação é extensiva ao resto da aplicação).

Heurística Violada: Reconhecimento em vez de memorização.

Solução: Associar uma *label* aos botões que só têm representação gráfica (mochos), por exemplo por baixo.

Problema: As instruções por fala quando são enunciados os nomes dos comandos não têm correspondência com os nomes das etiquetas dos botões. Por exemplo, para o comando “Inserir Jogador” é falado “Insere o teu nome” ou para selecionar é dito “Seleciona-o”.

Heurística Violada: Reconhecimento em vez de memorização.

Solução: Associar uma *label* aos botões que só têm representação gráfica (mochos).

Interface avaliado (#3):

Problema: As instruções por fala quando se passa sobre as opções dos ramos não explicam convenientemente as tarefas os alunos podem realizar. Em vez disso são dadas instruções sobre os passos da tarefa seguinte. Ou seja, dizer neste ecrã que se devem arrastar as letras para os retângulos da mesma cor (quando aqui não se visualizam letras nem retângulos) pode tornar-se confuso para o aluno.

Heurística Violada: Correspondência entre o Sistema e o Mundo Real.

Solução: As instruções de fala devem explicar cada tarefa a que se pode aceder.

Exemplo (Construir Palavras): “Aqui podem construir palavras escolhendo as letras na ordem certa”.

Problema: O menu na base da aplicação contém diferentes tipos de botões. Dois com etiqueta e outros só com representação gráfica.

Heurística Violada: Consistência e Aderência a Padrões

Solução: Os botões deveriam ser consistentes, tanto graficamente, como na presença das etiquetas, para que os utilizadores identifiquem as suas ações mais facilmente.

Interface avaliado (#4):

Problema: Após a instrução inicial por fala (se por acaso o aluno não ouviu), não temos outra forma de ajuda sobre a tarefa que podemos realizar nesta interface.

Heurística Violada: Ajuda e Documentação e Visibilidade do Estado do Sistema

Solução: O sistema deve permitir repetir a instrução inicial por fala (deve haver uma metáfora no sistema que permita aceder a este tipo de ajudas – Um botão de Ajuda (talvez)) e/ou fornecer a mesma mensagem por escrito, por baixo da área das letras.

Interface avaliado (#5):

Problema: Não existe um título que identifique a tarefa em causa.

Heurística Violada: Visibilidade do Estado do Sistema.

Solução: Colocar no topo da janela o título “Constrói Palavras”.

Problema: Não sabemos o nível do jogo em que estamos, quando estamos nesta interface.

Heurística Violada: Reconhecimento em vez de memorização e Visibilidade do Estado do Sistema.

Solução: Associar uma informação à interface, numa posição adequada que permita identificar qual o nível.

Problema: Os botões dos níveis não têm coerência com os outros botões que são utilizados previamente no sistema. Inicialmente pareciam-me etiquetas. Só depois percebi que tinham ações associadas.

Heurística Violada: Consistência e Aderência a Padrões.

Solução: Colocar cada um daqueles botões com o formato dos anteriores, ou seja, suportados por caixas de moldura verde.

Problema: Quando entramos nesta interface não temos qualquer instrução sobre o que o aluno deve fazer (nem falada, nem por escrito).

Heurística Violada: Ajuda e Documentação e Visibilidade do Estado do Sistema.

Solução: Quando se entra no sistema deve ser transmitida ao utilizador uma mensagem de ajuda (semelhante à que estava no interface #3 para a opção construir palavras) e/ou fornecer a mesma mensagem por escrito (teria que se estudar o enquadramento tanto para este caso como para outros).

Problema: Não existe forma de eu voltar à interface anterior a esta (a que permite escolher um jogo a partir das letras).

Heurística Violada: Liberdade e Controle do Sistema pelo Utilizador.

Solução: Deveria haver um botão “voltar” ou com outra etiqueta que me permitisse escolher um novo jogo.

Anexo M

```
if (janela.Imagem.Width >= picJanela.Width || janela.Imagem.Height >= picJanela.Height) {
    Image templImage = (Image)janela.Imagem;
    Size fitImageSize =
comum.getScaledImageDimensions(janela.Imagem.Width,janela.Imagem.Height,
picJanela.Width-10, picJanela.Height-10);
    Bitmap imgOutput = new Bitmap(templImage, fitImageSize.Width,fitImageSize.Height);
    picJanela.Image = imgOutput;
}
```

...

```
((Player)(this.jogadores[this.pos_jogador])).DP_Certas =
((Player)(jogadores[pos_jogador])).DP_Certas + 1;
if (windows_left == 0) {
    mySound ms = new mySound();
    comum.PlaySound(ms.getRandomParabensFinal(),Application.StartupPath);
} else {
    mySound ms = new mySound();
    comum.PlaySound(ms.getRandomParabens(),Application.StartupPath);
}
```

Mocho_Fala();

...

```
DateTime ti = DateTime.Now;
DateTime tf = DateTime.Now;
TimeSpan ts;
ts = tf - ti;
while (ts.Seconds < 2) {
    tf = DateTime.Now;
    ts = tf-ti;
}
if (num_ultima_janela >= 0) {
    this.picJanelas[num_ultima_janela].Image = null;
    this.lblJanelas[num_ultima_janela].Text = " ";
}

if (num_janela >= 0) {
    this.picJanelas[num_janela].Image = null;
    this.lblJanelas[num_janela].Text = " ";
}
```

Anexo N

```
private void DrawGraphic (int xi, int yi, int width, int height, ArrayList resultados, string jogo,
Graphics g) {
    // max value is ..
    int maxval = -1;
    foreach (Pontos pts in resultados) {
        if (jogo.Equals("CP"))
        {
            if (pts.CP_Certas > maxval)
                maxval = pts.CP_Certas;
            if (pts.CP_Erradas > maxval)
                maxval = pts.CP_Erradas;
        }
        else if (jogo.Equals("DP"))
        {
            if (pts.DP_Certas > maxval)
                maxval = pts.DP_Certas;
            if (pts.DP_Erradas > maxval)
                maxval = pts.DP_Erradas;
        }
        else
        {
            if (pts.CF_Certas > maxval)
                maxval = pts.CF_Certas;
            if (pts.CF_Erradas > maxval)
                maxval = pts.CF_Erradas;
        }
    }
    if (maxval > 0)
        g.DrawString(maxval+"",new Font("Comic Sans MS",8),new
        SolidBrush(Color.Green),new System.Drawing.PointF(23,58));

    System.Drawing.PointF [] pontos_certas = new
    System.Drawing.PointF[resultados.Count+1];
    System.Drawing.PointF [] pontos_erradas = new
    System.Drawing.PointF[resultados.Count+1];

    pontos_certas[0] = new System.Drawing.Point(xi,yi+height);
    pontos_erradas[0] = new System.Drawing.Point(xi,yi+height);

    int jogo_atual = 1;

    for (int i = 0; i < resultados.Count; i++) {
        float x_pos = (jogo_atual*(xi+width)) / resultados.Count;
        float y_pos1 = 0.0f;
        float y_pos2 = 0.0f;
    }
}
```

```

        if (jogo.Equals("CP"))
        {
            y_pos1 = ( ((Pontos)(resultados[i])).CP_Certas * height) / maxval;
            y_pos2 = ( ((Pontos)(resultados[i])).CP_Erradas * height)/ maxval;
        }
        else if (jogo.Equals("DP"))
        {
            y_pos1 = ( ((Pontos)(resultados[i])).DP_Certas * height) / maxval;
            y_pos2 = ( ((Pontos)(resultados[i])).DP_Erradas * height)/ maxval;
        }
        else
        {
            y_pos1 = (((Pontos)(resultados[i])).CF_Certas * height)/ maxval;
            y_pos2 =((Pontos)(resultados[i])).CF_Erradas * height)/ maxval;
        }
        pontos_certas[jogo_atual] = new System.Drawing.PointF(x_pos,
            (yi+height)-y_pos1);
        pontos_erradas[jogo_atual] = new System.Drawing.PointF(x_pos,      yi+height)-
y_pos2);
        jogo_atual++;
    }

    g.DrawLines(new Pen(Color.Green, 1),pontos_certas);
    g.DrawLines(new Pen(Color.Red, 1),pontos_erradas);
}

```