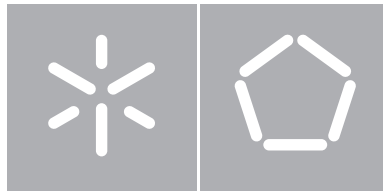




Universidade do Minho
Escola de Engenharia

Diogo Ferreira Soares da Silva
Assistente pessoal para suporte ao
ensino da leitura do Português a
crianças



Universidade do Minho

Escola de Engenharia
Departamento Informática

Diogo Ferreira Soares da Silva

Assistente pessoal para suporte ao
ensino da leitura do Português a
crianças

Dissertação de Mestrado
Mestrado em Engenharia Informática

Trabalho realizado sob orientação de
Professor Orlando Belo
Professor José Bernardo Barros

Agradecimentos

Ao meu orientador, professor Orlando Belo, que, graças ao seu conhecimento, experiência, dedicação e profissionalismo, me conduziu ao longo desta dissertação. Realçando a sua disponibilidade e acompanhamento, bem como as boas orientações que sempre me concedeu.

A minha namorada pela paciência, apoio, companheirismo e dedicação ao longo de toda a dissertação.

Aos meus pais e aos meus irmãos, pelo natural apoio demonstrado e esforço realizado, para que eu conseguisse culminar mais uma etapa da minha vida acadêmica e pessoal.

Por último, a todos aqueles que, apesar de não estarem aqui explicitamente referidos, me ajudaram e contribuíram direta ou indiretamente para a realização desta dissertação.

O meu mais sincero obrigada a todos.

Resumo

Um Assistente Pessoal para Suporte ao Ensino da Leitura do Português a Crianças

Os ITS (Intelligent Tutoring Systems) são sistemas informáticos que se enquadram na área de Inteligência Artificial e têm por objetivo permitir a um aluno o acesso a um processo de aprendizagem eficaz. Geralmente, estes sistemas são comparados a professores e a sua constituição desenvolve-se normalmente, em quatro componentes base, nomeadamente: o modelo do domínio, o modelo do aluno, o modelo do tutor e, por fim, o modelo de interface com o utilizador. Para alcançar o seu objetivo principal (uma aprendizagem orientada, completa e eficaz) é necessário que os ITS possuam uma base de conhecimento fiável e sustentável especialmente orientada para o tema a ser ensinado. Atendendo ao objetivo desta dissertação, que assenta na criação de um assistente pessoal para auxiliar o educador(a) no desenvolver do domínio da linguagem oral e abordagem à escrita, foi escolhido como base de trabalho o método de ensino e tutoria da Cartilha Maternal de João de Deus. Deste modo, o ITS a desenvolver deverá ser autónomo, fácil de utilizar e, bem como ter a capacidade de planear e organizar as ações, conforme o utilizador em questão e de forma, apoiá-lo na atividade a realizar. Neste trabalho de dissertação, exporemos e explicaremos o processo de desenvolvimento que foi levado a cabo para a produção do referido sistema.

Palavras-Chave: Sistemas de Ensino Inteligentes, Agentes Pessoais, Ferramentas de Ensino, Linguagem Oral e Abordagem à Escrita, Ensino do Português.

Abstract

A Personal Assistant for Supporting Teaching of Reading the Portuguese Language to Children.

ITS (Intelligent Tutoring Systems) are systems that fall under the area of Artificial Intelligence and aim to allow a student to access an effective learning process. Generally, these systems are compared to teachers and its incorporation develops normally in four basic components, namely: The domain model, the student model, the tutor model and, finally, the interface with the user model. To achieve its primary objective (learning-oriented, comprehensive and effective) it is necessary that ITS have a reliable and sustainable knowledge base especially oriented to the topic being taught. Given the objective of this dissertation, based on the creation of a personal assistant to assist the educator in developing the field of oral language and approach to writing, the “Cartilha Maternal João de Deus” was chosen as a basis method of teaching and mentoring. Thus, the ITS to develop should be autonomous, easy to use and have the ability to plan and organize the users actions and so, support them in the activity that’s to be perform. In this dissertation, we will expose and explain the development process that’s carried out for the production of such a system.

Keywords: Intelligent Teaching Systems, Personal Agents, Teaching Tools, Oral Language and Approach to Writing, Portuguese Teaching.

Índice

Introdução.....	1
1.1 <i>Software</i> Educativo	1
1.2 Sistemas de Tutores Inteligentes	3
1.3 O Caso de Estudo.....	4
1.4 Motivação	5
1.5 Objetivos do Trabalho	6
1.6 A Estrutura do documento	7
<i>Software educativo</i>	8
2.1 Caracterização e Propriedades.....	10
2.2 A Metodologia de Desenvolvimento	13
2.3 Exemplos de Aplicações.....	16
2.4 Modelos de Desenvolvimento	18
2.4.1 CAI.....	18
2.4.2 ICAI.....	22
Sistemas de Ensino Inteligentes.....	25
3.1 Caracterização Base	25
3.2 Propriedades de um ITS	29
3.3 A Arquitetura	30

3.4 Modelação	35
3.4.1 A Modelação clássica.....	35
3.4.2 Modelação com Agentes	36
3.5 Exemplos	37
O Sistema da Cartilha	42
4.1 Caracterização base	42
4.1.1 A Cartilha Maternal João de Deus.....	43
4.1.2 O Sistema Android	51
4.1.3 Texto para Voz - <i>Text-to-Speech</i>	52
4.1.4 Reconhecimento de Voz - <i>Speech Recognition</i>	54
4.1.5 Gestps - <i>Gestures</i>	55
4.2 Funcionalidades Gerais	55
4.3 Modelo e arquitetura	58
4.3.1 O Esquema da Base de Dados de Suporte	59
4.3.2 A Arquitetura Funcional do Sistema.....	62
4.4 Demonstração de Funcionalidades	64
4.4.1 A Cartilha de João de Deus	64
4.4.2 As Lições da Cartilha	65
4.4.6 Definição de Planos de Trabalhos	69
4.4.7 Caracterização de um Perfil de Utilização.....	69
Conclusões e Trabalho Futuro	72
5.1 Resumo e apreciação do trabalho realizado	72
5.2 Trabalho futuro	74
Bibliografia	77

Índice de Figuras

Figura 1 - Áreas abrangidas pelos ITS.....	27
Figura 2 – A Arquitetura Clássica dos ITS (Giraffa e Rodrigo Rafael Villarreal Goulart 2001).....	31
Figura 3 - Processamento de texto realizado pelo Text-To-Speech – figura extraída de (Wikipedia 2014d).....	53
Figura 4 – O menu principal da aplicação.	56
Figura 5 – O esquema lógico da base de dados do sistema.	59
Figura 6 – A Arquitetura funcional do sistema. – figura adaptada de (Giraffa e Rodrigo Rafael Villarreal Goulart 2001).....	62
Figura 7 – A Funcionalidade para a visualização da Cartilha Maternal.	64
Figura 8 – A funcionalidade das lições da cartilha.....	65
Figura 9 – O layout da execução das lições da cartilha.	66
Figura 10 – O <i>layout</i> do reconhecimento de voz.	67
Figura 11 – O <i>layout</i> do painel Quizz.....	68
Figura 12 – O layout do reconhecimento de caracteres.	68
Figura 13 – A funcionalidade Plano de Trabalhos	69
Figura 14 – caracterização de um perfil de utilização.....	70

Índice de Tabelas

Tabela 1 - Taxonomias de Programas Educacionais	11
Tabela 2 – Comparação de características dos sistemas CAI e ITS.	28
Tabela 3 - Tabela Comparativa de Sistemas ITS.	40
Tabela 4 - Jardins Escolas que leccionam com base na Cartilha Maternal João de Deus.	46
Tabela 5 – Caracterização dos atributos das relações da base de dados.	62

Lista de Siglas e Acrónimos

CAI	- Instrução Assistida por Computador
IA	- Inteligência Artificial
ICAI	- Instrução Assistida por Computador Inteligente
ITS	- Sistemas Tutores Inteligentes

Capítulo 1

Introdução

1.1 *Software* Educativo

O software educativo tem como principal objetivo facilitar o processo de ensino-aprendizagem, de modo a que um aluno consiga construir determinado conhecimento relativo ao conteúdo didático abordado pelo software. No entanto a definição deste tipo de software pode ser considerada generalista. Relativamente a este, Giraffa (1999) considera que, “todo programa que utiliza uma metodologia que o contextualize no processo de ensino e aprendizagem, pode ser considerado educacional.”. Neste sentido, percebe-se que esta classificação é adotada na atualidade por diversos tipos de software que são usados e por vezes adaptados ao contexto educacional, apesar de concebidos para fins empresariais. Como tal, a definição deste tipo de software pode ser considerada generalista.

Ao longo do tempo o seu processo de construção tem sofrido algumas alterações, uma vez que, inicialmente, eram concebidos para projetar a prática pedagógica que era transmitida na sala de aula. De referir que no seguimento deste paradigma, o software teria como função substituir o livro didático. No entanto este seria dos primeiros passos na inserção de tecnologia no processo educativo com o intuito de estimular os intervenientes para o mesmo, uma vez que “as tecnologias digitais ampliam as possibilidades de comunicação e expressão, criando novas alternativas para a leitura/escrita sequencial” (Valente, 1999).

Na década de 50 iniciou-se o uso da informática na educação com a inserção do conceito da instrução programada, elaborada por B. F. Skinner. Este conceito baseia-se na divisão do conteúdo a ser ensinada por módulos sequenciais. No fim de cada módulo o aprendiz é sujeito a uma avaliação dos conceitos adquiridos no mesmo módulo. De salientar que só avança para o módulo seguinte quando a avaliação do módulo anterior for aprovada. Com a chegada do computador a instrução programada teve grande impacto, passando a ser conhecida por Instrução assistida por computador (*Computer-Aided Instruction* ou CAI). Nesta modalidade o aluno assume um comportamento passivo, recebendo apenas informações. Devido à mesma projetar uma visão empirista do conhecimento, de acordo com (Carragher 1990).

Mais tarde com o avanço para os microcomputadores a modalidade começou a ser desenvolvida na forma de tutoriais, jogos educacionais e simulação, demonstração e programas do tipo exercício e prática. Neste campo do software educativo, por volta dos anos 60 assistiu-se ao desenvolvimento de um novo paradigma de aprendizagem diferente que era utilizado nos CAI. Quanto a este modelo, o seu foco incide na exploração do conhecimento e a interação com os objetos, em que eram utilizadas as ideias de micromundos e ambientes criados por computador. Por volta dos anos 70 e 80 o campo da Inteligência Artificial tornou-se um pouco dominante no processamento de informação. Surgiram, então, algumas investigações que destacavam a melhor forma de

representar o conhecimento num sistema inteligente. Com o decorrer dessas investigações foi adaptado a modalidade CAI para ICAI (Instruções Assistidas por Computador Inteligentes), estas apresentam uma estrutura diferenciada do seu antecessor visto que usufruem de técnicas de Inteligência Artificial e Psicologia Cognitiva para guiar o processo educativo.

Em 1982, o “estado da arte” dos sistemas CAI foi revisto por Sleeman e Brown, e daí foi criado o termo Sistemas Tutores Inteligentes (Intelligent Tutoring Systems ou, simplesmente, ITS). Este termo surgiu para os distinguir dos sistemas antecessores CAI, tendo como suposição implícita o aprender focalizado no aprender fazendo. Esta modalidade facilita muito o processo educativo tornando-o mais efetivo, correto e também mais agradável.

1.2 Sistemas de Tutores Inteligentes

Os Sistemas de Tutores Inteligentes surgiram inicialmente rotulados como ICAI, uma vez que esta nova modalidade deu-se com o crescimento da Inteligência Artificial, domínio em que se iniciaram investigações sobre a modalidade CAI, do qual emergiu o nome ICAI. Mais tarde o estado da Arte dos ICAI foi revisto e a modalidade foi renomeada por Sistemas de Tutores Inteligentes (ITS - Intelligent Tutoring Systems). Para esta modalidade surgiram diversas definições, tais como:

- “Os STI são programas de software que dão suporte às actividades de aprendizagem.” (Gamboa e Fred, 2002).
- “Sistema Tutor Inteligente, é um termo amplo, abrangendo qualquer programa de computador que contem alguma inteligência e pode ser usado em aprendizagem.” (Freedman et al., 2000).
- “Os STI são programas de computador com propósitos educacionais que incorporam técnicas de Inteligência Artificial. Oferecem vantagens sobre os CAIs (Instrução Assistida

por computador), pois podem simular o processo do pensamento humano para auxiliar na resolução de problemas ou em tomadas de decisões” (Fowler, 1991)

A modalidade em assunto foi pensada para funcionar como um “docente substituto”, seguindo um conceito baseado em exercícios de pergunta/estímulo-resposta. Quanto ao software, este contém um determinado domínio a ser ensinado, ao que o discente responde de acordo com as questões do sistema, e o software procederá à avaliação das suas respostas. O resultado da avaliação determina os passos a seguir. Ou seja, a modalidade ITS é um tutor computadorizado que disponibiliza aos discentes um conjunto de instruções focadas no domínio e personalizadas. Para além disso, transmite-lhes feedback sem intervenção direta de seres humanos. Isto acontece em paralelo com a execução das tarefas, no processo educativo.

Os ITS, utilizam um diverso leque de tecnologias, todas bastante diferentes. No entanto, tal como o nome indica, estes são na sua grande maioria concebidos como sistemas de IA, uma área de sistemas especialistas, concebidos para simular aspetos do tutor humano. Estes implementam a teoria de aprender fazendo, o que permite a exploração de conteúdos e uma maior motivação/captação do discente.

1.3 O Caso de Estudo

O caso de estudo escolhido para suporte a este trabalho de dissertação foi a Cartilha Maternal (Deus 1878), escrita pelo poeta e pedagogo João de Deus e publicada em 1876. Trata-se de um método com base no ensino da leitura a crianças. Uma das primeiras condições do ensino por este método é o estudo da fala. A cartilha defende que é falando e ouvindo, que as crianças conseguem segmentar as palavras que lhes interessam, e isto muito antes de serem capazes de empregar frases. Além disso umas das características da cartilha é exatamente o tipo de impressão adotado

nas lições, nas quais são apresentadas palavras segmentadas silabicamente através do recurso ao preto/cinza. Com este modo de apresentação gráfica, esta metodologia recusa-se a tratar as sílabas independentemente das palavras que estão inseridas. João de Deus defende que este aspeto permite obter a decomposição das palavras sem quebrar a unidade gráfica e sonora das mesmas. Na atualidade, este método é aplicado em alguns jardins-de-infância, nos quais as crianças chegam aos seus 5 anos de idade e já possuem a capacidade de ler, ao que muitos chamam o “ano 0” do primeiro ciclo.

Este projeto tem como objetivo empregar este método já com sucesso comprovado e aplicar nas novas tecnologias e poder inovar a forma de apresentação e de trabalho do método, bem como auxiliar os processos já existentes no mesmo método de ensino.

1.4 Motivação

A grande evolução que as tecnologias da informação e da comunicação sofreram nos últimos anos facilitaram o acesso a este tipo de ferramentas e tornaram mais notória a sua utilização em variados processos educativos. Além disso, verificou-se, também, que a área educacional precisa de acompanhar o avanço das novas tecnologias adotando-as e adaptando-as ao processo educativo, permitindo a criação de novos paradigmas de ensino e, como tal, otimizar e inovar os seus recursos educativos. Hoje, pode-se verificar que já existem alguns sistemas computadorizados interativos que auxiliam neste processo de adoção e adaptação. Por exemplo, disponibilizar programas, material didático ou artigos pedagógicos e científicos, ou até mesmo sistemas com algumas capacidades de Inteligência Artificial que são capazes de tornar os processos de ensino mais personalizados, mais expeditos e menos convencionais.

A motivação deste projeto assenta na criação de um sistema móvel que irá permitir às crianças, em particular, a aprender de forma lúdica e dinâmica a escrever e falar português, seguindo o método de ensino da cartilha maternal João de Deus, disponibilizando uma ferramenta bastante versátil e capaz de auxiliar as atividades de ensino do método referido. Para além disso, permitirá prover aos tutores uma ferramenta que irá auxiliar, não só no processo de ensino como também irá permitir ter uma noção mais exata e real da evolução de cada aluno ao longo do processo.

1.5 Objetivos do Trabalho

A presente dissertação tem como objetivo desenvolver um sistema de ensino para ambientes móveis, mais concretamente para plataformas computacionais do tipo *tablet*. Tal como referido anteriormente, este sistema suportará o método de ensino da cartilha maternal João de Deus, como caso de estudo. De salientar que o sistema terá por base a metodologia dos ITS existente em software educativo. Atendendo à faixa etária do público-alvo, para o sistema em questão, a interface deverá ser muito amigável, colorida e com elementos numa escala de maiores dimensões, de forma a que possa ser utilizado e explorado por crianças por volta dos 5 anos. Preocupações acrescidas deverão também existir relativamente aos contrastes de cores nos diversos elementos apresentados e utilizados como suporte ao processo de aprendizagem escolhido para o sistema. Porém, o sistema deve ser suficientemente versátil para acolher outro tipo qualquer de caso de estudo. Assim, pretendeu-se desenhar e construir um sistema protótipo, inicial, que fosse o mais abrangente possível e aplicável a outros métodos de ensino, isto, obviamente, sem esquecer o caso de estudo escolhido.

1.6 A Estrutura do documento

Para além do presente capítulo, este documento encontra-se dividido em mais quatro capítulos, com o intuito de apresentar o trabalho desenvolvido nesta dissertação. Esta encontra-se organizada de forma a proporcionar ao leitor uma suave e gradual transição entre os conceitos nela apresentada. Assim, no segundo capítulo será abordada a área do software educativo, fazendo-se uma descrição desse tipo de software, apresentando as suas características e propriedades fundamentais, bem como alguns exemplos da sua aplicação. Além disso, também serão relatados os modelos de desenvolvimento que existem neste tipo de software e apresentada a arquitetura base usualmente adotada neste tipo de software.

No que concerne ao terceiro capítulo, será abordada uma metodologia do software educativo que tem por nome Sistemas de Tutores Inteligentes. Neste capítulo apresentar-se-á a sua caracterização base e as suas propriedades. Além disto, serão explanados os tipos desta metodologia. Por fim, destacar-se-á alguns exemplos da metodologia em questão. O quarto capítulo destina-se à apresentação do Sistema da Cartilha, o protótipo que foi implementado nesta dissertação, onde consta a caracterização geral do sistema. Além disso são expostas as suas funcionalidades e as ferramentas aplicadas para a realização dessas funcionalidades. De seguida é apresentado o modelo do sistema e a sua arquitetura para melhor compreensão do mesmo. Para finalizar o capítulo será apresentada uma demonstração e explicação das funcionalidades existentes no protótipo. Por último emerge o quinto capítulo com as conclusões desta dissertação. Primeiramente surge o resumo do trabalho realizado e conseqüentemente, uma avaliação crítica do mesmo. No final, seguir-se-á um tópico onde salienta o trabalho futuro a realizar nesta dissertação com o intuito de melhorara e refinar este protótipo.

Capítulo 2

Software educativo

Com o avanço das novas tecnologias e com o seu acesso cada vez mais facilitado, tem sido notória a sua utilização na sala de aula como suporte a um grande número de atividades. Isto acontece, no meu entender, com o intuito de melhorar o processo de aprendizagem e torná-lo mais acessível, intuitivo, e dinâmico, entre muitas outras novas possibilidades que criam. Este tipo de ferramentas acrescenta aos docentes uma capacidade para inovarem a prática pedagógica, proporcionando aos seus discentes o acesso a ambientes informatizados e a novas formas capazes de os auxiliarem no processo de ensino-aprendizagem.

Contudo este conceito é demasiado amplo, pois é usado para classificar alguns tipos de software que são usados para fins educacionais, mas que não foram desenvolvidos para esse mesmo efeito, como por exemplo processadores de texto, folhas de cálculo ou geradores de gráficos. De acordo com (Giraffa 1999), a visão mais consensual na comunidade Informática Educativa é a de que “todo programa que utiliza uma metodologia que o contextualize no processo de ensino e aprendizagem, pode ser considerado educacional.”. Partindo deste princípio, podemos afirmar que

o conceito é demasiado abrangente. Porém este *software* pode ser classificado em duas classes, que são:

- **Software genérico**, que pode ser usado para lecionar em qualquer disciplina, ou seja não apresenta uma temática exclusiva para lecionar, nos qual se pode encontrar os processadores de texto, as folhas de cálculo, etc.
- **Software específico**, que é desenvolvido com a finalidade de ser usados no ensino, em que lhe é atribuído um tema que deve lecionar, como é o caso do *software* que é desenvolvido para o ensino de temas como a ciência, prática de idiomas, exercícios de matemática, entre outras (Fontes, 2006).

Nesta dissertação vamos abordar a questão do *software* educacional como software específico, ou seja, a vertente em que o software educacional é desenvolvido com o objetivo de lecionar um tema, que no nosso caso assenta na prática da leitura de português a crianças, usando o método da “Cartilha Maternal de João de Deus”. O *software* deverá ser capaz de tornar o método de aprendizagem mais motivador, desenvolver a criatividade do discente, a concentração, a memória entre outros aspetos. Além disso o sistema deve permitir, de forma ponderada, entrar em viagens de estudo virtuais e tornar os momentos de aprendizagem em momentos de lazer, em que num dado processo educativo é dada uma resposta educativa contextualizada, respeitando as especificidades do método de ensino e da identidade de cada aluno. Porém, o sistema deve ter como preocupação a faixa etária em que é pretendido este incidir, visto que o método usado no caso de estudo é direcionado para crianças a partir dos 3 anos. De salientar que para além das funcionalidades que esta peça de software deve conter, de acordo com o usual do *software* educativo, estas devem ser otimizadas de modo a cativar as crianças desta faixa etária.

2.1 Caracterização e Propriedades

Como referido anteriormente, o *software* educativo tem como principal objetivo facilitar e otimizar o processo de ensino-aprendizagem, direcionando o aluno na construção de conhecimento relativo a um determinado tema aplicado neste *software*. Porém, o *software* apresenta algumas características que o diferenciam de outros tipos de *software*. Essas características são:

- A presença de fundamentação pedagógica que é definida pelo sistema no seu desenvolvimento.
- A condução do aluno na “construção” do conhecimento, no contexto didático existente.
- A interação com o sistema, garantido que deve ser permitido ao aluno que no primeiro contacto seja capaz de desenvolver as atividades, sem conhecimentos computacionais prévios.
- Recursos lúdicos, apelando à utilização de recursos multimédia (sons, imagens, etc.), com o intuito de motivar e captar a atenção do discente.
- Avaliação do discente, que deve ter a capacidade de avaliar de acordo com o tema a ser lecionado, em prol da interação do discente, com o sistema a suportar as respostas por ele apresentadas.

Além das características anteriormente referidas, nesta dissertação, o *software* educacional a desenvolver deverá ter em conta o facto de ter sido idealizado para uma população infantil, ou seja, deve ser simples, intuitivo e de fácil utilização. Para tal, deve conter imensas imagens, cores e sons que estimulem a curiosidade e captem a atenção do discente. Deste modo, o conteúdo deve ser apresentado e transmitido de uma forma descontraída, uma vez que nestas idades o que importa é a diversão, apesar de se pretender que o sistema possa conduzir a criança a apresentar diferentes hipóteses, levando-a a explorar novos cenários de aprendizagem e consequentemente desenvolver

as suas capacidades. Considerando que o *software* pode ser algorítmico ou heurístico, podemos ainda classificar este tipo de *software* relativamente atividade do discente da seguinte maneira:

- **Algorítmico**, que tem como objetivo principal a transmissão de conhecimentos do especialista para o sujeito que tem vontade de aprender. O sistema é projetado com uma sequência bem organizada de atividades, de modo a poder levar o discente ao conhecimento desejado.
- **Heurístico**, que tem a capacidade de conduzir o discente num processo de aprendizagem experimental ou por descoberta, devido à sua capacidade de criação de ambientes ricos em situações que o aluno deve explorar.

Para auxiliar o programador a definir algumas das características básicas que o *software* deve conter, que posteriormente serão predominantes no sistema, na definição do processo de modelação, bem como na implementação do ambiente através de uma metodologia adequada, as taxonomias foram divididas em dois grandes grupos (Vicari e Giraffa, 2003): aprendizagem de capacidades específicas e aprendizagem por capacidades cognitivas amplas.(Tabela 1).

Programas educacionais	
1- Aprendizagem de capacidades específicas	2- Aprendizagem de capacidades cognitivas amplas
CAI <ul style="list-style-type: none"> • Exercício-prática • Jogos e Simulação • Demonstração • Tutoriais 	<ul style="list-style-type: none"> • Micromundos • Sistemas de autoria • Jogos educacionais • ILE
ICAI <ul style="list-style-type: none"> • Sistemas Especialistas • Sistemas de Tutores Inteligentes 	

Tabela 1 - Taxonomias de Programas Educacionais

Como podemos verificar, no primeiro grupo de aprendizagem por capacidades específicas destacam-se dois grandes grupos que são os CAI (*Computer Assisted Instruction*) e os ICAI (*Intelligent Computer Assisted Instruction*). Inicialmente o objetivo principal do CAI assenta na aprendizagem do aluno seguindo procedimentos de ensino específicos, nos quais as atividades são sequenciadas e as respostas previamente definidas. Este tipo de método de ensino é classificado como *behaviorista*. Com o crescimento da inteligência artificial os CAI sofreram algumas alterações devido à inserção de novas técnicas de IA, que permitiram tornar os procedimentos de ensino mais reais e adaptativos. Com isso, o nome foi alterado ficando com a designação de ICAI. Mais tarde com o sucesso da aplicação das técnicas de IA, e com os resultados que essa transformação trouxe, foi necessário distinguir os CAI dos ICAI. Assim, o estado da Arte foi revisto e a modalidade foi renomeada para Sistemas de Tutores Inteligentes (ITS - Intelligent Tutoring Systems).

Apesar de ser reconhecido o sucesso das aplicações de software educacional é preciso retirar deles quais são os seus benefícios. Como tal, é necessário, de antemão, definir de forma clara e precisa a finalidade a atingir. Pois, ao aplicar o *software* educativo de forma genérica, torna-o de certa forma limitado e o levará mais rápido ao insucesso no meio educacional. Quanto à sua concepção e utilização por parte do discente é necessária a colaboração crítica de um ou mais especialistas da área na qual este software vai incidir. Assim é necessário conhecer quais os critérios e os métodos que devem ser aplicados para abordar o tema e a forma como adaptá-los (informatizá-los), de modo a atingir o conhecimento pretendido, tal como é citado por (Ramos, 1996):

“No caso da informática, essa postura deve estar implícita tanto nas metodologias de aprendizagem, quanto nos próprios instrumentos tecnológicos desde o seu projeto. Ela deve, portanto, ser considerada nas metodologias de (desenvolvimento de) “software”, no projeto das interfaces e das funcionalidades dos sistemas de “software” e nas políticas de informatização das instituições”.

Tendo em conta este facto, podemos afirmar que o sucesso do software educacional a desenvolver depende de dois factores iniciais muito importantes: a aceitação de introduzir uma nova tecnologia nos procedimentos educativos e a participação e crítica do seu planeamento. De seguida, iremos abordar a metodologia que foi usada para o desenvolvimento do software referido.

2.2 A Metodologia de Desenvolvimento

Neste processo de desenvolvimento de software seguimos de perto uma metodologia, uma vez que este é um trabalho complexo e demasiado abrangente. Qualquer iniciativa de desenvolvimento de software educativo deve ser abordado de uma forma sistemática e interativa. Qualquer peça de software deste tipo deve assegurar o processo educativo e prestar o devido auxílio aos seus utilizadores, devendo para isso ser capaz de permitir o acesso a informação seleccionada e credível, que deve ser catalogada com o intuito de responder às necessidades dos utilizadores.

A metodologia de desenvolvimento de software educativo que adotámos divide o processo de desenvolvimento nas seguintes etapas:

- **Análise e planeamento.** Em todo o processo de desenvolvimento, esta é uma das fases mais determinantes, pois irá delinear todo o processo e definir alguns dos seus pontos chave. No entanto, além de ser a primeira fase, também será uma fase contínua ao longo de todo o processo, visto ser um procedimento fundamental em qualquer metodologia adoptada. O facto de ser das fases mais importantes, deve-se projetar nela o produto a conceber, o tema a lecionar, os recursos existentes, as plataformas nas quais se irá ser desenvolvido o produto e a recolha de dados necessária para posterior análise. Nesta fase, além de serem definidas as suas características

necessárias, expomos também algumas das limitações do projeto, uma vez que será aqui que será definido o conceito base do produto e diminuído o leque de possibilidades que um software educativo pode abranger. Esta fase deverá ajudar-nos a dar resposta a algumas questões determinantes no sucesso do produto, tais como:

- Como será usado?
- Qual(is) o(s) objetivo(s) do software?
- Qual o seu público-alvo?
- O conteúdo que será apresentado?
- Qual será o acesso ao software?

No entanto, esta análise não deverá ser apenas focada no âmbito interno de desenvolvimento, pois para a tornar mais rica deverão ser analisados produtos externos similares, quando isso for aplicável. Em suma, um dos motivos mais importantes desta fase assenta no facto de que no software educativo a ser desenvolvido seja feita uma exposição do conteúdo educativo de forma didática, sem esquecer os elementos pedagógicos, incluindo-se definições estratégicas adequadas. Também será preciso ter em conta os pré-requisitos e as características do público-alvo.

- **Modelação.** A fase de modelação do software educacional é determinante na definição qualitativa da suas funcionalidades, já que é nesta fase que os procedimentos necessários na implementação do software são desenhados, ajudando a compreender e aprovar o sistema antes dele ser concebido. É, pois, nesses modelos que vamos definir como é que o processo educativo será transcrito para o mundo digital, de modo a transmitir o conhecimento que tem como objetivo. Isto sem excluir os elementos pedagógicos e o público-alvo a incidir. Nesta fase podemos encontrar, essencialmente, três modelos, nomeadamente:

- **O Modelo Conceptual.** Este modelo tem como principal finalidade definir o domínio do software educativo, no qual será estruturada e organizada toda a informação, de modo a transmitir o conhecimento educativo que aplicação pretende lecionar. Ou seja, na preparação deste modelo pretende-se encontrar o conteúdo educativo da aplicação e quais as estratégias para a disposição desse mesmo conteúdo ao utilizador do software.
 - **O Modelo de Navegação.** Sendo este um modelo mais técnico, não deixa porém de ter a sua importância no processo, uma vez que será o responsável por definir as estruturas de acesso do software. Com este modelo podemos perceber como serão as ligações e a transferência de informação no fluxo de trabalho com a aplicação. Além disso, deve-se ainda ter em conta, que a navegação do software deverá ser intuitiva e fluida.
 - **O Modelo de Interface.** Neste modelo definir-se-á como serão apresentados os conteúdos didáticos ao utilizador, uma vez que é através dele que são definidos os *layouts* das interfaces com o utilizador, qual a aparência dos objetos apresentados no software e quais as suas identidades visuais. Este modelo depende, obviamente, dos modelos anteriores, uma vez que deve estar de acordo com as especificações do modelo conceptual e do modelo de navegação.
- **Implementação.** Tal como o próprio nome indica, esta fase consiste em implementar as projeções definidas e os resultados obtidos nas fases anteriores (planeamento e modelação), ou seja, começar, de facto, a produção do software. Contudo o alcançar desta fase não quer dizer que todas as anteriores já estão completamente definidas e fechadas. Isto porque, no decorrer desta fase é usual fazer-se uma verificação exaustiva

do conteúdo educativo, de modo a não existir erros conceptuais nem gramaticais. Por isso, alguns ajustamentos poderão ser feitos no decorrer desta fase.

- **Avaliação e Manutenção.** Por último, a fase de avaliação e manutenção, na qual se pretende passar o software desenvolvido por uma fase de testes e de verificação, com o intuito de proceder a eventuais processos de correção de erros, de conteúdo (lógicos) e de gramática (sintáticos). No entanto, esta fase deverá ocorrer ao longo de todos os processos anteriores, de modo a suprimir alguns erros que possam surgir. Além da realização dos processos de testes, nesta fase proceder-se-á à criação de manuais técnicos.

Em suma, esta metodologia é aplicada na educação mas, como foi referido anteriormente, qualquer desenvolvimento de software segue uma metodologia de desenvolvimento. Esta é apenas uma metodologia que segue os passos básicos de desenvolvimento de software. No entanto, quando aplicada ao desenvolvimento de um produto educacional, consegue-se fazer uma melhor estruturação do projeto do produto de software em causa.

2.3 Exemplos de Aplicações

Atualmente, já existe um leque muito variado de aplicações educacionais, como por exemplo sites educacionais, salas virtuais, entre outros. De seguida, são enumerados alguns exemplos de aplicações existentes:

- **Microsoft PowerPoint** (Wikipedia 2014b) Este software foi desenvolvido para editar e exibir apresentações gráficas. Este software é usado para apresentações, tendo como objetivo informar sobre algum tema, usando recursos hipermedia: imagens, sons, textos e vídeos. Além disso podem ser aplicados efeitos e animações sobre estes recursos.

Além disto o software tem suporte a objetos OLE e inclui uma ferramenta especial de formatação de texto (WordArt), modelos de apresentação pré-definidos, galeria de objetos gráficos e uma gama de efeitos de animação e composição de slides.

- **Microsoft Excel** (Wikipedia 2014a) Este software é um editor de folhas de cálculo, que conta com recursos com uma interface intuitiva e com ferramentas de cálculo e de construção de gráficos, que tornaram esta ferramenta um dos mais populares produtos de software até hoje. Apesar de estas não serem desenvolvidas como software educativo, são bastante usadas num contexto educacional.

De seguida são apresentados alguns softwares, em que o seu desenvolvimento tem como objetivo a criação de software educacional.

- **Moodle** (Wikipedia 2014c) Software para gestão da aprendizagem e de trabalho colaborativo, no qual é permitido a criação de cursos online, páginas de disciplinas e de grupos de trabalho. Nesta plataforma a aprendizagem é mais efetiva quando se constrói algo que os outros possam experimentar, que pode ser qualquer coisa, desde uma frase falada ou um comentário num fórum na Internet, até algum artifício como uma pintura, uma casa ou um pacote de software de educação.
- **Study Web** (Santos 1999) É um website que reúne uma coleção de 28.000 sites, entre sites educacionais sobre conteúdos curriculares, classificados por disciplina e por ano escolar, e sites de referência. Um motor de busca e mecanismos de orientação ao utilizador torna as consultas Study Web rápidas e seguras.
- **TopClass** (Santos 1999) Integra ferramentas de aprendizagem colaborativa, de entrega e gestão de conteúdo, e de gestão de pessoas. A ligação entre os participantes é feita através da Web. Este sistema possui uma funcionalidade de troca de mensagens entre os discentes-docente, bem como possibilita a participação em múltiplas listas de debate e atividades personalizadas para os discentes. Neste sistema, os cursos são

construídos pelo professor a partir de unidades de material de aprendizagem que podem ser facilmente exportadas ou importadas de curso para curso, nos quais podem ser criados testes de múltipla escolha.

- **AulaNet** (Santos 1999) Neste sistema, um curso é um conjunto de aulas direcionadas para a apresentação de conteúdos aos alunos. Os conteúdos podem ser apresentados como transparências, textos de aulas, vídeos e imagens. De modo a tornar mais rico o processo de aprendizagem, o sistema prevê a indicação de fontes complementares de informação. A interatividade do curso é garantida por uma série de serviços Internet de comunicação e cooperação entre alunos ou entre discente-docente, simultâneos ou não, tais como correio electrónico, listas e grupos de debate, sessões de *chat* e videoconferências.

2.4 Modelos de Desenvolvimento

Os programas educacionais podem ter o seu modo de aprendizagem em dois grandes focos, tal como foi referido anteriormente neste capítulo. Referimo-nos à aprendizagem por capacidades específicas e aprendizagem por capacidades cognitivas. Nesses grupos foram criadas taxonomias com o intuito de ajudar os programadores a perceberem as características base que o software educacional deve conter. Tendo em conta o âmbito desta dissertação, o foco deste tópico irá centrar-se na aprendizagem por habilidades específicas. Nos seguintes tópicos iremos abordar os sistemas CAI e os ICAI, que foram os antecessores do principal sistema (ITS) que iremos abordar nesta dissertação.

2.4.1 CAI

Estes representam o primeiro passo na criação de software educativo, em que o sistema seguia uma teoria comportamentalista (Skinner, 1958). Nestes sistemas era pressuposto que as pessoas atuassem por estímulos que recebiam e, à *posteriori*, respondiam a esses estímulos conforme o

ambiente. Ou seja, o sistema atuava apenas como um transmissor de conhecimento e o discente era o receptor, assumindo assim um papel passivo.

Na década de 50, surgiram os primeiros sistemas CAI que eram intitulados de programas lineares. Como o próprio nome indica, estes programas apresentavam o conhecimento de uma forma linear, ou seja, a ordem de ensino era pré-programada e em momento algum podia ser alterada após a sua criação. Seguindo esta abordagem o sistema apenas apresentava o conteúdo didático e não possuía a capacidade de seguir o aluno numa aprendizagem evolutiva, uma vez que este apenas acompanhava uma sequência de passos pré-determinados e, com isso, o software não estimulava o raciocínio dos discentes perante situações diferentes das apresentadas. Mais tarde foi pensado usar as respostas dos alunos para controlar o material de estudo, permitindo aos discentes o acesso ao esquema de resolução pessoal. Neste sentido, eles aprenderiam com base na análise das suas resoluções de problemas de um determinado grau de dificuldade, até encontrar uma forma organizada de resolução.

Mais tarde surgiram os sucessores dos programas lineares, ao qual se deram o nome de programas ramificados. Estes, forneciam um feedback que era adaptado ao ensino, sendo capazes de dar respostas aos alunos. A principal diferença destes programas para os antecessores é que eles eram capazes de atuar conforme a resposta do aluno, uma vez que continham uma técnica acrescida em relação aos anteriores, cujo nome é *Pattern-matching* e a utilização de linguagens de autoria. Isto permitia-lhe que considerasse as resposta do aluno como aceitáveis ou parcialmente aceitáveis. Ao contrário do antecessor que seguia a abordagem comportamentalista.

Com a evolução da informática e dos CAI, no final dos anos 60 e princípio dos anos 70, apareceram novos programas, sendo estes chamados de programas gerativos. Neste tipo de programas a filosofia educacional adotada deixou de ser a comportamentalista e partiram para um

novo paradigma. Nestes programas é defendido que os alunos são capazes de aprender melhor, enfrentando problemas com a dificuldade adequada, do que apenas assistir a explicações sequenciadas para adquirir o conhecimento. Com isto, o processo de aprendizagem é adaptado as necessidades do discente. Com os programas gerativos, houve um grande avanço nos CAI, pois os programas passaram a ser capazes de gerar um problema adaptando esse problema ao conhecimento do discente, e de ter a capacidade de construir a solução e diagnosticar a resposta do aluno.

Apesar dos CAI terem evoluído nos seus recursos gráficos, e com isso possuem melhorias significativas, estes possuem a mesma síntese pré-estabelecida pelo docente e não adaptável a cada tipo de aluno que o estiver a utilizar. Para além de bastante significativos, não possuem o conhecimento do domínio a ser ensinado, como o de um ser humano. Também não são capazes de responder a perguntas mais complexas dos discentes das tarefas a realizar, tais como o “porque” e o “como”.

Em suma, os algoritmos usados nos CAI evoluíram bastante e com isso o software educacional conquista um lugar mais aceitável pelos docentes para usar como complemento no seu processo educativo. No entanto, estes estão longe de serem comparáveis a um professor humano, já que são sistemas que apenas possuem um conhecimento limitado, não tão extenso quanto um professor humano, não conseguindo (ainda) ter em conta os processos cognitivos do discente.

Características

O CAI é um grupo de programas que foram denominados de “*Page-Turners*”, uma espécie de “virar a página” na história deste tipo de sistemas. Além disso, nestes sistemas o aluno é quem precisa de se adaptar ao sistema e não o sistema ao modelo de aprendizagem do aluno. Segundo Maite Urretavizcaya (2001), as características de um CAI são:

- Os cursos são muito extensos.
- A comunicação entre o tutor e o aluno não está muito bem definida.
- O conhecimento do como e porque se executam as tarefas de ensino agrupadas, isto é, os sistemas de ensino reagem segundo os modelos estabelecidos e com certa independência das atitudes e preferências do aluno concreto.
- O desenho e implementação dos sistemas são feitos sob medida.
- O conhecimento que inclui não se vê modificado com o tempo, ou seja, não evolui.

Tipos

Os sistemas CAI foram classificados de acordo com o método que neles foi projetado. Algumas dessas classificações são as seguintes:

- **Exercício-prática** - Estes sistemas apresentam as lições do domínio a ser ensinado e exercícios que permitem a verificação da aprendizagem realizada. As tarefas que são apresentadas concentram-se no fazer e no memorizar de informação que o sistema fornece. O discente só pode mudar de nível quando o resultado das tarefas anteriores for avaliado pelo sistema.
- **Jogos** - Este tipo de sistemas recorre vulgarmente a mecanismos muito diversos de entretenimento, nos quais os utilizador são constantemente desafiados nas suas capacidades cognitivas, crescendo-lhe uma motivação para a resolução do problema que lhe é apresentado, com base em algum conhecimento prévio. A competição neste tipo de sistema pode ocorrer entre o discente e o sistema. Porém pode ir um pouco mais longe e a competição ocorrer entre discentes, apesar de mediada pelo dispositivo.
- **Simulação** - Neste tipo de sistema podemos encontrar um mais limitado que outro. Podemos, encontrar sistemas em que o discentes escolhem a simulação ou as simulações

que pretende assistir. Outros são mais ricos e interativos, onde o próprio aluno cria o modelo do fenômeno, altera parâmetros e assiste à simulação resultante.

- **Demonstração** - Neste tipo de sistema a interatividade do utilizador é limitada ou até inexistente. Neste caso, o sistema assenta apenas num processo de disponibilização de informação pedagógica para consulta.
- **Tutoriais** – Os tutoriais apresentam a informação numa sequência pedagógica rígida, apesar de o discente poder escolher a informação que pretende visualizar. Neste tipo de sistema a interatividade do discente é reduzida, uma vez que esta se limita apenas à leitura de textos ou à observação de vídeos e de animações.

2.4.2 ICAI

Por volta dos anos 70 e 80, altura em que a Psicologia Cognitiva começou a questionar as suposições da teoria de aprendizagem comportamentalista, com a evolução da IA novas ideias começaram a ganhar força neste campo, como por exemplo, a ideia de processamento de informação simbólico e o processamento de linguagem natural, introduzido por Chomsky juntamente com *Newell* e outros (Collins et al., 1992). Este aparecimento suscitou algum interesse na comunidade de IA o que fez com que o processamento de informação se tornasse um paradigma dominante no final dos anos 70, início dos anos 80. Isto originou algumas preocupações por parte dos investigadores da IA acerca das falhas que os sistemas gerativos apresentavam. Para isso focaram as investigações no melhoramento de representar o conhecimento dentro de um sistema inteligente. Com esse intuito criaram os ICAI, que eram sistemas que se diferenciam na estrutura que apresentavam para trabalhar com os domínios educacionais, usando técnicas de IA e de Psicologia Cognitiva para orientar o processo educativo. Estes sistemas surgiram como uma espécie de sucessores dos CAI, com o objetivo de colmatar algumas falhas que existiam, e, além

disso, suprir a necessidade que existia de ser aplicado alguma inteligência aos produtos de software educativo criados até a data. Nestes sistemas:

1. A representação do domínio de conhecimento é mais completa.
2. As estratégias de ensino são independentes do conhecimento a ser ensinado.
3. Contem um modelo atualizado da performance do discente.
4. É permitido personalizar a instrução, adaptando ao nível de conhecimento e ao modo de aprendizagem do discente.

Capítulo 3

Sistemas de Ensino Inteligentes

3.1 Caracterização Base

Os ITS vieram revolucionar a forma de construir software educativo. Isto porque se pensou no modo como se poderia facilitar o método de ensino-aprendizagem, tornando-o mais moderno, correto e também mais agradável. Nesta perspectiva foi definido o método de aprendizagem através do aprender fazendo. Na literatura existe algumas definições relativamente a este tipo de método, como por exemplo:

“Os STI são programas de computador com propósitos educacionais e que incorporam técnicas de Inteligência Artificial. Oferecem vantagens sobre os CAIs (Instrução Assistida por computador), pois podem simular o processo do pensamento humano para auxiliar na resolução de problemas ou em tomadas de decisões” (Fowler 1991).

“Sistema Tutor Inteligente, é um termo amplo, abrangendo qualquer programa de computador que contem alguma inteligência e pode ser usado em aprendizagem” (Freedman et al. 2000).

Como já referimos anteriormente, o termo ITS, que agora caracteriza estes sistemas, foi inicialmente conhecido por ICAI. O “estado da arte” destes foi revisto devido às suas características base, bem como à sua construção, que é bastante diferente dos antecessores CAI e como tal ser necessário distingui-los. Com esse intuito, Sleeman e Brown (1982) procederam à sua revisão intitulando-os de ITS.

O principal objetivo de um ITS é proporcionar um sistema no qual o ensino é adaptado ao aluno. Ou seja, com este tipo de sistema pretendemos que o dispositivo tecnológico seja capaz de adoptar uma forma inteligente e de passar a informação do domínio a ser ensinado ao aluno, simulando o mais próximo da realidade de um professor (humano), apesar de existir, obviamente, muitas limitações a esse nível. Mas com a evolução na área de Interação Homem-Computador (IHC) essa realidade tem sofrido bastantes progressos. Veja-se, por exemplo, o seguinte:

“A utilização de técnicas de IA na elaboração e no desenvolvimento de ambientes de ensino-aprendizagem computadorizados tem se constituído em objeto de investigação por parte dos pesquisadores da área de informática aplicada à Educação, devido as suas potencialidades. A utilização de agentes na modelagem e no projeto de STI permite-nos resgatar antigos problemas em aberto, como por exemplo, a melhoria da interação entre tutor e aluno e a possibilidade de investigação dos processos mentais em nível mais estratificado.” (Giraffa 1999).

Com o intuito de assimilar o comportamento de um professor, os ITS adoptaram técnicas de IA e psicologia cognitiva, o que ajudou a revolucionar este tipo de software educativo em relação aos antecessores CAI (Figura 1). Além disso, existem outras diferenças de destaque entre os ITS e os CAI que se encontram centradas na forma de como construir o projeto. Nos sistemas CAI, usualmente, o aluno é conduzido a uma resposta “correta” seguindo um conjunto de procedimentos previamente projetados. Por seu lado, os ITS apresentam como objetivo simular algumas das capacidades cognitivas do aluno e utilizar esses resultados como base das decisões pedagógicas a tomar. Na tabela 2 podemos ver expressas algumas diferenças entre esses dois tipos de sistemas - CAI e ITS:

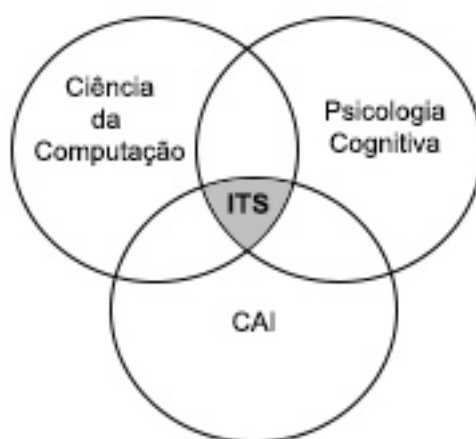


Figura 1 - Áreas abrangidas pelos ITS.

	CAI	ITS
Origem	Educação	Computação
Fundamentação Teórica	Behaviorista	Psicologia Cognitiva
Estruturação e	Única estrutura algorítmica pré-	Estrutura dividida em vários

funções	definida, onde a apresentação do conteúdo é uma sequência estática	módulos, onde a sequencia varia conforme as respostas do aluno
Estruturação do conhecimento	Algorítmica	Heurística
Modelagem do aluno	Apenas a última resposta do aluno conta, e é descartado as dúvidas existentes.	É feita uma avaliação da interação com o aluno
Modalidades encontradas	Tutorial, exercício e prática, simulação e jogos interativos	Socrático, ambiente, interativo, diálogo bidirecional e guia.

Tabela 2 – Comparação de características dos sistemas CAI e ITS.

Em suma, de acordo com (Nwana, 1990) as maiores diferenças entre os *softwares* ITS e os CAI são:

- Os ITS fornecem uma clara articulação do domínio de conhecimento a ser ensinado.
- Um ITS possuem um modelo com a performance do estudante que é dinamicamente mantido e é usado para conduzir a instrução.
- Os “programadores” de ITS definem o conhecimento e as regras de dedução, mas não a sequencia de procedimentos no processo educativo que serão direcionadas pelo software;
- ITS fornecem um diagnóstico detalhado dos erros, ao invés de exercitarem apenas a pratica.
- Os ITS dão a possibilidade do discente fazer questões ao sistema (uma das suas principais características).

Quanto à arquitetura conseguida nos ITS, esta permite uma maior individualização no processo de aprendizagem. Neste sentido, os ITS devem ter a capacidades de modelar o entendimento do discente sobre um determinado tópico apresentado e comparar o conhecimento do discente com o

de um “especialista” do domínio a lecionar. A flexibilidade de trabalho dos ITS torna-os inteligentes, pois estes devem ser capazes de aprender através do meio ambiente onde a base do conhecimento será atualizada, isto no decorrer da interação do discente com o sistema.

Em jeito de conclusão, a arquitetura dos ITS são modulares e baseados em conhecimento especializado. Aí o domínio da aplicação está representado e modelado por algum tipo de formalismo usado em inteligência artificial. Além do grande avanço que estes sistemas sofreram relativamente aos sucessores a nível de arquitetura, fornecem uma instrução adaptada ao discente. De referir que para quantificar valores cognitivos, a interação com o discente é factor importante e determinante neste sistema.

3.2 Propriedades de um ITS

Quanto às propriedades dos ITS algumas já foram mencionadas anteriormente. Além das referenciadas existe uma propriedade com um papel preponderante: a interface de comunicação. Qualquer módulo que sustente esta propriedade deve ser bem arquitetado e, de certa forma, ligado com o módulo de domínio. Além disso, deve ser um módulo intuitivo, de fácil compreensão e com bastante importância no processo de comunicação docente-discente. Como já foi referido, uma das propriedades que identificam estes sistemas é a sua arquitetura modular, uma vez que esta permite que seja feita uma separação do conhecimento que se pretende ensinar (modelo do domínio) e as suas estratégias de ensino (modelo tutor). Para além disso ainda é capaz de caracterizar o discente (modelo do aluno), com o intuito de personalizar e adaptar o método de ensino. Tal como referido em (Gavidia e Andrade, 2003) as propriedades dos ITS podem resumir-se em:

- Dá-se o adjetivo de “inteligente” para contrastá-lo com sistemas tradicionais de instruções assistidas por computador (CAI), sendo uma diferença marcante o uso de técnicas de Inteligência Artificial nos ITS.

- Uma razão para atribuir “inteligência” a estes sistemas, está na sua capacidade de resolver os problemas que apresentam aos alunos e de explicar a sua resolução.
- Ao contrário dos sistemas CAI tradicionais, os ITS permitem um maior grau de individualização na instrução. Em particular, um ITS relaciona a instrução com o entendimento das metas e crenças do aluno.
- Num sistema CAI não inteligente, a ordem e o plano de interação estão pré-definidos, enquanto que num ITS se usam técnicas de IA tais como o planeamento, a otimização e buscas, deixando que o sistema decida “inteligentemente” a ordem de apresentação do conteúdo ao aluno.
- A interação pode ser muito variada num ITS. Desde os sistemas passivos ou reativos que dependem completamente da realização (ou não) de uma ação bem determinada pelo aluno, até os que constantemente apresentam nova informação, como os assessores que observam o aluno enquanto este faz uma tarefa, não interferindo constantemente, mas explicando ou ensinando um conceito num momento importante ou quando o aluno pede.
- Recentemente, tem-se ampliado o uso de interfaces mais sofisticadas, que reforçam a disponibilidade e a facilidade do uso de Sistemas Tutores Inteligentes. Duas tendências importantes neste sentido são a utilização da multimídia e da WWW.

3.3 A Arquitetura

Relativamente à arquitetura de um ITS, esta constitui um dos aspetos claros para a sua diferenciação dos seus sucessores: os sistemas CAI. Usualmente a arquitetura de um ITS é constituída por 4 módulos. Com a divisão desses módulos foi conseguido separar o conhecimento a ser ensinado do método de ensino a ser usado. Existe ainda um módulo que contém a informação do discente, para que, a posteriori, se possa tomar decisões quanto ao método de ensino a ser usado.

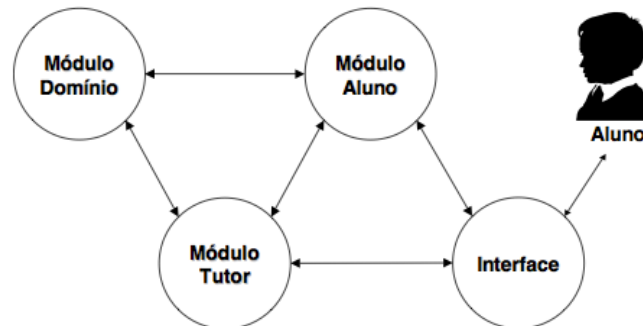


Figura 2 – A Arquitetura Clássica dos ITS (Giraffa e Rodrigo Rafael Villarreal Goulart 2001)

Na Figura 2 podemos visualizar uma ilustração da arquitetura base, um modelo consensual segundo vários investigadores da área (Nwana 1990), (Freedman et al.. 2000) e (Nkambou et al. 2010) e que é constituída pelos seguintes módulos:

- **Modelo do aluno.** Neste módulo é onde está contida a informação relativa as características do aluno.
- **Modelo do tutor.** Neste módulo é armazenado o conhecimento sobre as estratégias e métodos de ensino do domínio e as diversas formas como adaptá-lo ao aluno.
- **Modelo do Domínio.** Neste módulo está contido todo o conhecimento sobre o domínio a ser ensinado, codificado em regras de produção, estereótipos, etc.
- **Modelo da Interface.** Este módulo é responsável por intermediar a interação entre o aluno e o tutor.

A arquitetura apresentada é a arquitetura tradicional, também conhecida como função tripartida. O facto de usar o termo tripartido refere-se a funções associadas aos modelos de tutor, aluno e do domínio. Posteriormente é apresentada uma descrição mais pormenorizada dos modelos e de algumas das técnicas que lhe serão aplicadas. A saber:

- **Modelo do Aluno.** Geralmente, o modelo analisa as interações do estudante com referência ao modelo de domínio. Isto acontece com o objetivo de detetar falhas de conhecimento e erros na intervenção instrutiva do aluno com o sistema. Do ponto de vista construtivista, o modelo deve registrar todas as interações do aluno, considerando o contexto na qual estas ocorrem, bem como as suas estruturas cognitivas. Além disso deve ser capaz de registrar as mudanças de perfil do mesmo, em que irá ser representado o conhecimento e as suas capacidades cognitivas. Para isso, são usados dados estáticos e dinâmicos (Vicari, 1989), que assumem um papel relevante para o tutor comprovar hipóteses a respeito do aluno. Com este modelo, o sistema é capaz de definir a melhor estratégia de ensino a ser usada para o aluno em questão. Para que este modelo se torne próximo da realidade é necessário uma dinâmica constante na atualização do mesmo, consoante a avaliação do desempenho feita ao aluno. Quanto aos dados dinâmicos, estes avaliam o desempenho do aluno quando lhes são apresentadas questões pelo tutor e confrontadas com hipóteses destacadas pelo aluno. Também avaliam o uso que o mesmo faz do sistema, bem como os novos conhecimentos que pode ensinar ao tutor. Para poder construir este modelo podemos recorrer a várias técnicas, tais como: colocar preferências do aluno, as suas metas a alcançar, aplicar reconhecimento de padrões ao histórico de respostas fornecidas pelo aluno, entre outras.
- **Modelo do Tutor.** Este modelo representa o conhecimento pedagógico do sistema, onde por vezes é intitulado de Modelo pedagógico. Este será responsável por selecionar o conteúdo a ensinar e quais as táticas para o fazer. Além disso é responsável por fazer um acompanhamento do discente no processo educativo e intervir quando solicitado, assemelhando-se a um tutor humano. Para isso, as suas decisões serão sustentadas pelo perfil do aluno que é avaliado no processo de diagnóstico realizado no modelo de aluno, tornando-o fortemente dependente. Existem alguns modelos teóricos que são empregues

nos ITS, nomeadamente o método socrático, o modelo *coaching* e o modelo de hipertextos.

De forma mais detalhada:

- **Método Socrático.** Neste método o tutor regula o seu processo educativo através de diálogo e perguntas, direcionando o aluno a atingir as suas próprias conclusões. Posto isto, esse processo educativo baseia-se nos conhecimentos que o aluno já domina.
 - **Modelo *Coaching*.** Neste modelo o conhecimento que o aluno adquire é uma consequência indireta das atividades que são empregues neste modelo, jogos que transmitem conceitos relacionados e atividades de entretenimento. Neste caso são usadas estratégias pedagógicas de orientação e de cooperação.
 - **Modelo Hipertexto.** Neste modelo o aluno poderá assumir um papel mais participativo e dinâmico, no qual é orientado no processo educativo de uma forma diversificada, uma vez que é permitido, no mesmo documento, albergar diferentes formas de representação do conhecimento. Em suma, este modelo tem um factor importante, pois ele irá manipular todo o processo educativo e, juntamente com o modelo de aluno, adaptá-lo ao utilizador. Com isso pretende-se chegar a um tutor virtual mais real, em que o tutor humano adapta a sua estratégia em função de múltiplos *inputs* que recebe dos seus alunos numa sala de aula. Para isso a sua tomada de decisão é baseada na sua experiência e instinto.
- **Modelo do Domínio.** Este é responsável por manipular o conhecimento que é pretendido ensinar, ou seja representa o conhecimento do especialista. Este modelo deve ser compatível com o raciocínio do estudante, ou seja, está ligado com o modelo do aluno. Para modelar o conhecimento existe alguns formalismos, como, por exemplo, as redes semânticas, scripts, regras de produção, entre outros. Quanto a escolha do formalismo deve basear-se em qual o método que melhor e mais facilmente atenda os requisitos de representação e manipulação do raciocínio. Portanto, este módulo armazena a informação

que o tutor irá lecionar. Para a representação do conhecimento deve ser tido em conta o crescimento incremental do domínio e sua respetiva atualização.

- **Modelo da Interface.** Este módulo tem como responsabilidade interagir de forma direta com o estudante. Para isso deve ser constituída uma boa interface que é considerado, com sabemos, um ponto fulcral no sucesso de todos os sistemas interativos. Além dessa característica, o sistema deve ser de fácil manuseamento, com tempos de resposta razoáveis e uma representação clara e intuitiva do conhecimento que possui. Ou seja, este modelo além das características que deve ter é responsável pela apresentação dos recursos educativos e pelo controlo do progresso do aluno através das respostas que o mesmo aluno apresenta, no qual essa monitorização deve ser realizada de forma “escondida” para evitar formulários extensos. Porém é necessário que exista uma facilidade que permita intervir no diálogo de ambas as partes, por exemplo o aluno intervir no discurso do tutor e vice-versa. À semelhança dos modelos anteriores, neste existe também algumas técnicas para a transmissão do conhecimento. Tais como os sistemas de hipertexto e *hipermédia*, na qual os sistemas hipertexto remete-nos a um texto, onde é agregado conjuntos de informação na forma de blocos de texto, palavras, imagens ou sons, em que o acesso a estes ocorre através de referências específicas, conhecidos no meio digital como *hyperlinks*, ou simplesmente links. Devido à panóplia de recursos que as técnicas de hipermédia nos fornecem, e ainda e termos de semântica do conteúdo, estas fazem destes sistemas uma ferramenta importante e bastante vantajosa para a apresentação de conhecimento em ITS.

Em suma, para além desta arquitetura trazer várias vantagens pelo facto acolher a modelação de um qualquer processo educativo e, assim, permitir uma melhor gestão e adaptabilidade desse processo, os módulos encontra-se fortemente ligados para que os ITS se aproximem o máximo da realidade.

3.4 Modelação

Quanto à modelação dos ITS, segundo Giraffa, Rodrigo Rafael Villarreal Goulart (2001) encontra-se presente na literatura algumas abordagens que são seguidas, no entanto as mais abordadas são a clássica e a que se baseia em agentes. Na clássica segue uma arquitetura como a referida anteriormente, já a de agentes substitui os módulos que existe na arquitetura clássica por um conjunto de agentes.

3.4.1 A Modelação clássica

Na abordagem clássica as decisões pedagógicas do tutor assentam em regras que são definidas com o intuito de representar o seu conhecimento e perícia. Tal como apresentado em (Giraffa e Rodrigo Rafael Villarreal Goulart 2001) sustentado também em (Wenger, 1987), na abordagem clássica o papel do módulo tutor é tomar as decisões pedagógicas em função das interações com o aluno. Estas decisões derivam de regras ou estruturas de conhecimento que representam o conhecimento do tutor a respeito do domínio, e estas estão representadas de forma definida no sistema, no seu módulo tutor.

Porém, neste modelo encontra-mos algumas limitações, como por exemplo a limitação existente relativamente à quantidade de regras que podem ser definidas na representação do conhecimento. Outra limitação reside no facto da abordagem estar limitada a um único método de ensino, o que delimita bastante quando o comparamos a um tutor humano. Apesar desta última limitação ser característica primordial nesta abordagem, relativamente ao seu comportamento pedagógico, esta não proporciona uma adaptabilidade a nível de estratégias consoante as características individuais de aprendizagem de cada aluno.

3.4.2 Modelação com Agentes

Nesta abordagem os ITS usufruem das abordagens da IA distribuída, em particular da aplicação de Sistemas Multiagente. Com isso os ITS herdam algumas características dos agentes, em particular: os agentes têm a capacidade de agir e tomar decisões de forma autónoma, com o intuito de alcançar os seus objetivos, e o facto de os agentes serem capazes de interagir com outros agentes, usando protocolos de interação inspirados no ser humano, tais como: coordenação, cooperação, competição e negociação. Tendo em conta estas características dos agentes, os ITS tem ganhos significativos a nível da comunicação e interação que o sistema é capaz de fornecer. Segundo (Giraffa, 1999) a razão fundamental para modelar um ITS com uma arquitetura multiagente são as suas capacidades de comunicação e interação, bem como a sua capacidade de se adaptarem e aprenderem durante uma sessão educativa.

Com uma abordagem orientada ao agente podemos obter diferentes formas de atuação. Os agentes podem atuar como tutores, colegas ou alunos virtuais. Além disso, podem ser criados vários grupos de agentes, pelos quais será distribuído um conjunto de tarefas. Estas tarefas podem ser muito abrangentes, assentando a sua definição na modelação do próprio aluno ou na seleção de estratégias e táticas de ensino. Além da sua forma de atuação, os objetivos dos agentes podem ser descritos consoante as estratégias que irão abordar no processo educativo. Giraffa e Viccari (1999) apresentaram alguns comportamentos para um agente, nomeadamente:

- Guia, que conduz o aluno na resolução do problema, em todo o processo educativo, fazendo a sua respetiva monitorização.
- Assistente, que conduz o aluno na resolução de problemas com base em processos de descoberta, usando heurísticas sobre o processo de resolução no domínio em questão. Além disso é capaz de executar a respetiva monitorização do processo.
- Facilitador, que é um agente capaz de assumir um papel de apoiante na resolução de um problema, pois tem como função fornecer dicas sobre a sua resolução, intervindo

apenas quando é solicitado. Porém, assume o mesmo papel que os outros tipos de agentes na monitorização do aluno.

Uma das vantagens dos agentes é que os tutores não têm que ter um comportamento estático. Estes podem adotar vários comportamentos consoante as informações que recebem das interações do aluno e o objetivo que se pretende atingir. Em suma, a modelação por agentes trouxe ainda mais realce aos ITS, uma vez que estes ajudam a criar a simulação de um ambiente mais real no processo educativo, podendo-se mesmo, com um grupo de agentes específico, criar um cenário parecido com o de uma sala de aula. Aí, os agentes podem assumir um papel de tutores, alunos e colegas de sala onde todos eles interagem entre si.

3.5 Exemplos

Desde cedo que existem sistemas que foram implementados em vários domínios de ensino. Segundo (Nwana 1990) existe alguns projetos, tais como:

SCHOLAR (Carbonell, 1970) O programa SCHOLAR foi o projeto pioneiro na tentativa de inclusão da modelação do conteúdo. O conhecimento do sistema é representado em uma rede semântica cujos nodos significam objetos e conceitos geográficos sobre a América do Sul. Carbonell achava que uma rede semântica completa, podia ser utilizada para modelar o conhecimento do estudante. Assim, ele sugeriu uma rede completa para modelar o “estudante perfeito” e progressivamente perturbá-lo para refletir o seu desempenho até aquele momento, apagando e até mesmo modificando nodos e ligações. O modelo do estudante no sistema SCHOLAR, em termos de avaliações ligadas aos conceitos individuais de uma representação modular do conhecimento especialista, constitui uma primeira versão do que foi posteriormente chamado o método “*overlay*”. Um detalhe interessante neste sistema, é que o estudante pode alterar a qualquer momento o modo de controle para o diálogo (teclando “Q/A”). Assim o estudante pode solicitar ao sistema

simplesmente responder suas questões, conduzir um diálogo de iniciativa mista, ou assumir o comando e fazer perguntas. O sistema SCHOLAR representou uma grande descoberta nesta área. Com seu conhecimento de domínio representado como uma rede semântica, o sistema faz uso de procedimentos de inferência para uma interação tutorial simples.

WEST - O projeto WEST (Burton e Brown 1979) foi iniciado no contexto do SOPHIE (Brown 1982) . Uma vez que o domínio da eletrônica parecia muito complexa para uma primeira investigação da arte de treinar, o domínio escolhido para este propósito foi o jogo educacional para computador chamado "How the WEST was won", desenvolvido para o projeto PLATO Elementary Mathematics Project. Este domínio estava de acordo, portanto, com o conceito de ambiente de aprendizagem reativo. O propósito do jogo é exercitar técnicas de aritmética. Para acompanhar todas as necessidades do programa: determinar o que o estudante conhece, quando interromper, e o que dizer, Burton e Brown (1979) adicionaram um componente ao sistema – o *Coach* -, porque ele agia como um treinador, observando e auxiliando o jogador a melhorar. O *Coach* desenvolvido para o WEST baseava-se na ideia de questões e exemplo. Questões são conceitos usados no processo de diagnóstico para identificar, à qualquer momento, o que é relevante. Exemplos são instâncias concretas destes conceitos abstratos. O fundamento é que a melhor maneira de aumentar a técnica do estudante, ilustrando a ajuda educacional com exemplos concretos. Neste sistema havia três níveis de questões : capacidades matemáticas, regras e estratégias ao WEST, e capacidades relevantes para jogar, tal como a aprendizagem a partir do oponente. A maioria das informações sobre as quais o *Coach* baseava as suas decisões fundamentando-se na comparação das interações do jogador com o que o especialista faria nas mesmas circunstâncias. Portanto, o WEST continha, não somente uma representação do conhecimento do especialista, mas também técnicas tutoriais para utilizar a informação produtivamente.

Hydrive. O HYDRaulics Interactive Video Experience (KAPLAN, R.; ROCK 1995) é um ITS que incorpora multimídia para resolver problemas de sistema hidráulico de um avião F-15. Apesar do HYDRIVE utilizar um disco laser externo para suportar imagens de vídeo, o conteúdo do disco laser

(sequência de animações e imagens) é representado na base de conhecimento utilizada pelo sistema. O propósito deste ITS é dar instrução aos técnicos de voo para a solução de problemas complexos. No HYDRIVE, o material multimídia é disponível para o sistema através da codificação de uma representação do conteúdo na forma de regras na base de conhecimento. Quando uma sequência em particular necessita ser mostrada, ela pode ser localizada através desta representação. Esta incorporação de vídeo no HYDRIVE satisfaz as exigências de um ITS para proporcionar uma boa instrução aos funcionários sendo treinados. Os modelos definem o que o sistema deve fazer após a ação da pessoa em processo de treino, e a resposta do sistema pode então ser interpretada como uma determinada meta de apresentação. Quando a implementação do sistema HYDRIVE começou, havia poucas alternativas para a implementação de um ITS como este. Atualmente existem novas opções para a criação de sofisticadas interfaces, pois a ferramenta escolhida para a criação de um ITS deve simplificar este processo ao máximo. Além dos projetos apresentados anteriormente existe mais alguns projetos nesta área. Em jeito de conclusão apresentamos na tabela 3 vários sistemas ITS diferentes, revelando algumas das suas características.

Sistema	Domínio	Representação do conhecimento	Modelo do Aluno	Estratégia do Tutor	Referência
Scholar	Geografia	Rede Semântica	“Overlay”	Socrático	(Carbonell 1970)
Why	Tempo (Causas da chuva)	“Scripts”	Bugs	Socrático	(Collins et al. 1992)
Sophie	Detector de falhas eletrônicas	Rede Semântica	“Overlay”	Ambiente reativo com interações	(Brown 1982)

West	Expressões Aritméticas	Regras	“Overlay”	Ambiente reativo com treino	(Burton & Brown 1979)
Guidon	Medicina	Regras	“Overlay”	Ambiente reativo com interações estruturais	(Clancey 1984)
Proust	Programação em Pascal	Rede Semântica	Bugs	Ambiente reativo com orientação	(Soloway & Johnson 1984)
Hydrive	Falhas Sistemas Hidráulicos de Avião	Regras	—	—	(KAPLAN, R.; ROCK 1995)

Tabela 3 - Tabela Comparativa de Sistemas ITS.

Atualmente existem alguns projetos de sucesso no campo dos ITS, como é o caso do projeto referido a seguir:

PAT2Math (*Personal Affective Tutor to Math*) (Fábio Rafael Damasceno 2010)– este sistema é um dos projetos mais atuais que é enquadrado nos ITS. Tem como objetivo, auxiliar os alunos na aprendizagem para resolução de problemas algébricos. Este sistema é capaz de resolver qualquer equação de primeiro e segundo grau, além disso é capaz de corrigir a resolução apresentada pelo aluno e sugere dicas de resolução quando este fornecer uma resposta incorreta ou quando não souber como deve resolver. Este sistema utiliza diversas técnicas de inteligência artificial de modo a permitir um apoio personalizado aos alunos. Este sistema está a ser desenvolvido no PIPCA/UNISINOS, no grupo de inteligência artificial aplicado á educação.

Capítulo 4

O Sistema da Cartilha

4.1 Caracterização base

Nesta dissertação pretendemos contribuir para a criação de um assistente pessoal para suporte ao ensino da leitura do português a crianças, tendo como base o método de ensino da Cartilha Maternal João de Deus. Nesse sentido, decidimos desenvolver esse sistema num ambiente móvel, mais concretamente em ambientes suportados por plataformas computacionais do tipo *tablet*, dada a sua grande expansão e adoção pelos utilizadores em termos gerais e, em particular, em ambientes educativos, bem como ao seu acesso, cada vez mais facilitado, por parte do público-alvo a esse tipo de plataformas.

Para atingir o objetivo pretendido torna-se necessário, enquadrar este software na categoria de software educativo, mais em concreto em sistemas ITS. Nesse sentido foi criado um protótipo inicial no qual algumas das lições do método de ensino referido foram projetadas no sistema a ser desenvolvido. Através do estudo que foi feito inicialmente sobre o método de ensino que será projetado, verificamos a necessidade de implementar nos sistemas, alguns serviços que são

importantes para a informatização do método, tais como: reconhecimentos de caracteres, reconhecimento de voz e sintetizadores de voz. Tendo em atenção esses factores, e algumas das limitações presentes neste projeto, a escolha do sistema operativo para desenvolvimento do sistema para plataforma computacionais do tipo *tablet* recaiu sobre o sistema Android. No decorrer deste tópico iremos abordar o domínio da aplicação, o ambiente em que foi desenvolvido, bem como os serviços que são usados neste protótipo.

4.1.1 A Cartilha Maternal João de Deus

O método da Cartilha Maternal de João de Deus é o método que servirá de base para o protótipo a ser desenvolvido. A Cartilha Maternal foi criada pelo poeta e pedagogo João de Deus, que nasceu a 8 de Março de 1830 em São Bartolomeu de Messines (Algarve – Portugal). João de Deus era sensível aos problemas da atualidade, nomeadamente ao analfabetismo. Nesta linha, defendia que:

“Ser homem é saber ler. E nada mais importante, nada mais essencial que essa modesta e humilde coisa chamada – primeiras letras.” (Deus, 1877)

Segundo João de Deus, toda a gente devia ter acesso a aprender a ler, ou seja, além de todos os problemas sociais como fome e pobreza, aprender a ler deveria ser um direito do ser humano. Em 1876, a publicação da Cartilha Maternal veio revolucionar o ensino e “destruir” velhos métodos existentes. Tal como referenciado por (Ruivo, 2006) sustentado na opinião de Dona Carolina Michaelis de Vasconcelos, escrita no jornal do Colégio Portuense O Ensino, em 1877:

“com a Cartilha do Senhor João de Deus entramos num mundo novo; tudo mudou de aspecto, tudo se tornou simples, lúdico, transparente. O novo pedagogo vai guiando o discípulo passo a passo; não o mete num labirinto; apresenta-lhe um plano disposto na melhor ordem e assenta no seu lugar, uma a uma, as pedras do edifício, que são os elementos da língua. Dá a conhecer as letras uma por uma, assim como a sua

aplicação e só no fim constitui a cadeia do alfabeto, ligando estes seus elos; não desmembra as palavras em sílabas, as sílabas em letras, apresenta à criança a flor intacta.”

Isto comprova a revolução do método de ensino na altura. No entanto, para além de ser uma afirmação passada, esta encontra-se bastante atual, e devidamente enquadrada com o âmbito desta dissertação. Pois, neste trabalho é pretendido levar este método de ensino para um mundo novo, o mundo digital, tornando-o mais lúdico, interativo e simples, adaptando este mundo para a realidade de hoje. Para isso é criado um tutor virtual que o irá guiar ao longo do seu percurso no método educativo implementado no software, apoiando e orientando o aluno na construção do conhecimento. Porém, na criação do método da Cartilha Maternal, João de Deus não se preocupou apenas só na informação que ia passar, mas também no modo como a iria passar, visto que foi inovador no que se refere aos aspetos visuais do método, sendo esta, uma das principais características que identifica este método. Pois é feita uma divisão silábica das palavras sem quebrar o seu aspecto gráfico e sonoro, para isso ele recorre ao preto/cinzeno. A divisão tem por objetivo recusar o tratamento das sílabas independentemente das palavras em que estão inseridas, de modo a que seja ensinado o código alfabético num contexto de leitura com significado.

João de Deus defendia o uso do livro da sala de aula, pois assim permitia uma maior interação no ato da leitura, para qual o método foi criado. Tal como é afirmado por (Ruivo 2006):

“O uso de um livro grande na sala de aula é uma ideia original de João de Deus e permite que a criança tenha uma maior e mais activa participação no acto de ler em voz alta, desenvolvendo-lhe o conceito de leitura e habilidades pró-leitoras. Desenvolvendo um método que permitia «massificar» o acesso à leitura, João de Deus não esqueceu a criança na sua individualidade e, por isso, com necessidades educativas particulares.

Desta forma, cada criança segue a Cartilha a seu ritmo próprio (e não ao ritmo da classe)."

O nosso projeto enquadra-se neste âmbito, uma vez que se pretendemos transferir a Cartilha Maternal para formato digital, privilegiando a interação do aluno com o nosso "livro digital". Nessa interação o sistema deve avaliar o aluno e, consoante essa avaliação, deve ir adaptando o método ao perfil do aluno, fazendo com que o aluno siga a cartilha ao seu ritmo e não ao da turma na qual se encontra inserido. Além de ser um método já com alguns anos de existência, podemos comprovar que não deixa de ser um método inovador e atual, pois em alguns aspetos como a divisão gráfica das sílabas, ainda não existe concorrentes para esse efeito. Mais do que isso, podemos verificar que o método se encontra bastante atual, pois ainda hoje é usado por algumas escolas. Veja-se como exemplo o caso noticiado em (Expresso, 2014) acerca do jardim-escola João de Deus, no bairro de Alvalade. Nessa notícia reporta-se o facto de que as listas de espera para essa escola são grandes, devendo-se isto, essencialmente, aos resultados que a escola obtém seguindo o método da Cartilha Maternal, que o classificam como uma disciplina rigorosa e com conteúdos exigentes. Porém Cristina Lázaro afirma nessa mesma notícia que:

"As crianças saem de cá com muitas capacidades para as engenharias. Até houve uma altura em que éramos chamados a escola dos génios".

Por fim é comprovado pela diretora desta escola que o grupo de alunos dos 5 anos acabam o ano letivo a saber ler. Apesar de expormos o caso desta escola, a rede de escolas que utiliza este método ainda é bastante alargada, como podemos ver na Tabela 4.

Rede de Jardins Escolas João de Deus					
Albarraque	Coimbra 2	Figueira da Foz 2, Centro Infantil	Lisboa, Olivais	Ponte de Sôr	Tomar 2
Alcobaça	Coimbra, Centro Infantil	Funchal	Matosinhos	Porto	Torres Novas
Alhadas	Entroncamento	Leiria	Mortágua	Santarém	Torres Vedras
Braga	Estarreja	Lisboa, Alvalade	Mortágua, Centro Infantil	Santo Tirso	Tramagal
Castelo Branco	Faro	Lisboa, Centro Infantil 1	Odivelas	São Bartolomeu de Messines	Urgeiriça
Chaves	Figueira da Foz 1	Lisboa, Centro Infantil 2 / Creche Familiar	Penafiel	Tavira	Vila Nova de Gaia
Coimbra 1	Figueira da Foz 2	Lisboa, Estrela	Ponta Delgada	Tomar 1	Viseu

Tabela 4 - Jardins Escolas que leccionam com base na Cartilha Maternal João de Deus.

Quanto à escolha deste método para o modelo de domínio nesta dissertação, baseia-se em todos os factores apresentados até agora e, ainda, com o facto de se poder validar o sistema perante uma

escola que leciona o método em questão. No entanto é um domínio em que pouco ou nada existe em termos de software educativo e, como tal, poderá ser uma mais-valia para renovar e fornecer uma nova ferramenta de apoio a este método. Algumas das características do método de João de Deus apresentadas por (Ruivo, 2006) identificam-se com os sistemas ITS, tais como:

- **Respeita o ritmo individual de cada criança**, ou seja este método adapta-se conforme o progresso da leitura do aluno. O ritmo de aprendizagem do método não varia em torno da turma na qual o aluno se encontra inserido. O que acontece nos sistemas ITS, em que o progresso do aluno é avaliado consoante a sua progressão, sendo o método de aprendizagem adaptado ao aluno, ajudando-o na aquisição de conhecimento.
- **Estimula as capacidades cognitivas**. Neste método a criança é conduzida como se de um jogo se tratasse, no qual vai aprendendo as regras e evoluindo, adquirindo gradualmente conhecimento. No discurso com o sistema a criança insere conhecimentos seus na aprendizagem, construindo assim o vocabulário e a construção frásica de uma forma lúdica. No entanto podemos relacionar esta propriedade do método com os ITS pois eles também estimulam as capacidades cognitivas, com o intuito de adaptar o método de ensino a cada aluno e levar o aluno a construção do conhecimento através da exploração do domínio apresentado.
- **Fomenta a criança a autocorreção**. A criança ao longo de todo o processo de aprendizagem é estimulada a ser analista da linguagem, conseguindo autocorrigir-se através da explicitação de regras. O mesmo acontece nos sistemas ITS, em que o aluno é seguido em todo o processo de aprendizagem com o objetivo de o estimular e ajudar no progresso educativo, simulando a interação entre o aluno e o tutor humano.

Quanto à organização do método da Cartilha Maternal de João de Deus, esta é composta por 25 lições (Ruivo, 2006):

- **1ª Lição** - **esta** tem como principal objectivo dar uma noção fundamental, a compreensão do que é a letra, tirando todo o aspecto abstracto à figura e dando uma forma quase corporizada ao símbolo linguístico.
- **2ª Lição** – esta introduz a primeira consoante, “V”, uma constrictiva fricativa, cujo valor se pode proferir e prolongar; não damos o nome mas a sua leitura.
- **3ª Lição** – aqui apresenta-se a letra “F”, cujo valor se pode proferir e prolongar.
- **4ª Lição** – em que se apresenta a letra “J”; à semelhança das lições anteriores, ensinar-se-á à criança apenas o valor, a leitura da letra, a forma como soa antes da vogal com a qual formará sílaba e palavra.
- **5ª Lição** – esta lição apresenta a letra “T”, uma consoante oclusiva, que não tem leitura proferível.
- **6ª Lição** – nesta lição, o discurso a ter com os alunos é semelhante: «Também esta é uma consoante oclusiva “D”, o que quer dizer que, não tendo leitura proferível, para lermos palavras com esta letra, temos de fazer o ponto de articulação – língua nos dentes com pouca força – e juntar a vogal que se segue – exemplo: di a.»..
- **7ª Lição** – nesta parte do processo de ensino ensina-se a letra “B”, que se lê unindo os lábios – é uma bilabial, quanto ao ponto de articulação e oclusiva quanto ao modo de articulação.
- **8ª Lição** – nesta lição apresenta-se mais uma consoante oclusiva cuja leitura não se pode proferir nem prolongar a letra “P”.
- **9ª Lição** – aqui apresenta-se a letra lê, “L”; esta letra lê-se colocando a língua no céu-da-boca (alvéolos); lemos algumas palavras, dentro do critério já indicado e das regras já dadas, que se vão sempre repetindo.
- **10,11,12ª e primeira parte da 13ª Lição** – neste conjunto de lições não se apresentam letras, apenas regras de leitura: dizemos que o “O” no fim de palavra se lê [u],

que o “E” no fim de palavra se lê fraco; o aluno aprende ainda os ditongos ou e ei, bem como o nome e a função do acento circunflexo.

- **13ª Lição segunda parte** – aqui a criança aprende uma série de informações que lhe vão permitir ler e escrever corretamente; vai aprender exatamente como deve ler as vogais tónicas e átonas, o que lhe permitirá ler a palavra com a acentuação certa.
- **14ª Lição** – nesta lição aprende-se a última consoante, a que João de Deus chamou de «certa», ou seja, uma consoante com apenas um valor, uma leitura; esta letra chama-se quê [k] e lê-se com a língua encolhida.
- **15ª Lição** – aqui estudamos a consoante “C”; aqui devemos voltar atrás e rever todas as lições e regras já aprendidas, dada a complexidade das aprendizagens subsequentes; agora a criança deve estar preparada para iniciar uma aprendizagem mais complexa, introduzindo-se a primeira consoante com mais do que um valor, mais do que uma leitura – João de Deus chamou a estas letras «consoantes incertas».
- **16ª Lição** – esta lição é dedicada ao “G”; diz-se às crianças que esta letra se lê jjj, e que, por isso, se chama jê; lembramos-lhes que já conhecem uma letra que tem este mesmo som (o “J”; se necessário for, voltamos à 4ª lição); lê-se desta maneira quando tem à frente um “E” (gelo) ou um “I” (girafa).
- **17ª Lição** – esta é uma lição dedicada ao “R”, dizemos à criança: «Esta letra tem dois valores, ou seja, pode ler-se de duas maneiras diferentes; hoje vamos aprender o primeiro valor; esta letra lê-se rrr [R] quando está no princípio da palavra (rato) ou quando estão dois juntos (carro); “Cada elemento do grupo vai repetir este valor – à semelhança do que se fez nas lições anteriores – e dizer quando é que ele se usa. Lêem palavras, fazem frases e falam com o professor/educador sobre as palavras lidas, o que elas representam para cada um, se conheciam a palavra ou não, se sabem o seu significado, etc.
- **18ª Lição** – esta é a lição do “Z”; dizemos que esta letra também tem duas maneiras diferentes e se ler. Lê-se zz... [z] quando está no princípio da palavra (zelo) ou no princípio

de sílaba (azeite); tem o valor de [ʃ] quando está no fim da palavra (rapaz) ou no fim de sílaba (felizmente).

- **19ª Lição** - esta é considerada a lição mais difícil da Cartilha, porque na língua portuguesa. A letra “S” é uma letra com vários casos de leitura e vários valores, consoante a sua posição na palavra; para a criança esta dificuldade deve ser devidamente doseada, transformando a dificuldade em desafio.
- **20ª Lição** - dedicada ao “X”, nesta lição explicitamos que esta letra, ao contrário de todas as outras letras abordadas nas lições anteriores, não tem regras para a sua leitura.
- **21ª Lição** – esta é a lição do til (ã, õ, e, i, u), na qual se aprende a sua função, sabendo que ele só aparece em cima do “A” e do “O”; diz-se que ele (til) serve para nasalar as vogais.
- **22ª Lição** – esta lição é dedicada à letra “M”; dizemos que esta letra se chama metil porque tem duas funções: serve de letra e serve de til; serve de letra quando tem uma vogal à frente (mala) e de til quando não tem vogal à frente, nasalando a vogal que tem atrás (comboio); enquanto letra chama-se mê e lê-se unindo os lábios; no final da palavra, esta consoante apresenta exceções quanto à acentuação.
- **23ª Lição** – esta lição é basicamente igual à anterior; estamos perante uma consoante nasal e, por isso, a letra também serve de til; assim, dizemos «O “N” chama-se netil porque serve de letra “N” quando tem uma vogal à frente (nada), e serve de til quando não tem vogal à frente e nasala a vogal que tem atrás (branco).».
- **24ª Lição** – nesta lição as crianças aprendem a última letra: o “H”; aqui dizemos que esta letra se chama agá e nunca se lê; vale como um sinal etimológico, marca nas palavras a língua de onde vieram, quase sempre do latim. Nas palavras «hera» e «hora» podemos dizer que o “H” pode alterar o significado (semântica) de algumas palavras; o “H” não se lê mas serve para palatalizar três consoantes, nomeadamente o “N”, o “C” e o “L”;

palatalizar quer dizer que as letras palatalizadas mudaram de som e passaram a ler-se com a língua recolhida no véu do palato (céu-da-boca).

- **25ª Lição – Por fim a “lição” do alfabeto;** usando agora a ordem alfabética internacional, fazemos primeiramente uma recapitulação de todas as letras, mnemónicas e regras que foram sendo aprendidas, separadamente, em cada lição.

Para o desenvolvimento deste primeiro protótipo focámo-nos apenas nas duas primeiras lições, desenvolvendo as estruturas e os serviços de base para a implementação do método e construir o modelo de domínio do sistema. Posteriormente, em processos de desenvolvimento criados para o efeito, serão implementadas as restantes lições que integram a estrutura base da cartilha.

4.1.2 O Sistema Android

Este sistema operativo móvel tem como sua base o *kernel* do Linux, este possui uma interface com o utilizado baseado na manipulação direta. Ou seja este sistema foi projetado para sistemas *touchscreen* em dispositivos móveis, como *smartphones* e *tablets*. Os inputs do sistema operativo baseiam-se no toque, como o *swiping*, *tapping*, *pinching*, e *reverse pinching* para manipular os objetos disponibilizado no ecrã e o teclado virtual. Inicialmente este sistema operativo teve o seu grande sucesso em *smartphones*. Já em *tablets* o seu sucesso foi mais lento, pois os programadores não dispensaram grande tempo em desenvolver aplicações de qualidade para esta plataforma. Em 2012, o lançamento do Nexus 7 originou um pico de desenvolvimento de aplicações de melhor qualidade para *tablets*.

No âmbito desta dissertação a plataforma escolhida para o desenvolvimento deste software foi o *tablet*, pois é pretendido com este sistema usar uma ferramenta de apoio ao método da Cartilha Maternal João de Deus e tal como é defendido por este, o uso do livro é fundamental neste método.

Com o uso do *tablet* podemos simular um livro digital. Quanto à escolha do sistema operativo esta recaiu sobre o Android, devido a necessidade de este projeto usar alguns serviços que serão mencionados nos tópicos a seguir e que são fundamentais para a implementação do método. Por esse motivo escolhemos desenvolver nativamente este projeto, pois as *frameworks* existentes, como por exemplo PhoneGap limitam o uso desses serviços em que maior parte das mesmas não permite acesso a esses serviços. Por fim, um dos motivos que levou a escolha deste sistema operativo diz respeito ao número de dispositivos usados neste sistema e o acesso a dispositivos que, com este sistema operativo se torna mais facilitado.

4.1.3 Texto para Voz - *Text-to-Speech*

Este serviço, também intitulado de sintetizador de voz permite ao sistema produzir artificialmente o discurso de um humano. Para isso ele converte o texto em voz simulando o discurso humano. A qualidade deste tipo de sistemas é classificada consoante a similaridade com a voz humana e a capacidade que fornece de entendimento do discurso. Com este serviço pessoas com deficiências visuais ou com dificuldades na leitura de poderem ouvir textos, como por exemplo livros. Fornecendo o acesso de recursos digitais a pessoas com esse tipo de deficiência. Estes sistemas são compostos por duas partes, sendo elas:

- **Front-End.** Este tem duas tarefas principais, onde na primeira ele converte o texto obtido, contendo símbolos, números e abreviaturas para o equivalente de palavras escritas. A este processo processos é intitulado de normalização do texto ou pré-processamento. De seguida, é são atribuídas transcrições fonéticas para cada palavra e divide-se o texto em unidades compostas, como frases e orações. O processo de atribuição de transcrições fonéticas para palavras é intitulado de conversão *text-to-phoneme* ou *grapheme-to-phoneme*.

As transcrições fonéticas e informações compostas em conjunto constituem a representação linguística simbólica que é emitido pelo front-end.

- **Back-End.** Esta é responsável pelo discurso, em procede converter da representação simbólica linguística em voz (som), ao que poderemos chamar de sintetizador de voz.

Na imagem a seguir podemos visualizar uma visão global do processamento do texto realizado pelo Text-To-Speech.

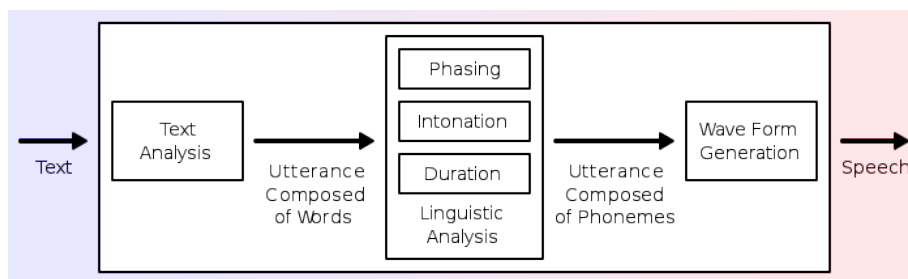


Figura 3 - Processamento de texto realizado pelo Text-To-Speech – figura extraída de (Wikipedia 2014d)

Em que numa primeira fase o texto dá entrada no sistema no seu estado natural, depois da sua entrada é feita uma análise ao mesmo é feita uma decomposição do texto por palavras. Após isso é feita uma análise linguística, onde se verifica as propriedades como a entoação e a duração da palavra, de seguida é decomposta em fonemas e numa última fase de processamento com base nos fonemas criados ele gera o áudio respetivo. Feito este processamento ele reproduz o áudio do texto fornecido inicialmente.

No protótipo desenvolvido nesta dissertação foi usada uma aplicação com o intuito de usar o text-to-speech intitulada de SVOX. A escolha desta recaiu sobre existirem poucas aplicações que possuam a linguagem portuguesa (pt-PT) e devido à qualidade apresentada pelo sintetizador de voz na linguagem referida. Tendo em conta o domínio do protótipo desenvolvido é determinante que a linguagem que o sistema é capaz de produzir seja a linguagem portuguesa.

SVOX

SVOX é um dos fornecedores líderes relativamente a soluções de voz, onde incorpora soluções de *text-to-speech* e reconhecimento de voz em sistemas disponíveis em carros, dispositivos móveis entre outros produtos electrónicos. Agora existe a possibilidade de usar diretamente SVOX *mobile voices*, com soluções *text-to-speech* nas aplicações Android. SVOX *mobile voices* permite por exemplo ler um e-book em voz alta, entre outras possibilidades. Além disso este sistema conta com um portfólio de 25 idiomas, contendo a língua portuguesa (pt-PT) que é determinante para este protótipo.

4.1.4 Reconhecimento de Voz - *Speech Recognition*

Este serviço permite ao sistema receber inputs através do diálogo entre o sistema e o utilizador. O utilizador conversa com o dispositivo electrónico a ser usado e este usa o que foi dito para desencadear alguma ação. Na qual, a ação pode ser apenas reconhecimento de texto, execução de algum comando, entre outras. O objetivo deste serviço é tornar os dispositivos mais amigáveis e interativos e com isso aumentar a produtividade dos mesmos.

Hoje, já existem alguns *smartphones/tablets* que usufruem deste serviço, por exemplo para iniciar uma chamada, para informar da meteorologia, entre outras hipóteses. Como é o caso do SIRI disponibilizado no IOS em que o utilizador pode fazer uma determinada pergunta e o dispositivo irá responder criando assim um diálogo entre homem/máquina. Com a evolução dos *smartphones/tablets* este serviço tem evoluído bastante, o que permite o uso do mesmo em diversas situações que nem sempre são propícias a isso, por exemplo quando se conduz um veículo. Com o uso mais regular deste serviço o seu vocabulário tem vindo a ser aumentado e onde já são capazes de detectar texto completo com um grande nível de precisão e não somente palavras como inicialmente era feito.

Para o desenvolvimento do protótipo realizado nesta dissertação, foi usado o *Speech Recognition* da Google que é disponibilizado pelo sistema operativo Android, pois este já disponibiliza a linguagem portuguesa (pt-PT). E encontra-se disponível em todos os *tablets* que usem o Android.

4.1.5 Gestps - *Gestures*

O *touch screen* trouxe uma nova vida à interação entre o utilizador e o dispositivo tecnológico, onde a sua interação é mais diversificada em comparação com o tradicional teclado. Os utilizadores através do *touch* podem tocar, arrastar, atirar, ou até deslizar elementos visuais, com o intuito de executar determinadas tarefas. No Android é possível fazer de uma forma simples o reconhecimento de inputs através do *touch*, tais como o *swipe*. Porém fazer o reconhecimento de gestos mais complexo tornou-se desafiador e a partir da versão Android 1.6 (Donut) foi incluído o *package* “*android.gesture.**” para alcançar esse fim. Para definir os gestos que serão usados na aplicação a desenvolver, é usado a aplicação *Gesture Builder* que vem incluída no emulador do android. Nessa aplicação é definido o gesto e o seu respetivo identificador. Tendo os respetivos gestos definidos, na aplicação onde foi implementado o serviço com a base de gestos que foi construída. Basta no local respetivo a utilização de gestos, executar o gesto pretendido e a aplicação irá identificar qual o gesto e o seu grau de precisão no processo de deteção. A escolha deste serviço para implementar neste protótipo deve-se ao reconhecimento de caracteres que é necessário para a execução do método.

4.2 Funcionalidades Gerais

Nesta primeira fase de desenvolvimento pretendíamos construir um sistema protótipo capaz de acolher a generalidade das funcionalidades pretendidas. Basicamente, queríamos desenvolver a arquitetura base do sistema e delinear todos os seus serviços fundamentais. Porém, dada a dimensão do projeto, neste trabalho de dissertação focámos a nossa atenção na implementação de

um pequeno conjunto de lições da cartilha, e conforme a lição ia sendo implementada e validada fazer a extensão do sistema à medida. Além disso, o design da aplicação também mereceu um pouco da nossa atenção, apesar do resultado final não se adaptar totalmente à faixa etária ao qual o sistema se destina. Tentou-se, mesmo assim, procurar um design mais generalista, fácil de entender, utilizando sempre o recurso a áudio para o aluno ir ouvindo as palavras e assimilar. Mais tarde o sistema de interface terá que ser estudado e moldado à sua comunidade de utilizadores.

Na Figura 4 podemos ver o ambiente principal da aplicação, no qual estão apresentadas as principais funcionalidades que pretendemos implementar no sistema, com o objetivo de criar um ITS para suporte à Cartilha Maternal João de Deus.



Figura 4 – O menu principal da aplicação.

De seguida, iremos agora abordar cada uma das funcionalidades apresentadas na Figura 4, deixando para mais tarde a apresentação, em concreto, das funcionalidades que foram de facto implementadas. As funcionalidades do sistema são então as seguintes:

- **Cartilha**, com esta opção, é permitido ao utilizador realizar uma leitura da Cartilha Maternal João de Deus, obtida através da Biblioteca Nacional (Deus, 1878). Este sistema contém um PDF com a cartilha original e o utilizador poderá fazer a leitura da mesma quando pretender. Não é necessário recorrer à Internet para descarregar o PDF, indo de encontro ao que era defendido pelo próprio João de Deus, que dizia que o uso do livro para este método é indispensável. Como tal decidimos ter disponível no sistema um exemplar da cartilha em formato digital.
- **Lições**: esta opção é a mais importante desta fase do projeto e onde foi dispensado mais tempo na sua concepção. Nesta opção iremos encontrar as várias lições da Cartilha Maternal já implementadas num meio informatizado e onde se pode escolher e realizar as lições que preferir e as vezes que pretender. Aqui o aluno é levado a escolher a lição que quer fazer ou onde pretende começar e após selecionar a lição começa a aula da respectiva lição em que o aluno faz passo-a-passo, as tarefas da respectiva lição tal como no método da Cartilha Maternal. Ao longo da aula o aluno vai sendo avaliado consoante as resoluções das tarefas apresentadas e receberá uma mensagem por parte do tutor consoante o a correção da mesma.
- **Revisões**: nesta opção o aluno poderá consultar os resultados das várias lições que fez agrupadas por “Teste”. Ou seja, cada vez que o aluno inicia uma lição é começado um novo “Teste” e nele pode encontrar várias lições ou apenas uma. Não tem número de lições definidas nem passos obrigatórios. A qualquer altura o aluno pode consultar os passos da lição que errou ou acertou em determinado “Teste”.
- **Plano**: esta opção permite ao utilizador executar planos pré-definidos. Onde pode ser criado um plano com várias lições a executar, por exemplo pode ser definido vários planos trabalho com a ideia de o aluno treinar determinadas aulas específicas, onde possa ter mais dificuldade. E assim, a qualquer altura o aluno pode escolher o plano que quer fazer e

exercita apenas as lições do plano, além disso pode repeti-lo o número de vezes que pretender.

- **Metas:** nesta opção estão disponibilizados os objetivos de cada lição da cartilha, identificando as metas que se pretende atingir perante o aluno. Assim, pode ser consultado a qualquer momento por parte do utilizador (não tem de ser necessariamente o aluno) de modo a este conhecer mais um pouco da cartilha maternal e os seus objetivos de ensino.
- **Perfil:** nesta opção irá estar definido os dados do aluno, tal como o seu nome, o ano de escolaridade, a turma e a instituição de ensino. Para além disso possui uma avaliação consoante os “testes” realizados, da sua evolução a nível da voz (capacidade do aluno pronunciar o valor das letras), reconhecimento (capacidade do aluno reconhecer as letras através da visualização de imagens) e por fim a escrita (capacidade que o aluno tem de conseguir desenhar a letra).

Estas são as funcionalidades que é pretendido implementar, no entanto como referido em cima as funcionalidades em que nos concentramos mais foi a opção de Lições, devido a esta ser a opção mais importante para esta fase do projeto. Porém foram desenvolvidas as opções de Cartilha e a de Perfil.

4.3 Modelo e arquitetura

Para desenvolver este sistema protótipo utilizamos o sistema operativo Android, além de alguns dos serviços que este fornece, nomeadamente *Voice Recognition* e *Gestures*, tendo sido utilizado este último para representar o modelo do domínio de uma base de dados em *SQLite*, um motor de base de dados suportado pelo sistema operativo Android. Neste modelo de dados, para além do modelo de domínio representado, é também guardada a informação do aluno e de algumas das regras que se encontram no modelo de tutoria, incluídas posteriormente no código do sistema em Java.

4.3.1 O Esquema da Base de Dados de Suporte

Na Figura 5 podemos ver o esquema lógico da base de dados que foi desenvolvido para suportar a operacionalidade do sistema. De seguida, descreveremos de forma breve cada uma das relações integradas neste esquema bem como os atributos que as integram..

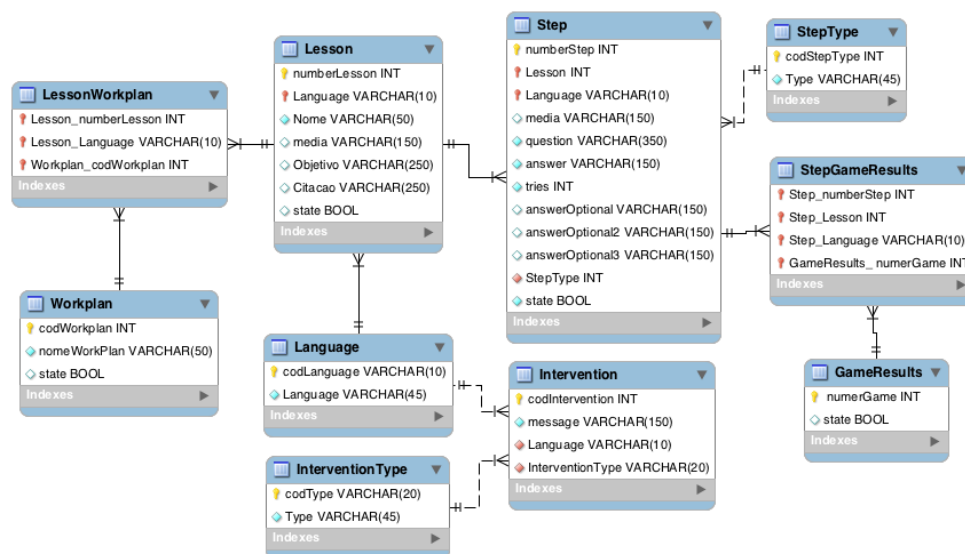


Figura 5 – O esquema lógico da base de dados do sistema.

O esquema da base de dados do sistema é constituído por 10 relações, nomeadamente:

- **Lesson**: esta entidade representa todas as lições que o método de ensino da Cartilha Maternal João de Deus possui. Nesta entidade esta descrita o nome da lição e qual o seu objetivo e uma citação da mesma.
- **Step**: esta entidade representa os passos referentes a cada lição, além disso é definida a questão para cada passo e um conjunto de respostas possíveis.
- **StepType**: nesta entidade é definido o conjunto de tipos que os passos (Step) das lições podem ser, por exemplo se são de escrita de reconhecimento de voz, entre outros.

- **GameResults:** nesta entidade fica registado as várias interações que o aluno tem para executar as lições que escolhem.
- **StepGameResults:** nesta entidade é registado a avaliação de cada passo nas interações que o aluno vai executando ao longo do uso da aplicação.
- **Workplan:** nesta entidade fica registado os planos de trabalho que são criados pelo utilizador.
- **LessonWorkplan:** nesta entidade é registado quais são as lições escolhidas para os planos de trabalho criados.
- **Language:** nesta entidade é registado as linguagens que a aplicação pode suportar.
- **Intervention:** nesta entidade está registado várias mensagens de intervenção por parte do tutor, no processo de avaliação do passo das lições.
- **IntervetionType:** nesta entidade está registado o tipo de intervenções que existem no sistema, como por exemplo se são de erro, estímulo, entre outras.

Entidade	Atributos	Tipo
Lesson	numberLesson	INT
	Language	VARCHAR(10)
	Nome	VARCHAR(50)
	Media	VARCHAR(150)
	Objetivo	VARCHAR(250)
	Citacao	VARCHAR(250)
	State	Boolean
Step	numberStep	INT
	Lesson	INT
	Language	VARCHAR(10)

	Media	VARCHAR(150)
	Question	VARCHAR(350)
	Answer	VARCHAR(150)
	Tries	INT
	answerOptional	VARCHAR(150)
	answerOptional2	VARCHAR(150)
	answerOptional3	VARCHAR(150)
	stepType	INT
	State	BOOLEAN
StepType	codStepType	INT
	Type	VARCHAR(45)
StepGameResults	Step_numberStep	INT
	Step_Lesson	INT
	Step_Language	VARCHAR(10)
	GameResults_numberGame	INT
GameResults	numberGame	INT
	State	BOOLEAN
Language	codLanguage	VARCHAR(10)
	Language	VARCHAR(45)
InterventionType	codType	VARCHAR(20)
	Type	VARCHAR(45)
Intervention	codIntervention	INT
	Message	VARCHAR(150)
	Language	VARCHAR(10)
	InterventionType	VARCHAR(20)

LessonWorkPlan	Lesson_numberLesson	INT
	Lesson_Language	VARCHAR(10)
	Workplan_codWorkplan	INT
Workplan	codWorkplan	INT
	nomeWorkPlan	VARCHAR(50)
	State	BOOLEAN

Tabela 5 – Caracterização dos atributos das relações da base de dados.

4.3.2 A Arquitetura Funcional do Sistema

No sistema que desenvolvemos seguimos a arquitetura clássica dos sistemas ITS, devido às prioridades de implementação das funcionalidades que definimos para o desenvolvimento deste protótipo. O módulo que se encontra mais desenvolvido até ao momento é o Módulo de Domínio. Neste módulo está representado, de forma digital, o método de ensino da Cartilha Maternal de João de Deus. Para armazenar a sua informação utilizámos o *SQLite*.

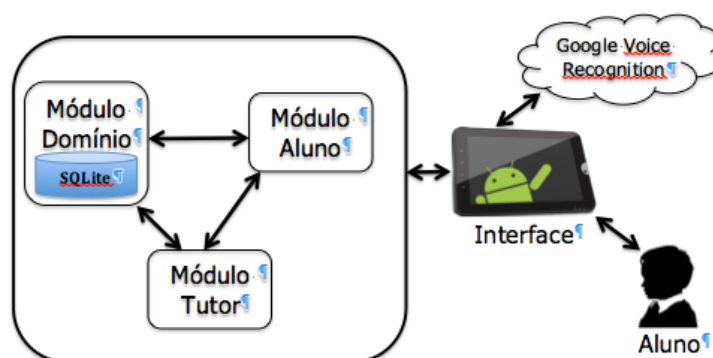


Figura 6 – A Arquitetura funcional do sistema. – figura adaptada de (Giraffa e Rodrigo Rafael Villarreal Goulart 2001)

Na Figura 6 podemos ver um esquema simplificado da arquitetura funcional do sistema desenvolvido, que é constituída pelos seguintes módulos:

- **Módulo de Domínio.** Neste módulo foi implementado o método de ensino da Cartilha Maternal de João de Deus. Através da modelação de dados conseguimos estruturar o método para que fosse possível reproduzi-lo no sistema.
- **Módulo de Aluno.** Este módulo está representado no sistema através de uma avaliação das interações que o aluno vai executando ao longo do uso da aplicação, mais concretamente, através das lições que vai executando. Ou seja, conforme o aluno vai realizando as lições e executa os seus passos, o sistema vai fazendo a sua avaliação de acordo com as resposta dadas e o número de tentativas que teve de usar para resolver a tarefa em causa.
- **Módulo de Tutor.** Este módulo encontra-se implementado de uma forma mais discreta. É o módulo do sistema que contém algumas regras de forma a que seja possível avaliar as respostas do aluno, intervir no processo de resolução das tarefas propostas pela lição e transmitir a aula e os respetivos passos ao aluno.
- **Interface.** Neste módulo está representado o ambiente gráfico da aplicação, que suporta os diversos tipos de interação do utilizador com o sistema. Este módulo comunica com alguns serviços externos, como por exemplo o reconhecimento de voz da Google. Para uma utilização correta do sistema é necessário que este esteja ligado à Internet.

Contudo, é importante referir que nesta arquitetura encontra-se ainda uma primeira fase de construção do protótipo para o sistema ITS que é pretendido criar. Ressalve-se, assim, a circunstância de que arquitetura definida até à data poderá sofrer alterações, consoante a necessidade de implementar novos recursos ou mesmo alterar os que já foram implementados.

4.4 Demonstração de Funcionalidades

4.4.1 A Cartilha de João de Deus

Na funcionalidade da cartilha disponibilizada no sistema o utilizador poderá visualizar uma cópia da Cartilha Maternal João de Deus. Basta para isso clicar em cima do botão “Cartilha”, disponibilizado no menu principal do sistema. Com a seleção dessa opção, o sistema irá visualizar, para consulta, uma cópia da cartilha, devendo-se para isso ter um leitor de ficheiros PDF instalado na plataforma computacional em utilização.

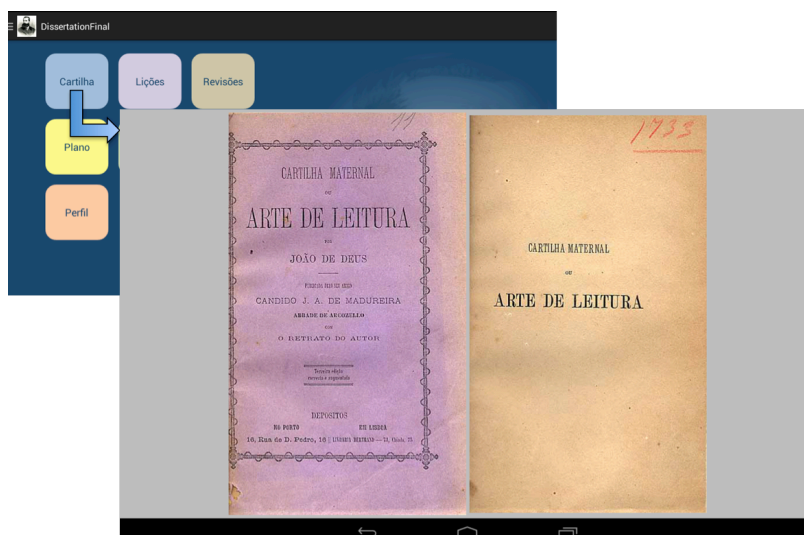


Figura 7 – A Funcionalidade para a visualização da Cartilha Maternal.

Com esta funcionalidade o utilizador poderá ter sempre acesso a Cartilha Maternal de João de Deus, acedendo assim a um “livro digital”, tal como era defendido por João de Deus que, recordemos, afirmava que a utilização do manual escolar é indispensável na utilização do método da Cartilha Maternal.

4.4.2 As Lições da Cartilha

Na funcionalidade de lições o sistema apresenta as lições da cartilha que estão disponíveis, podendo aqui o usuário escolher qual é a lição que pretende executar. Nesta opção as lições são disponibilizadas com uma capa ilustrativa da lição (Figura 8).

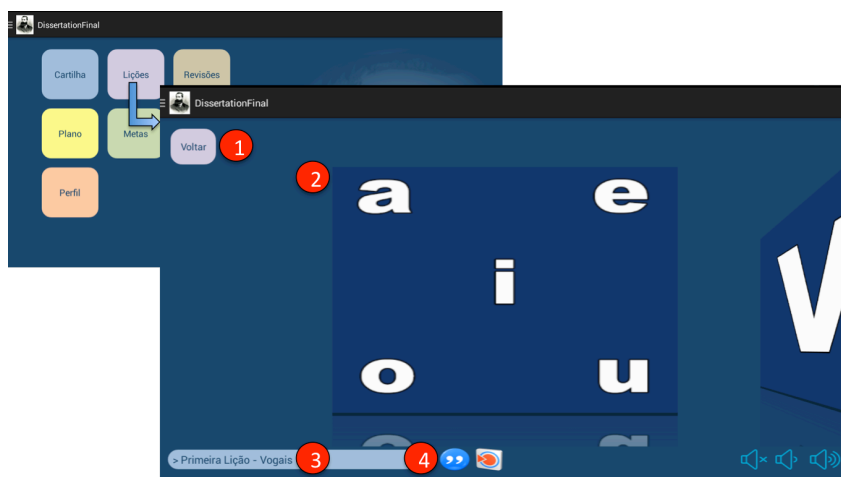


Figura 8 – A funcionalidade das lições da cartilha.

No painel de trabalho do ambiente das lições podemos encontrar as diversas lições já preparadas e disponíveis para estudo, bem como um conjunto de pequenas informações associadas com os tópicos abordados na lição em foco. A informação disponibilizada no ecrã encontra-se estruturada da seguinte maneira:

- 1 - Botão de voltar, o utilizador a qualquer momento pode voltar para o menu principal.
- 2 – Capas que ilustram as lições da cartilha.
- 3 – Nome da lição que se encontra em destaque no ecrã.
- 4 – Botões que fornecem informação relativa a lição em destaque no ecrã, nomeadamente a citação da lição ou o objetivo que se pretende atingir com a lição.

Após selecionar a lição que pretende executar o utilizador é levado para um segundo ambiente onde lhe serão apresentados os diferentes passos referentes à lição que escolheu (Figura 9).

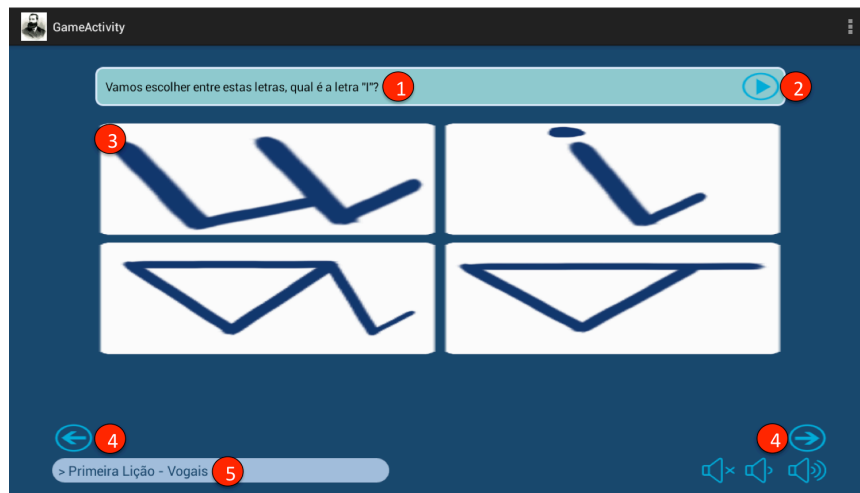


Figura 9 – O layout da execução das lições da cartilha.

Nesse segundo ambiente, o utilizador tem à sua disposição outro conjunto de funcionalidades, de acordo com o tipo de evento que de momento esteja a ocorrer. Em particular:

- 1- Uma pergunta que o tutor executa.
- 2- Um botão para voltar a ouvir a pergunta.
- 3- Um painel no qual são disponibilizadas as opções de resposta, variando este número conforme o tipo de passo.
- 4- Vários botões de navegação; o utilizador pode andar para trás e para frente nos vários passos de cada lição.
- 5- O nome da lição escolhida.

Como foi referido anteriormente, no ponto 3, este painel varia conforme o tipo da lição e do momento de cada lição. De seguida, serão apresentados os vários formatos que este painel pode adquirir.

4.4.3 Reconhecimento de voz

Quando durante uma lição existe um passo em que é necessário o utilizador pronunciar, por exemplo, o valor de uma letra, é necessário o sistema usar as suas funcionalidades de reconhecimento de voz. Nesse caso, o painel central muda de aspecto (Figura 10), mostrando, do lado esquerdo, uma imagem com a letra que o utilizador deve pronunciar, e, do lado direito, um microfone que o utilizador deve seleccionar para que, de seguida, possa dizer o valor da mesma. O *Voice Recognition* reconhece a letra e valida se a resposta esta correta ou não.

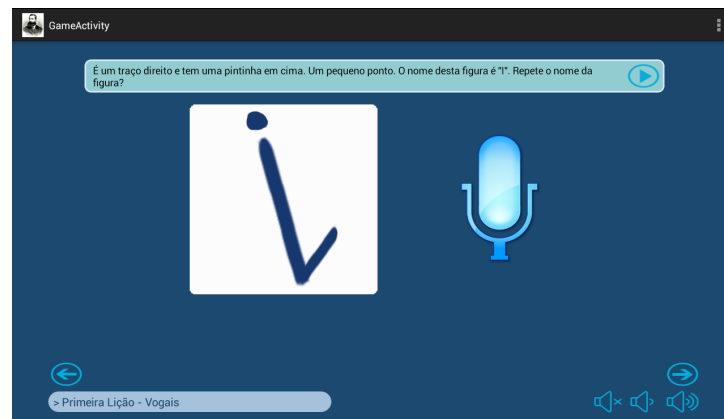


Figura 10 – O *layout* do reconhecimento de voz.

4.4.4 Quizz

O painel central toma o aspecto da Figura 11 quando o utilizador é levado a descobrir qual é imagem que corresponde a letra que o sistema soletra. Nessa altura, é disponibilizado ao utilizador um conjunto de imagens que o utilizador terá que escolher como a opção correta.



Figura 11 – O layout do painel Quiz

4.4.5 O Reconhecimento de Caracteres

No reconhecimento de caracteres o painel central do sistema assume o aspecto da Figura 12. Nesse ambiente é disponibilizado um quadro no qual a imagem de fundo é a letra que o utilizador terá que escrever. A seguir, o utilizador desenha a letra e o sistema procede ao seu reconhecimento, revelando o resultado da escrita e sugerindo uma próxima ação.



Figura 12 – O layout do reconhecimento de caracteres.

4.4.6 Definição de Planos de Trabalhos

Esta funcionalidade permite aos utilizadores criar vários planos de trabalho (Figura 13). Para cada um deles o utilizador escolhe as lições que pretende executar. Desta forma, pode-se definir vários planos de leccionação, nos quais se podem definir as principais aulas que um utilizador poderá realizar nos tópicos em que tenha mais dificuldades.

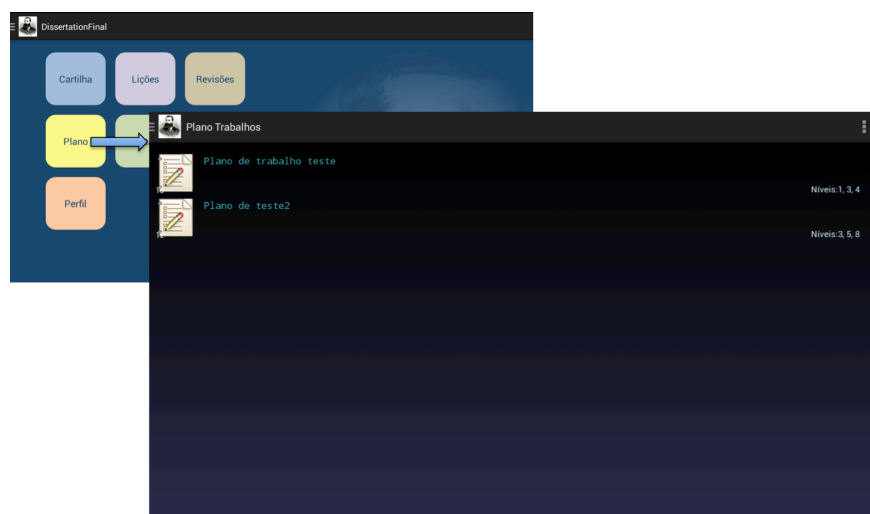


Figura 13 – A funcionalidade Plano de Trabalhos

4.4.7 Caracterização de um Perfil de Utilização

Esta funcionalidade permite ao utilizador visualizar o progresso do seu processo de aprendizagem ao longo das lições que vai executando na no sistema. Aqui, é feita uma avaliação dos passos executados, fazendo o sistema uma análise quantitativa desses resultados conforme o tipo da tarefa em execução, permitindo assim ter uma avaliação de cada um dos atributos envolvidos, tal como podemos ver na Figura 14.

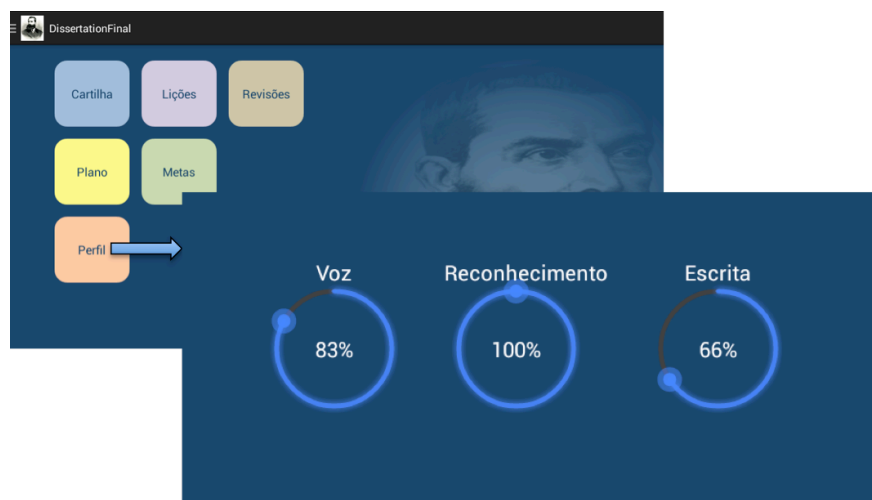


Figura 14 – caracterização de um perfil de utilização

Capítulo 5

Conclusões e Trabalho Futuro

5.1 Resumo e apreciação do trabalho realizado

Ao longo desta dissertação tentámos interligar os benefícios dos ITS, com o ambiente móvel que tem vindo a substituir o computador na sala de aula, dando particular atenção às novas plataformas computacionais como os *tablets*. Esta dissertação apresenta um estudo sobre ITS e sobre o que estes trazem de novo em termos de software educacional. Além disso, também realça as suas vantagens relativamente aos sistemas anteriores, vulgarmente reconhecidos como convencionais, e até que ponto podem simular (ou não) um tutor humano. Ao mesmo tempo que executámos tais tarefas, construímos também um primeiro protótipo para este tipo de sistema, no qual abordámos um dos métodos de ensino da leitura do português disponíveis para crianças: o método da Cartilha Maternal João de Deus.

Hoje em dia, as plataformas móveis como os *tablets* são cada vez mais usadas por utilizadores em todo o mundo, ao nível dos mais diversos perfis profissionais. O acesso a este tipo de plataformas está mais facilitado e tem vindo a ser já usado como um recurso educativo na sala de aula,

substituindo assim o computador. Perante esta realidade a nossa escolha incidiu sobre os *tablets*, sabendo que assim poderíamos prover um qualquer aluno com os meios necessários para aceder aos serviços de um qualquer tutor virtual disponível para instalação nessas plataformas.

No decorrer da dissertação, e após um estudo sobre como aplicar método de ensino escolhido, verificamos que este apresentava aos requisitos bastante próprios, em particular ao nível dos seus interfaces, uma vez que é destinada a um grupo de utilizadores bastante novos. A aplicação do método requeria a implementação de algumas questões menos vulgares, tais como a escrita e reconhecimento de caracteres, o reconhecimento de voz e a capacidade do sistema reproduzir oralmente um texto. Apesar do período de tempo limitado para a execução deste trabalho de dissertação, foi possível contribuir para a execução e desenvolvimento de um protótipo inicial, apresentando a estrutura e serviços base para o sistema de ensino, como também de suportar já duas das lições do método de ensino em questão, incluindo já a parte de reconhecimento de caracteres e processamento de voz.

Com a criação deste sistema pretendemos dar um novo meio para suporte e interação com o método da Cartilha Maternal, que este esteja sempre disponível, em qualquer lugar, além fornecer ao utilizador uma ferramenta para complementar o seu processo educativo, permitindo-lhe saber qual a sua evolução no processo educativo, de uma forma quantitativa e singular. Porém, sistemas como este podem trazer algumas desvantagens relativamente ao ensino, porque estes, apesar de tentarem simular o raciocínio de um discente humano e estarem a caminhar a passos largos para atingir esse objetivo, por vezes possuem diálogos demasiados rígidos, bem como processos de aprendizagem incompletos e superficiais. No entanto, no campo da IA, alguns progressos relativamente a essas limitações têm-se verificado. Além das limitações que estes sistemas possuem, é importante referir que o uso de tecnologia deve ser sempre controlado, porque quando usada em excesso, principalmente nas idades aos quais se destina o nosso sistema, podem levar ao isolamento do aluno ou a situações de cansaço visual, entre outros problemas físicos.

Quanto ao protótipo que desenvolvemos, a sua principal limitação prende-se com o facto de haver a necessidade de este necessitar de uma ligação à Internet para executar alguns dos seus serviços, em especial o reconhecimento de voz, algo que numa futuras versão merece algum tipo de solução alternativa. No entanto, este sistema traz ao aluno uma nova forma de usar e ver o método que aprende na sala de aula e ter um de tutor particular, privado, no qual o aluno pode definir o ritmo do método de aprendizagem, tal como defendido por João de Deus. Procura-se, assim, aumentar a motivação do aluno em aprender e a estimulá-lo na descoberta da língua portuguesa. Para isso o sistema disponibiliza uma interface interativa e atraente para captar a atenção do aluno, contribuindo para uma maior produtividade em termos de todo o processo de ensino levado a cabo. Além disso apresenta as lições de forma criativa, atrativa e integrada, e procura avaliar o aluno de uma forma particular, adaptando assim as estratégias de ensino à sua forma de estar. Em suma, o estudo feito sobre os sistema de ITS e sobre o domínio que é pretendido ensinar com o sistema a desenvolver permitiu que criássemos um sistema protótipo como ponto de partida ao desenvolvimento do sistema ITS final. Na sua atual versão, o sistema responde aos objetivos inicialmente traçados, sendo já possível a um aluno praticar algumas das lições do método da Cartilha Maternal João de Deus e como tal contactar com o seu modelo de ensino e requisitos de aprendizagem.

5.2 Trabalho futuro

Esta dissertação foi iniciada com o objetivo de dar um contributo para a criação de um sistema ITS, que suportasse o método de ensino da Cartilha Maternal João de Deus, permitindo a criação de um tutor virtual inteligente para esse efeito. No decorrer da realização dos trabalhos desta dissertação verificamos que um dos vários problemas a colmatar estava relacionado com a criação de um sistema que operasse em *offline*, o que não foi possível realizar neste trabalho devido aos recursos

necessários pelo método. Assim, é algo que se sugere que seja trabalhado numa próxima evolução do sistema, promovendo um estudo de quais seriam as limitações e as necessidades de trazer o reconhecimento de voz para modo *offline*.

Outro aspeto a considerar numa próxima edição deste trabalho seria estudar a aplicação de uma abordagem orientada por agentes ao sistema. Isto poderia trazer ao aluno um ambiente semelhante ao de uma sala de aula, além do próprio tutor virtual. Com esta abordagem o aluno poderia usufruir de uma maior interatividade com o sistema. Além disso, o sistema ganharia e questões de autonomia e de interação com outras entidades de ensino (outros tutores) inseridas no mesmo ambiente a realizar outros processos de aprendizagem. Além destas vantagens que os agentes podem trazer, hoje temos outras abordagens que incorporam nas suas arquiteturas novos aspetos como crenças, desejos e intenções - BDI (Belief, Desire and Intention). Este segmento da IA tem explorado o modelo de agentes, no qual é descrito o processamento interno do estado do agente utilizando as já categorias mentais como:

- Crenças, que representam o possível conhecimento de um agente.
- Desejos, que representam o conjunto de objetivos a ser atingidos num dado período de tempo por um agente.
- Intenções, que representam as sequências de ações que um dado agente se compromete a fazer para atingir um determinado objetivo.

Com a utilização desta abordagem num ambiente de um ITS, podemos usufruir das suas características, substituindo numa arquitetura clássica de um ITS os seus componentes mais convencionais por agentes, que poderão cooperar, interagir e tomar decisões entre si. Por fim, um outro ponto que poderá ser melhorado nas próximas iterações será renovar a representação do conhecimento do módulo de domínio e a capacidade de raciocínio. Ou seja, realizar um estudo no qual será analisada a melhor forma de representar e atualizar o conhecimento (e.g. redes

neurona, raciocínio baseado em casos, etc.), tendo em conta o ambiente de desenvolvimento em que estamos posicionados. Assim, acreditamos que a médio prazo, o sistema aqui desenhado e iniciado trará um conjunto de inúmeras possibilidades para melhorar o processo de aprendizagem da Cartilha Maternal.

Bibliografia

- Brown, J.S., 1982. Pedagogical, natural language, and knowledge engineering techniques in SOPHIE-I, II and III. In *Intelligent tutoring systems*. pp. 227–282.
- Burton, R.R. & Brown, J.S., 1979. An investigation of computer coaching for informal learning activities. *International Journal of Man-Machine Studies*, 11(1), pp.5–24.
- Carbonell, J.R., 1970. AI in CAI: An artificial-intelligence approach to computer-assisted instruction. *Man-Machine Systems, IEEE Transactions on*, 11(4), pp.190–202.
- Carraher, D., 1990. O que esperamos de um software educacional? *Revista de Educação e Informação*.
- Clancey, W.J., 1984. Methodology for building an intelligent tutoring system. *Methods and tactics in cognitive science*, pp.51–84.
- Collins, A. et al., 1992. Cognition and learning. B. Berliner & R. Calfee, *Handbook of Educational Psychology*, New York: Simon & Shuster MacMillan, pp.1–151. Available at: <http://www.wou.edu/~girodm/611/GCR.pdf>.
- DEUS, J., 1877. A Cartilha Maternal e a Imprensa. , p.15.
- Deus, J. de, 1878. Cartilha maternal ou arte de leitura, Lisboa, 1878 - Biblioteca Nacional Digital. Available at: <http://purl.pt/145> [Accessed October 22, 2014].

-
- Expresso, 2014. O sucesso da velha cartilha maternal - Expresso.pt. *Expresso*. Available at: <http://expresso.sapo.pt/o-sucesso-da-velha-cartilha-maternal=f884235> [Accessed October 21, 2014].
- Fábio Rafael Damasceno, P.A.J., 2010. Sistema Tutor Inteligente PAT2Math: Caráter Pedagógico. *João Pessoa. XXI Simpósio Brasileiro de Informática na Educação . Porto Alegre : SBC, 2010*. Available at: http://www.ccae.ufpb.br/sbie2010/anais//Artigos_Completos_files/76669_1.pdf [Accessed September 22, 2014].
- Fowler, D.G., 1991. A model for designing intelligent tutoring systems. *Journal of medical systems*, 15(1), pp.47–63.
- Freedman, R., Ali, S.S. & McRoy, S., 2000. Links: what is an intelligent tutoring system? *intelligence*, 11(3), pp.15–16. Available at: http://dl.acm.org/ft_gateway.cfm?id=350756&type=html [Accessed October 21, 2014].
- Gamboa, H. & Fred, A., 2002. Designing intelligent tutoring systems: a bayesian approach. *Enterprise Information Systems III. Edited by J. Filipe, B. Sharp, and P. Miranda. Springer Verlag: New York*, pp.146–152.
- Gavidia, J.J.Z. & ANDRADE, L.C.V. de, 2003. Sistemas tutores inteligentes. *Trabalho de conclusão da disciplina Inteligência Artificial do Programa de Pós-Graduação da COPPE-Sistemas da Universidade Federal do Rio de Janeiro. Universidade Federal do Rio de Janeiro*. Available at: <http://tutoresinteligentes.pbworks.com/w/file/fetch/66930838/STImono.pdf>.
- Giraffa, L.M.M., 1999. Uma Arquitetura de tutor utilizando estados mentais. Available at: <http://www.lume.ufrgs.br/handle/10183/17620> [Accessed October 21, 2014].
- Giraffa, L.M.M. & Viccari, R.M., 1999. Estratégias de Ensino em Sistemas Tutores Inteligentes Modelados através da Tecnologia de Agentes. *Revista Brasileira de Informática na Educação*, 5(1), pp.9–18. Available at: <http://ceie-sbc.educacao.ws/pub/index.php/rbie/article/view/2276> [Accessed October 21, 2014].

-
- Giraffa, Rodrigo Rafael Villarreal Goulart, L.M.M., 2001. *ARQUITETURAS DE SISTEMAS TUTORES INTELIGENTES*, Brasil. Available at:
<http://www3.pucrs.br/pucrs/files/uni/poa/facin/pos/relatoriostec/tr011.pdf>.
- KAPLAN, R.; ROCK, D., 1995. New directions for intelligent tutoring. *AI Expert.* , pp.30–40.
- Maite Urretavizcaya, L., 2001. Sistemas Inteligentes en el ámbito de la Educación. , pp.5–12.
Available at: <http://polar.lsi.uned.es/revista/index.php/ia/article/viewFile/292/278> [Accessed October 21, 2014].
- Nkambou, R., Bourdeau, J. & Mizoguchi, R., 2010. *Advances in intelligent tutoring systems*, Springer. Available at: <http://link.springer.com/book/10.1007/978-3-642-14363-2>.
- Nwana, H., 1990. Intelligent tutoring systems: an overview. *Artificial Intelligence Review*, 4(4).
Available at: <http://link.springer.com/10.1007/BF00168958> [Accessed October 21, 2014].
- RAMOS, F.R.C., 1996. Relação do Design com o Marketing e engenharia como estratégia empresarial. Estudos em Design. Anais... P&D Design. *Estudos em Design – Anais do P&D Design*.
- Ruivo, I., 2006. João de Deus: Método de leitura com sentido. Available at:
http://www.casadaleitura.org/portalpha/bo/documentos/ot_metodo_leitura_joao_deus_b.pdf
[Accessed October 21, 2014].
- Santos, N., 1999. ESTADO DA ARTE EM ESPAÇOS VIRTUAIS DE ENSINO E APRENDIZAGEM. *Revista Brasileira de Informática na Educação*, 4(1), pp.75–94. Available at: <http://ceie-sbc.educacao.ws/pub/index.php/rbie/article/view/2295> [Accessed October 23, 2014].
- Skinner, B.F., 1958. Teaching Machines: From the experimental study of learning come devices which arrange optimal conditions for self-instruction. *Science*, 128(3330), pp.969–977. Available at:
<http://www.sciencemag.org/content/128/3330/969.extract> [Accessed October 21, 2014].
- Soloway, E.M. & Johnson, W., 1984. Remembrance of blunders past: a retrospective on the development of PROUST. In *Proceedings of the Sixth Cognitive Science Society Conference*. p. 60.

Valente, P.D.J.A., 1999. *O Computador na Sociedade do Conhecimento*, Campinas: NIED. Available at: <http://www.fe.unb.br/catedraunescoead/areas/menu/publicacoes/livros-de-interesse-na-area-de-tics-na-educacao/o-computador-na-sociedade-do-conhecimento> [Accessed October 21, 2014].

Vicari, R.M., 1989. Tutor inteligente para a programação em lógica : idealização, projecto e desenvolvimento. Available at: <http://www.lume.ufrgs.br/handle/10183/17971> [Accessed October 21, 2014].

Vicari, R.M. & Giraffa, L.M.M., 2003. Fundamentos dos sistemas tutores inteligentes. *Barone, D.; et alii. Sociedades artificiais: a nova fronteira da inteligência nas máquinas. Porto Alegre: Bookman.*

Wenger, E., 1987. Artificial intelligence and tutoring systems.

Wikipedia, 2014a. Microsoft Excel – Wikipédia, a enciclopédia livre. *Wikipédia, a enciclopédia livre.* Available at: http://pt.wikipedia.org/wiki/Microsoft_Excel [Accessed November 19, 2014].

Wikipedia, 2014b. Microsoft PowerPoint – Wikipédia, a enciclopédia livre. *Wikipédia, a enciclopédia livre.* Available at: http://pt.wikipedia.org/wiki/Microsoft_PowerPoint [Accessed November 19, 2014].

Wikipedia, 2014c. Moodle – Wikipédia, a enciclopédia livre. *Wikipédia, a enciclopédia livre.* Available at: <http://pt.wikipedia.org/wiki/Moodle> [Accessed November 20, 2014].

Wikipedia, 2014d. Speech synthesis. Available at: http://en.wikipedia.org/wiki/Speech_synthesis.

