

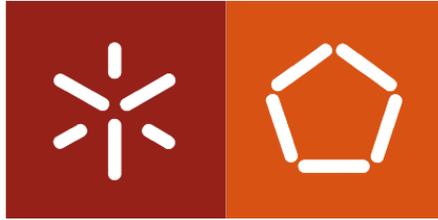
Universidade do Minho

Escola de Engenharia

Mónica Filipa Casanova Martins

**Especificação e Desenvolvimento de um
Módulo de Avaliação para um Sistema de
Ensino Inteligente**

Junho 2016



Universidade do Minho

Escola de Engenharia

Departamento de Informática

Mónica Filipa Casanova Martins

**Especificação e Desenvolvimento de um
Módulo de Avaliação para um Sistema de
Ensino Inteligente**

Dissertação de Mestrado em Engenharia Informática

Trabalho realizado sob orientação do

Professor Doutor Orlando Manuel de Oliveira Belo

Agradecimentos

Em primeiro lugar, gostaria de agradecer aos meus pais por me terem possibilitado seguir a vida Acadêmica e por todo o apoio prestado durante a mesma.

Aos meus irmãos por sempre me impulsionarem nesta tarefa.

Ao meu namorado pela paciência, apoio, companheirismo e dedicação ao longo de todo o processo de desenvolvimento desta dissertação.

Aos restantes amigos e familiares que me apoiaram durante todo o processo de realização desta tese.

Ao meu orientador, Professor Orlando Belo que, pelo acompanhamento e orientação prestados, tornaram possível a realização desta dissertação.

Gostaria também de agradecer à Sr.^a Diretora do Colégio João de Deus de Braga que, pelas sugestões dadas, contribuiu para a melhoria do projeto.

Por último, a todos aqueles que de alguma forma contribuíram direta, ou indiretamente, para a elaboração deste trabalho e que por algum motivo não estão aqui mencionados.

Resumo

Hoje em dia, o recurso às novas tecnologias nas mais diferentes áreas do conhecimento tem sido evidente. A área da educação não foi, pois, exceção. Cada vez mais se tentam arranjar ferramentas tecnológicas que permitam a qualquer aluno ter acesso a um processo de aprendizagem mais simples e eficaz. Dentro desse leque variadíssimo de ferramentas encontramos os ITS (*intelligent tutoring system*). Genericamente, estes sistemas têm como objetivo fornecer instruções a alunos sem a intervenção direta de um professor. Para que isso possa acontecer, com sucesso, é necessário que estes sistemas possuam uma base de conhecimento fiável, com conteúdos adequados em todas as vertentes de ensino que promovem. Nesta dissertação tivemos como base de trabalho a conceção e o desenvolvimento de um módulo de avaliação para um sistema de ensino inteligente, com capacidade para fazer a monitorização de todas as atividades desenvolvidas ao longo de um processo de aprendizagem ou de aferição de conhecimento. Com esta monitorização pretende-se fazer a criação de perfis de aprendizagem para todos os alunos que utilizem o sistema, de forma a que se possa personalizar os processos de ensino, indo em contra às principais necessidades de aprendizagem dos alunos. Nesta dissertação descrevemos o trabalho realizado no desenvolvimento do módulo de avaliação referido, dando particular atenção aos aspetos relacionados com a seleção e implementação dos mecanismos de profiling para a criação automática de perfis de aprendizagem, aos diferentes métodos de avaliação estudados, à preparação dos dados e serviços subjacentes ao seu armazenamento e processamento, à arquitetura do sistema de avaliação desenvolvido e, por fim, à demonstração das funcionalidades implementadas.

Palavras-Chave: Sistemas de Apoio ao Ensino, Sistemas de Ensino Inteligentes, Personalização de Processos de Aprendizagem, Geração e Manutenção de Perfis, Tutores Artificiais, Cartilha Maternal de João de Deus.

Abstract

Nowadays, the usage of technology in the different sectors has been evident. The same applies, with no exception, in education. More and more we try to use IT tools that allow any student to have access to a more simple and relying educational system. It's within this wide range of tools that the ITS (intelligent tutoring system) arises. Generically, these systems have the purpose to provide immediate instructions without the need for a direct teacher's intervention. For a successful and fully functional system, it's necessary that these have a reliable knowledge base, with suitable content in all educational features promoted. In this dissertation, we had as a work base the conception and development of an evaluation module for an intelligent tutoring system, with the ability to monitor all activity carried out during a learning process or the measurement of knowledge. The purpose of this monitoring is to create learning profiles for all students that use the system, so we can personalize the learning mechanisms, adapting to the special learning needs of every student. In this dissertation, we describe the accomplished work in the development of the evaluation module described, giving particular attention to the details related to the selection and implementation of the profiling mechanisms for the automatic creation of learning profiles, to the different evaluation modules studied, to the preparation of the data and its storage and processing, to the evaluation system's Architecture, and finally, to the demonstration of the implemented functionalities.

Keywords: Learning Support Systems, Intelligent Tutoring Systems, Personalization of Learning Processes, Profile Maintenance, Artificial Tutors

Índice

Índice de Figuras	8
Índice de Tabelas.....	10
Lista de Siglas e Acrónimos.....	11
Capítulo 1	12
Introdução	12
1.1. <i>Contextualização</i>	12
1.2. <i>Um Sistema de Ensino Inteligente.....</i>	13
1.3. <i>A Cartilha Maternal João de Deus</i>	16
1.4. <i>Caracterização do Sistema de Ensino Inteligente Desenvolvido</i>	17
1.5. <i>Motivação.....</i>	19
1.6. <i>Objetivos</i>	19
1.7. <i>Estrutura do documento</i>	20
Capítulo 2	22
Modelos de Avaliação.....	22
2.1. <i>Sistemas Genéricos de Ensino</i>	22
2.2. <i>A Organização e Estrutura de um EMS.....</i>	25
2.3. <i>Modelos de Avaliação.....</i>	30
2.3.1. <i>Utilizações da Avaliação de Programas</i>	31
2.3.2. <i>Apresentação dos Modelos CIPP, OBE e Kirkpatrick.....</i>	32
2.3.3. <i>Estrutura do Modelo de Avaliação CIPP</i>	34
2.3.4. <i>Estrutura do Modelo de Avaliação Kirkpatrick.....</i>	35
2.3.5. <i>Estrutura do Modelo de Avaliação Outcome-Based.....</i>	38
2.3.6. <i>Vantagens e Desvantagens</i>	41
Capítulo 3	43
Profiling.....	43
3.1. <i>O Processo de Profiling</i>	44
3.2. <i>Riscos e Problemas</i>	44
3.3. <i>Motor de Inferência.....</i>	45
3.3.1. <i>Tarefas de um Motor de Inferência</i>	45
3.3.2. <i>Abordagem</i>	46
Capítulo 4	54

O Desenvolvimento do ITS	54
4.1. <i>O Modelo da Base de Dados</i>	55
4.2. <i>O Algoritmo de Profiling e O Método de Avaliação</i>	63
4.2.1. Base Preliminar	63
4.2.2. Recolha de Dados.....	63
4.2.3. Preparação dos Dados	65
4.2.4. Mineração dos Dados	66
4.2.5. Interpretação	67
4.3. <i>Demonstração das Funcionalidades do Sistema</i>	72
4.3.1. As Lições	73
4.3.2. Os Testes de Avaliação	76
4.3.3. Avaliação.....	79
Capítulo 5	82
Conclusões e Trabalho Futuro	82
5.1. <i>Considerações, Contributos e Limitações</i>	82
5.2. <i>Trabalho Futuro.....</i>	83
Referências	87

Índice de Figuras

Figura 1: Fatores que afetam a aprendizagem em sistemas e-learning multiculturais....	26
Figura 2: Learning Technology Systems Architecture (LTSA) by IEEE 1484.....	28
Figura 3: Estrutura do Modelo de Avaliação CIPP.....	32
Figura 4: Estrutura do Modelo de Avaliação Kirkpatrick.....	35
Figura 5: Exemplo de uma árvore de decisão.....	46
Figura 6: Exemplo do funcionamento de uma rede neuronal artificial	47
Figura 7: Conjunto de dados após a utilização do algoritmo K-means	49
Figura 8: Cluster baseado em dados com distribuição Gaussiana	50
Figura 9: Exemplo de uma rede Bayesiana com 8 variáveis.....	51
Figura 10: Ciclo de um CBR.....	52
Figura 11: Esquema Lógico da Base de Dados.....	55
Figura 12: Registos da Tabela <i>Difficulty</i>	64
Figura 13: Exemplo de um registo de um teste na tabela <i>TestResults</i>	69
Figura 14: Representação da Tabela <i>TestResultSentences</i>	69
Figura 15: Exemplo de um teste em que o aluno não acertou nenhuma pergunta.....	70
Figura 16: Exemplo de um teste em que o aluno acertou duas perguntas.....	70
Figura 17: Exemplo de um teste em que o aluno acertou todas as perguntas.....	70
Figura 18: Visualização gráfica da nota para o exemplo do "Excelente"	71
Figura 19: Demonstração da imagem apresentada pelo sistema assim que o aluno termina um teste.....	71
Figura 20: Menu das funcionalidades até agora implementadas no sistema.....	73
Figura 21: Menu das lições.....	74
Figura 22: Apresentação inicial da lição da letra "V"	75
Figura 23: Exemplo de uma pergunta de pronúncia, neste caso da letra "V".....	75

Figura 24: Exemplo gráfico de quando um aluno acerta uma pergunta.....	76
Figura 25: Exemplo gráfico de quando um aluno erra uma pergunta.....	76
Figura 26: Botão para iniciar o teste de avaliação.....	77
Figura 27: Representação de um teste gerado pelo sistema.....	78
Figura 28: Representação dos diferentes tipos de avaliação que o aluno pode obter.....	79
Figura 29: Exemplo de uma pergunta que o aluno acertou.....	80
Figura 30: Exemplo de uma pergunta que o aluno errou.....	80
Figura 31: Demonstração da consulta de uma pergunta.....	81

Índice de Tabelas

Tabela 1: Passos Associativos da Estrutura de Cada Modelo de Avaliação.....	31
Tabela 2: Caracterização da tabela <i>Task</i>	56
Tabela 3: Caracterização da tabela <i>TaskType</i>	57
Tabela 4: Caracterização da tabela <i>Step</i>	58
Tabela 5: Caracterização da tabela <i>StepType</i>	58
Tabela 6: Caracterização da tabela <i>StepsHistory</i>	59
Tabela 7: Caracterização da tabela <i>Difficulty</i>	59
Tabela 8: Caracterização da tabela <i>StepTypeHistory</i>	60
Tabela 9: Caracterização da tabela <i>Test</i>	61
Tabela 10: Caracterização da tabela <i>TestResults</i>	61
Tabela 11: Caracterização da tabela <i>TestResultSentences</i>	62
Tabela 12: Caracterização da tabela <i>Grades</i>	63

Lista de Siglas e Acrónimos

BD Base de Dados

DM Data Mining

IA Inteligência Artificial

ITS Intelligent Tutoring System

MD Mineração de Dados

SO Sistema Operativo

SSD Sistema de Suporte à Decisão

IT Informação Tecnológica

TIC Tecnologia de Informação e Comunicação

EMS E-learning Management System

LTSA Learning Technology Systems Architecture

OBE Outcome-Based Evaluation

Capítulo 1

Introdução

1.1. Contextualização

No final do século XX, após inúmeras evoluções e modernizações na indústria, emergiu o primeiro Intelligent Tutoring System (ITS) num projeto que na altura envolvia 6 universidades (Kinshuk, 1996). Uma vez criado, os professores poderiam copiá-lo e modificá-lo para ser utilizado futuramente em outras aplicações. Este sistema revelou-se na altura eficiente e flexível. Contudo acreditava-se que o sistema não fora desenhado sob uma perspetiva educacional e, como tal, não estava desenvolvido de acordo com as necessidades dos alunos e dos professores. Um ITS moderno tenta replicar o papel de um professor, como um professor assistente, com uma alta capacidade de *feedback*.

No domínio dos ITS podemos apontar três projetos de referência que funcionam baseados em diálogos, nomeadamente: *AutoTutor*, *Atlas* e *Why2*. Estes projetos baseavam-se na ideia de que os alunos assimilam melhor conhecimento construindo eles mesmos esse conhecimento. Com base nisso, os sistemas começavam as suas intervenções com perguntas chave para os estudantes e só lhes davam uma resposta em último recurso. Por exemplo, os estudantes do *AutoTutor* focavam-se em responder a questões sobre tecnologia, os estudantes do *Atlas* focavam-se em problemas quantitativos e os do *Why2* na explicação de sistemas físicos quantitativos. Outros sistemas de ensino similares, como o *Andes* (Gertner, Conati, & VanLehn, 1998) davam dicas e *feedback* imediato aos estudantes quando estes hesitavam nas respostas. Os sistemas tentavam e adivinhavam as suas respostas sem nunca

ter um conhecimento profundo dos conceitos. Foram realizadas várias pesquisas com um pequeno grupo de estudantes utilizando os sistemas de ensino do *Atlas* e do *Andes*. Os resultados revelaram que os estudantes que utilizaram o *Atlas* mostravam melhorias significativas, quando comparados com os estudantes do *Andes*. Contudo, tendo em conta que os sistemas anteriormente referidos necessitam da análise do diálogo dos estudantes, ainda há muitas melhorias a fazer para que se consiga dar resposta a diálogos mais complicados. Hoje, a arquitetura mais utilizada é a do *AutoTutor*.

1.2. Um Sistema de Ensino Inteligente

Um sistema de ensino inteligente é um sistema informático capaz de providenciar os meios necessários para apoiar a preparação dos seus utilizadores com conhecimento e perícia na área de especialização para o qual foi preparado. Usualmente, estes sistemas proporcionam uma aprendizagem eficaz através da utilização de inúmeras técnicas de IT, conjugadas com estratégias de ensino estabelecidas por pedagogos experientes. Estes sistemas pretendem resolver o problema da extrema dependência dos alunos em relação aos professores, mantendo níveis elevados na qualidade do ensino que ministram. A estrutura mais comum de um sistema de ensino inteligente incorpora 4 componentes essenciais, nomeadamente (Nwana, 1990) (Freedman, 2000) (Nkambou, *Advances in intelligent tutoring systems.*, 2010):

- O Modelo de Domínio.
- O Modelo do Estudante.
- O Modelo do Tutorial.
- O Modelo da Interface do Utilizador.

O modelo de domínio, também conhecido como modelo cognitivo, integra os meios necessários para a resolução dos diversos problemas, dentro do domínio de aplicação do sistema. Mais especificamente, este modelo contém os conceitos, regras e estratégias de resolução de problemas do domínio a ser aprendido. Este modelo, consegue cumprir com diferentes papéis, nomeadamente como fonte de conhecimento especializado, como modelo base para a deteção de erros e, também, como um modelo capaz de avaliar a performance do aluno (Nkambou, *Advances in intelligent tutoring systems.*, 2010). Por sua vez, o modelo do estudante pode ser visto como uma sobreposição do modelo de domínio. É considerado o núcleo de um ITS que

presta especial atenção às fases cognitivas e afetivas do estudante, bem como à sua evolução à medida que os processos de aprendizagem avançam. Conforme o trabalho que o estudante vai realizando o seu processo de resolução de problemas, o sistema inicia um processo denominado *model tracing*. Sempre que um estudante se desvia do modelo de domínio, o sistema identifica que ocorreu um erro. Relativamente ao modelo do tutorial, este é o responsável por aceitar a informação do modelo de domínio e do modelo do estudante, tomando decisões sobre ações e estratégias de aprendizagem. A qualquer momento, no processo de resolução de problemas o aprendiz pode pedir assistência sobre o que fazer a seguir relativamente à sua localização atual no modelo. O sistema ainda reconhece quando o aprendiz se desviou das regras do modelo, dando nestes casos *feedback* de forma pronta, resultando num período de tempo mais curto para atingir os objetivos. Este modelo poderá conter várias regras de produção que se pode dizer que existem num de dois estados - assimilado ou não assimilado. Sempre que um estudante aplica uma regra com sucesso a um problema, o sistema atualiza a probabilidade estimada que define que o estudante aprendeu a regra. O sistema está continuamente a enviar aos alunos exercícios que requerem a aplicação com sucesso de uma regra até que a probabilidade de que essa regra está assimilada atinja no mínimo os 95%. Por fim, o modelo do interface de um utilizador. Este integra 3 tipos de informação, que são necessários para alimentar um diálogo entre o utilizador e o sistema. Esses tipos são:

1. o conhecimento de padrões de interpretação (para entender um dialogante) e de ação entre diálogos;
2. o conhecimento do domínio para saber o conteúdo da comunicação;
3. o conhecimento necessário para a intenção de comunicar.

No planeamento e desenvolvimento de um ITS existem 8 princípios a ter em conta. Um ITS deverá ser capaz de:

- Representar a competência do aluno como um conjunto de produção.
- Comunicar o objetivo da estrutura subjacente à resolução de problemas.
- Fornecer instruções no contexto da resolução do problema.
- Promover um conhecimento abstrato do conhecimento para a resolução de problemas.

- Diminuir a memória utilizada durante o processamento.
- Fornecer *feedback* em tempo real.
- Ajustar a grande quantidade de instruções consoante o conhecimento adquirido.
- Facultar aproximações sucessivas para a capacidade alvo a desenvolver.

Hoje existem já muitas aplicações de ITS em muitos domínios do conhecimento. Dentro desse leque tão grande e variado, seleccionámos, a título de exemplo, algumas dessas aplicações, nomeadamente:

- ***Mathematics Tutor*** (Beal, Beck, & Woolf, 1998). Esta ferramenta ajuda os estudantes a resolver problemas de frações, de números decimais e de percentagens. O tutor guarda as percentagens de sucesso, enquanto um estudante está a resolver um problema, providenciando problemas subsequentemente apropriados ao nível de conhecimento do estudante. Os problemas subsequentes que são seleccionados posteriormente são baseados na capacidade de resolução em tempo útil por parte do estudante.
- ***eTeacher*** (Schiaffino, Garcia, & Armandi, 2008). O *eTeacher* é um agente inteligente que suporta assistência personalizada online. Constrói o perfil dos estudantes enquanto observa a sua performance em aulas online. O *eTeacher* posteriormente, baseia-se nessa informação para sugerir aulas personalizadas desenhadas para auxiliar o processo de aprendizagem do aluno.
- ***ZOSMAT*** (Keles, Ocak, & Gulcu, 2009) . Este sistema de ensino foi desenhado a pensar nas necessidades de uma sala de aula. Segue e guia um estudante nas diferentes etapas do processo de aprendizagem. É um sistema de ensino inteligente centrado no aluno. De forma a ter sucesso neste processo, o sistema armazena o progresso na aprendizagem e o programa do aluno muda consoante o seu esforço. O *ZOSMAT* pode ser utilizado tanto na aprendizagem individual como num ambiente de uma sala de aula, com a assistência de um tutor humano.

1.3. A Cartilha Maternal João de Deus

O método da Cartilha Maternal João de Deus é a metodologia usada como base no ITS a ser desenvolvido. A Cartilha Maternal é uma obra de natureza pedagógica, escrita pelo poeta e pedagogo João de Deus. Nascido a 8 de Março de 1830, João de Deus foi um eminente poeta lírico, considerado à época o primeiro do seu tempo. O método de ensino da leitura por ele desenvolvido teve grande aceitação popular, sendo ainda utilizado.

A Cartilha Maternal, que tem o subtítulo de Arte de Leitura, foi publicada em 1877 e foi saudada de forma entusiasta por um país no qual o analfabetismo era uma tragédia nacional e o ensino mútuo ainda imperava. O ensino mútuo baseava-se no ensino oral da repetição e memorização, uma vez que o criador deste método acreditava que esta dinâmica inibia a preguiça e aumentava o desejo pela quietude. A Cartilha Maternal surgiu numa altura em que ainda não se tinha conseguido reconciliar o ensino mútuo com uma outra tentativa de alteração metodológica representada pelo Método Português de Castilho (Wikipédia, 2016). Este último, um método que tinha como princípio ser rápido e fácil para instruir a população. É, portanto, um método de ensino primário que Castilho foi buscar ao sistema do francês Lemare, publicado anos antes, e adaptou às particularidades do caso português. Algumas das principais características deste método de leitura repentina eram a acentuação do valor fónico das palavras, o ensino baseado no ritmo, com cantos e palmas, em ligação com uma nova ortografia que, inicialmente, Castilho propôs, baseada no som das palavras. Apesar da persistência de Castilho contra o ensino mútuo e os métodos de repetição, os professores mostraram grande resistência à adoção do seu novo método. Quando, no ano a seguir à morte de Castilho, João de Deus apresentou a sua Cartilha Maternal, os professores já estavam preparados para a aceitar a reestrutura metodológica. Em 1882, por decisão parlamentar, é decretado o uso da Cartilha Maternal nas escolas portuguesas. Este método tornou-se facultativo posteriormente, em 1903. Aquando do lançamento da sua cartilha, João de Deus já se tinha antecipado e advertido que “a primeira condição para ensinar por este método é o estudo da fala”, desenvolvendo uma metodologia por partes sobre o alfabeto, combinando os elementos que se ouvem nas palavras de forma a que, em vez do aluno aprender através da insistente repetição, se familiarize primeiro com as letras.

Uma das características da Cartilha é exatamente o tipo de impressão adotado nas lições, ao apresentar as palavras segmentadas silabicamente, através do recurso ao preto e ao cinzento. Com este recurso, a metodologia de João de Deus não trata as sílabas de forma independente das palavras em que estão inseridas. Para João de Deus, este aspeto permite obter a decomposição das palavras sem quebrar a unidade gráfica e sonora das mesmas. Outra das características da Cartilha é a sua capacidade de elevar um ambiente familiar pleno de amor, alegria, harmonia, serenidade e dedicação mútua. A Cartilha chama-se Maternal porque a paciência e a ternura são atributos essenciais da maternidade. Um último aspeto de realçar na Cartilha é o ritmo de progresso que, deve ser determinado pelo aluno. Há alunos que preferem recapitular as lições mais recentes antes de abordar matéria nova, enquanto que outros preferem avançar sempre, recapitulando apenas quando acham necessário. Há alunos que aprendem rapidamente e outros que precisam de numerosas repetições. Todas as lições parecem elementares para quem já sabe ler, mas, para os mais novos, é muito exigente decorar os símbolos juntamente com os seus múltiplos valores fonéticos. Desde a sua publicação que a Cartilha Maternal se dirige principalmente às crianças dos 5 aos 10 anos. Todavia, há alguns adultos que aprenderam a ler já depois dos quarenta anos de idade com a ajuda da Cartilha Maternal.

1.4. Caracterização do Sistema de Ensino Inteligente Desenvolvido

A introdução de novas tecnologias na sala de aula tem-se revelado benéfica em bastantes aspetos. A verdade é que a utilização destas tecnologias suscita sempre um grande interesse e curiosidade por parte dos mais novos. Neste caso, decidiu-se que o desenvolvimento de um sistema de ensino inteligente, especificamente direcionado para crianças do ensino básico, seria uma forma interessante de pôr em prática alguns dos nossos conhecimentos, tendo sempre em vista as implicações da utilização de um sistema deste género numa instituição de ensino real. Desta forma, ao longo dos últimos três anos, tem-se vindo a desenvolver algum trabalho de conceção e criação de uma plataforma digital com capacidade para suportar o ensino da língua portuguesa, baseado no modelo de ensino proposto na Cartilha Maternal João de Deus (Deus, 2008).

O sistema de ensino que temos em desenvolvimento tem sido direcionado essencialmente para um tipo específico de plataformas - *tablets* que corram o

sistema operativo *Android* – e, por isso, utiliza algumas das funcionalidades nativas neste tipo de sistema, como, por exemplo:

- ***Text-to-Speech***, que inclui as funcionalidades de um sintetizador de voz e tem como finalidade ler o texto escrito no ecrã em voz alta; é ideal para utilizadores que têm impedimentos visuais ou dificuldade de leitura; no nosso caso, pensamos ser a abordagem ideal para quem está a aprender a ler.
- ***Speech Recognition***, orientado para o reconhecimento de voz, este serviço permite ao computador reconhecer o que um utilizador diz, transformando os seus *inputs* em texto.
- ***Gestures***, este serviço, traduzido por gestos, é a base de todo o sistema *Android*, que através do ***touchscreen*** consegue captar as intenções do utilizador; a aplicação *Gesture Builder*, nativa em alguns sistemas como o *Android*, permitirá reconhecer um padrão arbitrário desenhado pelo utilizador no ecrã, e, com isso, permitir que a aplicação execute ações baseadas nesse *input*.

Este ITS tem como objetivo apresentar a matéria de uma forma descontraída e atrativa com o intuito de manter o interesse de uma criança. Para que tal seja possível, o design e as funcionalidades da aplicação são orientados para os mais novos. Para que isto fosse possível, o sistema apresenta os seus conteúdos de uma forma divertida incluindo imagens e sons para uma melhor interação com o aluno. Algumas das funcionalidades básicas integradas no protótipo deste sistema em desenvolvimento são:

- **Cartilha** – na qual é permitido ao utilizador efetuar a consulta ou a leitura da Cartilha Maternal João de Deus em formato eletrónico.
- **Lições** – o formato da Cartilha é composto por 25 lições; aqui podem-se encontrar algumas dessas lições, já adaptadas para formato digital, e escolher qualquer uma delas pela ordem que se pretender; as lições podem ser realizadas tantas vezes quantas se quiser, tal como pretendido por João de Deus; todas as lições são apresentadas de uma forma intuitiva, integrando funcionalidades de reconhecimento de voz, identificação e escrita de caracteres.
- **Teste** – na qual se pode realizar provas de avaliação de conhecimento; usualmente, uma prova inclui um conjunto de cinco perguntas, sendo

três delas as que o aluno tem mais dificuldade, e as restantes duas, perguntas em que o aluno se sente mais confortável; o modo de elaboração das provas foi feito desta maneira para se conseguir manter sempre o aluno motivado; esta funcionalidade permite avaliar o aluno através dos resultados obtidos nas lições e, assim, direcionar o estudo do aluno consoante a avaliação alcançada no teste.

Com o desenvolvimento deste ITS pretende-se criar uma plataforma de aprendizagem na qual o ensino seja orientado a cada aluno e que o ritmo de progresso seja determinado pelo mesmo, tal como defendia João de Deus.

1.5. Motivação

Nos últimos anos, com a evolução das novas tecnologias, o domínio da educação deu um salto importante na implementação de novas maneiras de lecionar e na modernização dos seus métodos de ensino. Com isto, surgem, em alguns casos, a utilização de sistemas computacionais interativos, com algumas técnicas de inteligência artificial embebidas, que os tornam capazes de se tornarem personalizados. Também, o facto de hoje em dia as crianças estarem em contacto cada vez mais com as novas tecnologias faz com que o seu uso deixe as crianças mais motivadas, em comparação com aquilo que usualmente se obtém com a utilização dos ditos métodos de ensino tradicionais. O trabalho desta dissertação está incluído no processo de desenvolvimento de um sistema de suporte ao ensino da leitura do português a crianças no ensino básico, tendo como objetivo principal a conceção e desenvolvimento da sua componente de avaliação do conhecimento adquirido. Como um todo, este sistema permitirá que as crianças aprendam a ler e a escrever Português de uma forma mais personalizada e interativa, disponibilizando aos professores um conjunto de diversas ferramentas inovadoras para auxílio ao desenvolvimento das atividades de ensino referidas.

1.6. Objetivos

Os trabalhos realizados no âmbito desta dissertação tiveram como finalidade a especificação e o desenvolvimento de um módulo de avaliação para um sistema de ensino inteligente. Para isso, foi desenvolvido um motor de regras com a capacidade de tornar o sistema referido capaz de se adaptar às diferentes escolhas do utilizador. A conceção e desenvolvimento de um motor

de regras é uma atividade científica que explora a construção e o estudo de algoritmos que conseguem aprender através de dados. Tais algoritmos funcionam a partir da construção de modelos baseados em *inputs*, utilizando isso como forma de conseguirem fazer previsões ou tomar decisões, ao invés de seguirem apenas instruções programadas. Com o desenvolvimento deste motor pretendia-se generalizar, com base na experiência que iria adquirindo ao longo da sua atividade, a habilidade de se conseguir uma prestação correta numa tarefa nova depois de já ter adquirido experiência na aprendizagem dos dados. Para isso, definimos vários exemplos de treino que foram distribuídos com uma probabilidade desconhecida de forma ao motor ser capaz de construir um modelo geral e fornecer previsões para novos casos.

Atendendo ao público alvo do sistema, crianças com idades compreendidas entre os 5 e os 6 anos, a interface do sistema em questão deveria ser amigável, colorida e com elementos numa escala de grandes dimensões. A informação relacionada com os processos de ensino que o aluno desenvolveu, nomeadamente o conhecimento adquirido ao longo das várias lições, as respostas dadas durante essas lições e o comportamento revelado durante a sua realização será devidamente acolhido pelo sistema numa base específica, permitindo posteriormente caracterizar o aluno, bem como o seu perfil de aprendizagem, e, assim, desenvolver ações de ensino personalizadas orientadas especificamente pelas suas maiores dificuldades e facilidades nos vários tópicos de ensino promovidos pelo sistema.

Quanto a este trabalho de dissertação, em particular, pretende-se incorporar no sistema base que tem vindo a ser desenvolvido um módulo de avaliação, com algumas características ditas inteligentes, que seja capaz de construir um teste de avaliação composto pelas maiores dificuldades e facilidades do aluno. Com a realização destas provas de aferição de conhecimento, espera-se munir o sistema com o conhecimento necessário para orientar o aluno no seu processo de aprendizagem, conseguindo perceber as matérias em que o aluno tem mais dificuldade e, assim, adaptar o processo aos requisitos específicos de aprendizagem deste ou daquele utilizador.

1.7. Estrutura do documento

Para além do presente capítulo, este documento integra mais quatro capítulos, nomeadamente:

- **Capítulo 2** – que apresenta e discute os sistemas genéricos de ensino, abordando a organização e a estrutura de um EMS, bem como demonstra os diversos modelos de avaliação existentes na literatura.
- **Capítulo 3** - Introduz a questão da realização de ações de *profiling* de utilizadores, analisando as suas vantagens e riscos, e apresenta um conjunto de motores de inferência de técnicas de mineração de dados que poderão ser utilizadas na implementação do kernel do módulo de avaliação pretendido.
- **Capítulo 4** – no qual é explicado a técnica de avaliação adotada, expondo-se a estrutura da base de dados de suporte que foi concebida e o processo de extração e transformação de dados que foi necessário realizar. Depois, aborda-se o método de avaliação desenvolvido e descrevem-se as novas funcionalidades de avaliação do sistema e a sua nova arquitetura.
- **Capítulo 5** – o último capítulo; aqui apresenta-se uma breve avaliação crítica do trabalho realizado, salientando-se os seus contributos e limitações, e uma reflexão sobre algumas orientações para desenvolvimento futuro do sistema atual, já com a incorporação do novo módulo de avaliação.

Capítulo 2

Modelos de Avaliação

2.1. Sistemas Genéricos de Ensino

Se uma dada estrutura de ensino pudesse ser passada de estudante para estudante e de disciplina para disciplina, então boas técnicas de aprendizagem e boas maneiras de ensinar poderiam ser compartilhadas. Estruturas de ensino eficazes promovem a envolvimento dos alunos, uma aprendizagem mais produtiva e otimizam a retenção dos alunos. (Scott, 2006)

As estruturas de ensino passam pela criação de experiências de aprendizagem dos alunos. Para isso, examina qual o suporte que é necessário para aprender, como é que os resultados de um processo de aprendizagem podem ser comunicados e como a aprendizagem e o suporte podem ser eficazes, eficientes, atrativos e acessíveis a todos os envolvidos no processo (Koper & Tattersall, 2005).

Quando os métodos de instrução e os seus contextos conseguem ser identificados, este conhecimento pode ser usado para criar novas experiências de aprendizagem, que são elementos bastante importantes para atingir o sucesso do processo de aprendizagem do que decisões *ad hoc* ou aleatórias (Reigeluth, 1999). Isto fez com que a aprendizagem personalizada, emergisse. Este tipo de aprendizagem não se baseia apenas nos resultados dos testes e na sua análise, mas também na capacidade de aprendizagem do aluno. Se um professor conseguir ensinar ao aluno como aprender, ao mesmo tempo que o assiste a adquirir conhecimentos, então o objetivo de aprender e atingir resultados conseguem ser atingidos em simultâneo. Assim, o propósito

do professor não é simplesmente “ensinar”, mas também criar contextos fortes de aprendizagem.

Os padrões curriculares e de ensino mais eficazes são aqueles que induzem o aluno a construir o conhecimento – para investigar áreas intensivamente. O resultado é o aumento da capacidade de o estudante aprender e trabalhar de forma mais inteligente. O truque, claro, é encontrar maneiras de aumentar os níveis de realização e, ao mesmo tempo, ajudar os estudantes a tornarem-se mais competentes, expandindo e articulando o seu repertório de estratégias aprendidas. Este modo de aprendizagem relaciona-se com a habilidade que os estudantes têm de responder com sucesso às tarefas que lhes são apresentadas e àquelas que definem para eles mesmos, em particular:

- fazer a integração de conhecimentos anteriores e novos;
- ser capaz de adquirir e usar uma variedade de competências de aprendizagem;
- conseguir resolver problemas individualmente e em grupo;
- pensar cuidadosamente nos seus sucessos e falhanços;
- avaliar provas conflituosas e ter pensamento crítico;
- aceitar que aprender envolve incertezas e dificuldades;

A implantação de uma tal variedade de estratégias de aprendizagem, comumente denominado por aspetos metacognitivos, pode ser considerada como a capacidade que o aluno tem em assumir o controlo sobre os seus próprios processos de aprendizagem. O ponto chave, seja qual for o contexto de aprendizagem, envolve sempre uma “construção ativa de significado”. Isto implica uma gestão de oportunidades de aprendizagem. Como a aprendizagem é interracional, que ocorre somente quando o aluno faz sentido de experiências particulares em contextos particulares. Este “faz sentido” envolve a ligação ao conhecimento e à experiência anterior do aluno. Assim, a nova aprendizagem tem que ser relacionada e, em última análise, enquadrar-se com aquilo que as pessoas já entendem. A aprendizagem deve, portanto, não só ser vista como um processo, mas também como a uma produção de resultados.

Tendo feito uma breve análise sobre o processo de aprendizagem, precisamos também de prestar atenção às capacidades de aprendizagem que precisam de ser ensinadas e adquiridas. Ao falarmos em aprendizagem personalizada, temos que ter atenção às capacidades que os alunos devem ter de forma a

desenvolver eficácia em termos pessoais. Estas capacidades recaem, essencialmente, sobre três categorias, nomeadamente:

- **Habilidades funcionais** – literacia, numeracia e tecnologia de informação e comunicação (TIC); na maioria dos países estas são consideradas como prioridades chave e, portanto, tendem a ser ensinadas e avaliadas nas disciplinas fulcrais como a da língua materna, matemática e TIC.
- **Habilidades de raciocínio e aprendizagem** – habilidades que os mais novos necessitam de adquirir de forma a se tornarem aprendizes eficazes; ao adquirirem a mestria destas habilidades, os alunos ficam preparados para aumentar o seu sentimento de progresso através do desenvolvimento de capacidades como:
 - melhorar o seu sentido de conquista aplicando uma ampla gama de abordagens de aprendizagem em diferentes disciplinas;
 - aprender como aprender, com a capacidade de monitorizar, avaliar e mudar a forma de como pensam e aprendem;
 - tornando-se aprendizes independentes, saber como construir as suas próprias ideias, adquirir conhecimento e aplicar o conhecimento em diferentes contextos.
- **Habilidades Pessoais** – habilidades que os mais novos precisam de adquirir de forma a desenvolver a sua eficácia pessoal; com o domínio destas habilidades os alunos são capazes de se gerir a si mesmos e de desenvolver relações sociais e de trabalho.

Esta estrutura clarifica os tipos de habilidades que um aluno deve adquirir. Mas, de forma a garantirmos que um aluno realmente possa vir a dominar estas habilidades, deve existir um acordo em como as habilidades devem:

1. ser incorporadas no processo de ensino e aprendizagem, especialmente tendo em conta que na maioria dos sistemas essas habilidades não são ensinadas da forma mais correta em todas as escolas e, assim, o desenvolvimento destas habilidades por parte dos alunos é irregular;
2. ser desenvolvidas coerentemente ao longo do percurso escolar, especialmente quando, na maioria dos sistemas, essas habilidades não são definidas ou desenvolvidas de forma sistemática e a progressão acaba por ser assumida.

Em suma, a clareza fornecida por um quadro de competências único, em conjunto com uma melhor orientação e formação sobre pedagogia irá, por si só, criar uma maior coerência em qualquer percurso escolar. Esta é a base necessária para assegurar que a essência do ensino personalizado está disponível para todos os alunos.

2.2. A Organização e Estrutura de um EMS

E-Learning refere-se à forma em como o conteúdo de aprendizagem é disponibilizado através de meios tecnológicos. Estes sistemas podem ser classificados como síncronos ou assíncronos. Os processos de aprendizagem disponibilizados por meios e-Learning são de tal forma eficazes, que a aprendizagem *online* tornou-se um paradigma promissor na educação. Existem várias características específicas para os sistemas E-Learning como a interação, personalização, adaptação, inteligência, interoperabilidade, acessibilidade e segurança (Sakthiyavathi & Palanivel, 2009).

Os sistemas *online* são, hoje em dia, um canal de comunicação bem estabelecido entre uma ampla variedade de organizações e os seus diversos públicos. A Internet torna acessível uma base de clientes muito diversa, tanto em termos globais como parciais. As oportunidades para aceder aos meios de competitividade global são consideráveis. A Internet Live Stats (Stats, Internet Live Stats, 2016a), estimou que em 2015 os utilizadores online seriam cerca de 3.1 biliões, distribuídos por três regiões principais: a Ásia(1.3 biliões), a América do Norte e do Sul (596 milhões) e a Europa(520 milhões). Hoje em dia, 40% da população mundial tem acesso à internet. Todavia, em 1995 essa taxa de acesso era menor que 1%. De facto, foi uma evolução extraordinária. O número de utilizadores aumentou dez vezes entre 1999 e 2013, sendo o primeiro bilião atingido em 2005, o segundo em 2010 e o terceiro em 2014 (Stats, World Wide Web, 2016b).

Enquanto que as projeções para o futuro variam muito, a Internet oferece um potencial enorme nas inúmeras áreas. No domínio do *E-Learning*, com o aparecimento da informação global, a sociedade fez com que muitas instituições de ensino, tanto públicas como privadas, oferecessem os seus serviços ao mercado global. De acordo com o Centro Europeu de Desenvolvimento de Treino Vocacional (CEDEFOP), o *E-Learning* tem o potencial de mudar radicalmente a educação, abrindo novas maneiras de

aprender e aumentando a possibilidade de as pessoas adquirirem novas habilidades (Cedefop, 2002). Contudo, em termos da percepção por parte do utilizador, existem evidências que ainda muito estudo precisa de ser feito para melhorar a qualidade desses sistemas de ensino. A facilidade de utilização é um dos fatores chave para encorajar os utilizadores a adotarem métodos de *E-Learning*. Melhorar a usabilidade destes sistemas passa por abordar vários aspetos tradicionais ao nível da interação do humano com o computador (Smith, 1997). No entanto, o sucesso internacional do *E-Learning* é afetado por uma ampla variedade de problemas, que se relacionam muitas vezes com diferenças culturais. Na Figura 1 apresenta-se um modelo que revela como as diferenças culturais poderão afetar o sucesso dos sistemas de *E-Learning*.



Figura 1 - Fatores que afetam a aprendizagem em sistemas de e-learning multiculturais - imagem extraída e adaptada de (Ghaoui, 2004).

Na camada mais profunda, nos problemas genéricos de aprendizagem, posiciona-se a forma como as pessoas realmente aprendem, o que pode ser considerado um problema complicado (Hofsted, 1986), uma vez que, por exemplo, identifica diversas áreas que podem influenciar o processo de aprendizagem multicultural, nomeadamente:

1. a posição social dos professores e estudantes;
2. a relevância do programa pedagógico;
3. o perfil das habilidades cognitivas entre a população de onde origina o professor e o aluno;
4. os padrões de comportamento esperados do professor/aluno;

5. a interação entre estudantes.

O nível intermédio da Figura 1 revela a forma como o *design* de sistemas de ensino online vai de encontro às necessidades de aprendizagem dos alunos multiculturais. Em (Haulmark, 2002) abordou-se a maneira de estar dos professores que usem este tipo de sistemas, referindo que estes deverão ter uma maneira diferente de pensar em relação às suas formas de ensinar e de comunicar em situações multiculturais. Nesse trabalho também se aborda as várias mudanças que têm que existir na natureza da interação entre o professor e o aluno, e também entre os alunos.

Os dois pontos anteriores são, pois, importantes para o processo de *e-learning*. Todavia, para os sistemas de ensino *online* existe uma série de problemas que ainda têm de ser abordados. Estes problemas estão relacionados com as interfaces dos sistemas. A usabilidade tem sido definida como estando relacionada com a eficiência, eficácia e satisfação com que os utilizadores conseguem completar as suas tarefas. Sem uma boa interface, amigável, os estudantes não serão capazes de conseguir explorar a um nível mais profundo destes sistemas e, conseqüentemente, obter um processo de ensino apropriado.

Nos últimos anos, os ambientes de aprendizagem são algo que é promovido por muitas agências governamentais, universidades e indústrias, um pouco por toda a Europa, América e Ásia. O grupo de trabalho IEEE “1484 Learning Technology Standards Committee (LTSC)” (Webmaster, Informatik, 2002) desenvolveu uma arquitetura que pode ser aplicada em todos os sistemas de aprendizagem tecnológicos (LTSA). A especificação do LTSA cobre um infindável número de sistemas, que são conhecidos como tecnologia de aprendizagem, treino baseado em computação, instruções assistidas informaticamente, ou tutores inteligentes, entre outras designações. A especificação do LTSA é pedagogicamente neutra, sendo composta por conteúdos neutros e ser culturalmente neutra. Além disso a especificação do LTSA fornece os elementos para perceber os sistemas atuais e futuros, promove a interoperabilidade e a portabilidade identificando interfaces críticas do sistema, e incorpora um horizonte técnico de pelo menos 5-10 anos, ao mesmo tempo que se mantém adaptável a novas tecnologias e a novos sistemas tecnológicos de aprendizagem.

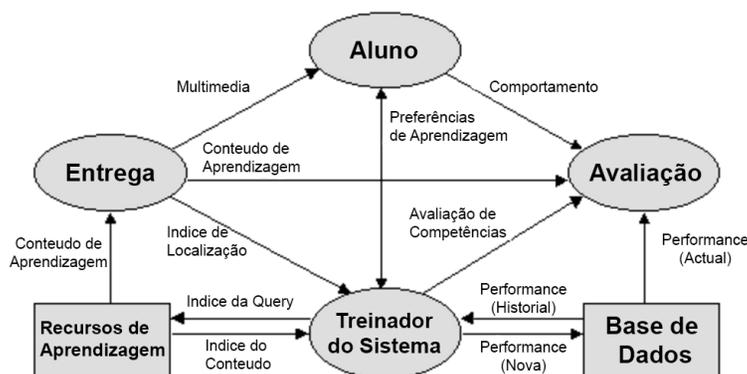


Figura 2 - IEEE 1484 Learning Technology Systems Architecture (LTSA) - imagem extraída e adaptada de (Webmaster, Informatik, 2002).

A especificação LTSA, Figura 2, considera o desenvolvimento de um conjunto de módulos bem definidos, nomeadamente:

- **Aluno** - O processo da entidade “Aluno” é uma abstração de um aluno humano. A entidade “Aluno” pode representar um único aluno, um grupo de alunos que aprendam individualmente, um grupo de alunos que aprendam em conjunto, um grupo de alunos que aprendam em diferentes papéis, etc. A entidade “Aluno” pode receber uma apresentação multimédia e o seu comportamento pode ser observado. Neste nível de abstração, a apresentação multimédia e o comportamento observado são guardados separadamente. Contudo, as implementações de sistemas normalmente combinam estas duas fontes numa só. Quanto às preferências de aprendizagem, estas são negociadas com a entidade “Treinador”.
- **Avaliação** – O processo de Avaliação envolve o processamento do comportamento para produzir a avaliação de competências e a performance (atual). O comportamento da entidade “Aluno” é um input do processo “Avaliação”. Este processo produz informação da avaliação de competências (“onde o aluno está”) e envia essa informação ao “Treinador”. O processo “Avaliação” cria informação sobre a performance que é armazenada na base de dados. O processo “Avaliação” usa o conteúdo de interação para fornecer o contexto ao comportamento da entidade “Aluno” para determinar a avaliação mais apropriada. Por exemplo, é esperado que a entidade “Aluno” selecione uma pergunta de escolha múltipla, a opção “#2”; porém, o processo de “Avaliação” não tem contexto no qual as opções “2” (ou “#2” ou “dois”)

sejam a resposta correta – o conteúdo de aprendizagem fornece o contexto (“a resposta correta é “#2”) para correlacionar o comportamento correto.

- **Base de Dados** – Aqui trata-se do armazenamento e recuperação da informação sobre a performance do passado (historial), presente (avaliação das competências), e futuro(objetivos). O comportamento da entidade “Aluno” é um input do processo de “Avaliação”. Este processo produz informação da avaliação de competências (“onde o aluno está”) e envia essa informação ao processo “Treinador”. O processo de “Avaliação” cria informação sobre a performance que é armazenada na base de dados. A base de dados armazena a informação sobre a performance. Esta informação pode ser fornecida pelo processo de “Avaliação” (ex. notas ou lições) ou pelo processo “Treinador” (ex. certificações). A base de dados contém informação sobre o passado (histórico do aluno), presente (ex. atual avaliação de competências) e futuro (ex. pedagogia, objetivos).
- **Treinador do Sistema** - O “Treinador” negocia as preferências de aprendizagem com a entidade “Aluno”. Um método de ensino, estratégia, etc. é escolhido ou pela entidade “Aluno”, pelo “Treinador”, pelos dois, ou por uma entidade externa. Depois, o processo “Treinador” recebe a atual avaliação de competências do processo de “Avaliação” e a informação sobre a performance da base de dados para ajudar no suporte à decisão de futuras experiências de aprendizagem. Baseado na atual avaliação de competências e na informação sobre a performance (historial), o “Treinador” envia as queries aos recursos de aprendizagem de forma a obter os materiais de aprendizagem mais apropriados. Os recursos de aprendizagem retornam o índice de localização, de acordo com a query enviada. Num passo seguinte, os índices de localização apropriados (ex. o plano de uma lição) são enviados ao processo de “Entrega”.
- **Recursos de Aprendizagem** – estes são constituídos pela base de dados que representa o “conhecimento”, informação e outros recursos utilizados nas experiências de aprendizagem. Os recursos de aprendizagem podem ser representados por apresentações, tutoriais, experiências, lições, etc. Os recursos de aprendizagem também contêm “conhecimento”, apresentações, tutoriais, tutores, ferramentas,

experiências, laboratórios e outros materiais de aprendizagem que sirvam como recurso para a experiência de aprendizagem. Estes recursos são pesquisados por queries, e a informação correspondente é retornada como um catálogo, isto é, um conjunto de conteúdos que são, conceptualmente, meta dados. Os índices de localização são extraídos do catálogo produzido. Os localizadores são usados pelo processo de Entrega como forma de obter o conteúdo de aprendizagem, não sendo especificado quem inicia o processo de transferência.

- **Entrega** - O processo de “Entrega”, recebe o conteúdo de aprendizagem dos recursos de aprendizagem baseado nos índices de localização e transforma o conteúdo de aprendizagem numa apresentação multimédia. O processo de “Entrega” recebe também os índices de localização do treinador e recebe o conteúdo de aprendizagem dos recursos de aprendizagem. Além disso, este processo transforma o conteúdo da Aprendizagem numa apresentação multimédia para a entidade “Aluno”. O mecanismo do processo de entrega pode variar entre apresentações e questões, um ITS, ou conferências de vídeo com um tutor humano, entre outras possibilidades. O processo de “Entrega” pode ser implementado com o processo de “Avaliação” de forma a atingir a combinação necessária para experiências de aprendizagem responsivas e interativas.

2.3. Modelos de Avaliação

O principal objetivo da aplicação será o de obter a capacidade de definir o processo de avaliação de um aluno. Isto traduz-se na capacidade de a aplicação conseguir fazer a avaliação de programas. Fitzpatrick, Sanders, e Worthen (2004) concluíram que avaliações de programas são conduzidas de forma sistemática e objetiva de forma a colecionarem, analisarem e interpretarem informação. Mais especificamente, a avaliação de programas lida com a obtenção e documentação de informação sobre um programa em particular, de forma a permitir uma tomada de decisão válida, tendo em conta um aspeto particular desse mesmo programa (McNamara, 2000). A principal finalidade da avaliação de programas é conseguir uma conclusão definitiva, inteligente, objetiva e válida acerca de objetivos específicos e questões relacionadas com a eficácia geral do programa (Fitzpatrick et al.). Tipicamente, a avaliação de programas é usada para medir o progresso na

obtenção de determinados objetivos, melhorar a implementação de um programa e assegurar que instituições que os programas que estão a financiar sejam eficazes.

2.3.1. Utilizações da Avaliação de Programas

A utilização da avaliação de programas pode ser benéfica numa grande variedade de contextos. Esta pode ajudar no desenvolvimento de um programa concreto sobre o qual se tem conhecimento, sobre as intenções do programa e dos requisitos pessoais, ou pode promover a análise da eficiência de um programa e sua relação preço-qualidade. O design da avaliação de programas está dependente da informação requisitada para atingir os objetivos estabelecidos pelo grupo que procura a avaliação (McNamara, 2000). Consequentemente, uma avaliação focada, que aborda um conjunto completo de objetivos de um grupo variado de pessoas, irá produzir a avaliação mais eficaz em termos de tempo e custo. Os três métodos de avaliação que iremos abordar de seguida são o modelo CIPP, o modelo Outcome-Based Evaluation (OBE) e o modelo Kirkpatrick's. Na Tabela 1 podemos ver os diversos passos associativos dentro da estrutura de cada um desses modelos.

Tabela 1 – Passos associativos da estrutura de cada modelo de avaliação.

CIPP		OBE		Kirkpatrick
Níveis de Avaliação	Tomada de Decisão	Níveis de Avaliação	Métodos	Níveis de estrutura
Contexto	Planeamento	Programa	Formativo	Reação
Input	Estrutura	Eficácia		Aprendizagem
Processo	Implementação	Impacto	Sumativa	Comportamento
Produto	Reciclagem	Política		Resultados

2.3.2. Apresentação dos Modelos CIPP, OBE e Kirkpatrick

O Modelo CIPP

O modelo CIPP, Figura 3, é um modelo de avaliação de programas desenvolvido nos anos 60 por Daniel Stufflebeam e seus colegas. Este modelo foi desenvolvido com o intuito de ajudar nos processos de tomada de decisão. Este modelo considera quem é que vai usar os resultados da avaliação, como os vão usar e em que aspetos do sistema eles serão aplicados. De acordo com este modelo, uma avaliação é definida como “uma investigação sistemática do valor do programa ou outro avaliando” (Stufflebeam, 2003). Stufflebeam desenhou o modelo de avaliação CIPP de forma a abordar quatro diferentes “classes” de tomada de decisão: planeamento, estrutura, implementação, e reciclagem, todos eles se correlacionam. Este método de avaliação tem ganho muita popularidade, ultrapassando o modelo do Kirkpatrick, que de seguida apresentaremos.



Figura 3 - Estrutura do Modelo de Avaliação CIPP – imagem extraída e adaptada de (Wilson, Sahay, & Calhoun, 2014)

O Modelo Kirkpatrick

O modelo de avaliação Kirkpatrick tem sido o modelo mais reconhecido e usado globalmente na avaliação de programas de treino. Este modelo afirma que a eficácia do programa de treino pode ser avaliada por quatro níveis distintos: reação (como os participantes reagem ao treino), aprendizagem (até que ponto os participantes mudam a atitude, aumentam o conhecimento e aumentam as habilidades), comportamento (até que ponto acontece uma

alteração do comportamento), e resultados (resultados finais do treino). Desta forma, muita informação é acumulada ao passar de um processo para o outro. A avaliação torna-se mais difícil de processar em cada nível. Consequentemente, a maioria da avaliação ocorre nos dois primeiros níveis, o que tem sido motivo de preocupação para alguns avaliadores (Arnold, 2003). Este modelo tem sido utilizado desde há muito tempo por ser simples, prático e eficaz, para além de ser flexível e completo. Foi também observado que o poder do modelo recai na sua simplicidade e na sua habilidade de ajudar as pessoas a pensar sobre o treino da avaliação dos critérios (Allinger & Janak, 1989). Existem também algumas críticas a este modelo. Alguns autores veem o modelo ir longe demais (Bates, 2004), a não aprofundar o suficiente (Riotto, 2004), bem como assumindo conceitos falsos (Allinger & Janak, 1989) (Bates, 2004), ou a precisar de ser substituído (Nickols, 2004), entre outras. Apesar destas oposições, muitos avaliadores continuam a utilizar o modelo em inúmeros casos de treino como academias (Boyle & Crosby, 1997), tecnologias da comunicação (Baskin, Barker, & Woods, 2005), simulações educacionais (Schumann, P.H. Anderson, & Lawton, 2001) e em *e-learning* (Kruse, n.d.).

Medidas como números de alunos, dados de recrutamento e retenção de alunos na educação são muito importantes (Boyle & Crosby, 1997). Contudo, Boyle e Crosby, referem que a sobrevivência de alguns programas académicos está mais dependente “de apresentar mais conteúdos relacionados com o programa de uma variedade de fontes” (Boyle & Crosby, 1997). Uma abordagem sistemática para a avaliação na educação, como o modelo do Kirkpatrick, é importante porque vai medir fatores como a satisfação dos alunos, proficiência na aprendizagem e a eficácia geral do programa, parâmetros estes que devem ser considerados (Boyle & Crosby, 1997)

O Modelo de Avaliação *Outcome-Based*

O modelo de avaliação *Outcome-Based* destaca a avaliação que as partes interessadas usualmente pedem, isto é, os resultados das organizações referenciadas que refletem a eficácia e eficiência da organização, englobando os objetivos que os programas devem atingir para as pessoas que recebem os seus serviços. Este tipo de avaliação é frequentemente utilizado na educação e na saúde.

2.3.3. Estrutura do Modelo de Avaliação CIPP

O modelo de avaliação CIPP, representado na Figura 3, desenvolve o seu processo em quatro etapas distintas: avaliação do contexto, avaliação de inputs, avaliação do processo e avaliação do produto. Vejamos, então, de seguida, com um pouco mais de detalhe, cada uma delas.

Avaliação do Contexto

Basicamente, o objetivo da avaliação do contexto no âmbito deste modelo é identificar a informação inicial acerca de como o programa irá funcionar (Fitzpatrick, Sanders, & Worthen, 2004). No entanto, a avaliação do contexto vai além da definição do contexto para incorporar a identificação da audiência e das suas necessidades, bem como a comparação das intenções do programa com o público alvo (Stufflebeam, 2003). Este método de avaliação aplica uma larga variedade de técnicas, como, por exemplo, estudos com as partes interessadas e os participantes do projeto, análises de sistema, revisão dos documentos e dados arquivados, implementação de testes de diagnóstico ou múltiplas entrevistas. O propósito deste tipo de avaliação é, pois, definir os objetivos e etapas da mesma.

Avaliação de Inputs

Para executar este método, o avaliador precisa de estabelecer os recursos disponíveis, incluindo materiais pessoais e acessíveis (Stufflebeam, 2003). O objetivo desta etapa é perceber como é que a avaliação deve ser feita e se a estrutura definida é a mais correta para o caso proposto.

Avaliação do Processo

A avaliação do processo é um meio de implementar e refinar o design do programa e os seus procedimentos. Este procedimento de avaliação aborda informação como o estado da estrutura implementada e se, existirem, os obstáculos no sucesso do programa. Adicionalmente, são instigadas revisões para assegurar que o programa não está destinado a fracassar. O fator chave para implementar este tipo de avaliações e assegurar o seu sucesso é a interação (Stufflebeam, 2003). A interação tem de incluir os avaliadores, as pessoas envolvidas nos processos de tomada de decisão, as partes interessadas e todas as pessoas relacionadas com o resultado do programa (Fitzpatrick, Sanders, & Worthen, 2004).

Avaliação do Produto

A avaliação do produto, que é o último elemento do modelo de avaliação CIPP, refere-se à decisão final associada ao destino do programa (Fitzpatrick, Sanders, & Worthen, 2004). Esta decisão poderá incluir a continuação, término, modificação ou revisão do programa (Stufflebeam, 2003). O resultado é um produto de coleções de descrições e vários julgamentos arquivados sobre os objetivos e méritos do programa. De forma, a que o avaliador chegue a uma conclusão, deve recolher tanto informação quantitativa como qualitativa de todo o pessoal e das partes interessadas que estão envolvidas. Se necessário, a avaliação do produto pode ser dividida em subcategorias como impacto, eficácia, sustentabilidade e transportabilidade, de forma a conseguir informação mais concisa sobre os efeitos a longo prazo do programa.

2.3.4. Estrutura do Modelo de Avaliação Kirkpatrick

Inicialmente, Kirkpatrick ponderou sobre um programa de treino de avaliação que, mais tarde, se desenvolveu num modelo hierárquico de quatro passos (Figura 4). A avaliação das reações dos participantes é a forma mais fácil e a que menos custa neste modelo – cerca de 95% das avaliações dos casos de treino ocorrem neste nível (Arthur, Gage-Little, & Munson, 2002).



Figura 4 - Estrutura do Modelo de Avaliação Kirkpatrick – imagem extraída e adaptada de (Learnnovators, 2013).

Reações

Este nível avalia a forma como os participantes reagem ao programa de treino. Aqui, tenta-se entender as percepções dos participantes, tentando-se obter respostas a questões como:

- Gostaram do programa de treino?
- O material de treino era relevante?
- O método de entrega foi relevante?

Nesta fase, as ferramentas para perceber estas reações são ferramentas de *feedback*, formulários, pesquisas pós-treino, ou questionários, que são fáceis de recolher e normalmente não dão cabo do orçamento definido. De acordo com o Kirkpatrick, este tipo de avaliação deveria estar embutido em todos os programas de treino, logo no primeiro nível, uma vez que oferece formas de perceber como o programa pode ser melhorado. Em segundo lugar, permite construir a base do segundo nível do programa de treino, já que as reações dos participantes servem como indicação se a aprendizagem será possível. Apesar de uma reação positiva não garantir que a aprendizagem será possível, uma reação negativa reduz significativamente as suas hipóteses de sucesso.

Aprendizagem

O nível dois mede o crescimento do conhecimento, antes e depois do programa de treino. Para que isto seja possível, são feitos testes aos participantes antes do treino (pré testes) e depois do treino (pós treino). Nesta fase, a avaliação passa pela análise das reações, capacidades e atitude dos participantes perante o conhecimento adquirido, caso existam, obviamente. O que é importante realçar, é que nesta fase não só se verifica o conhecimento e as capacidades aprendidas, mas também até que ponto os alunos avançaram no novo conhecimento adquirido. As definições da avaliação de aprendizagem mais comuns são avaliações ou testes, realizados antes e depois do treino. A realização de entrevistas e observações dos participantes também não é incomum acontecerem nesta fase.

Transferência

O terceiro nível avalia a mudança que ocorreu no comportamento dos participantes, resultante do programa de treino. Nesta fase, a avaliação foca-se e, alguns aspetos importantes, como saber se:

- Estão as capacidades, conhecimento e atitude adquiridos a ser usados diariamente no treino?
- Os participantes utilizaram o conhecimento e as capacidades relevantes? Houve uma mudança significativa na performance dos participantes aquando de volta aos seus empregos?
- A transferência no comportamento foi retida?
- Conseguiria o participante transferir o conhecimento para outra pessoa com sucesso?

A maioria dos avaliadores vê este passo como o mais fidedigno na avaliação do sucesso de um programa de treino. Contudo, esta fase levanta várias questões como, por exemplo, quando, com que frequência, e como avaliar, uma vez que se sabe que é praticamente impossível prever quando é que os participantes vão mostrar o seu novo conhecimento e comportamento. Daí que, durante o nível três, a observação e entrevistas aos participantes durante um certo período de tempo são obrigatórios, de forma a se poder medir características como a mudança, a relevância ou a sustentabilidade.

Resultados

O quarto e último nível tenta avaliar os resultados relativamente à vida profissional dos participantes. Por exemplo, determinar se as transações de venda melhoraram depois de treinar uma equipa de vendas. Frequentemente designado como a conclusão, o nível quatro de avaliação mede o quão bem-sucedido um programa de treino foi no contexto de que é facilmente compreendido, tanto por executivos como dirigentes, ou seja, a constatação da ocorrência de melhores níveis de produção, melhoria da qualidade, baixos custos, aumento das vendas, mudança do comportamento da equipa, falhanços, desperdícios, crescimento, retenção e aumento dos lucros ou retorno do investimento. De um ponto de vista de negócio, estas são as razões para pôr a equipa num programa de treino. Como os resultados financeiros são difíceis de medir e difíceis de associar diretamente ao treino, é da maior importância identificar e ligar os problemas ao participante desde o início do programa. Desta maneira consegue-se perceber exatamente o que se quer medir desde início. Falhando nesta ligação dificilmente se conseguirá ligar os resultados do programa de treino aos responsáveis. De notar, que a avaliação individual não é difícil, mas torna-se num grande desafio aquando da avaliação de uma organização completa.

2.3.5. Estrutura do Modelo de Avaliação *Outcome-Based*

Num modelo de avaliação *Outcome-Based* normalmente utilizam-se quatro tipos de avaliação. O primeiro é a avaliação do programa. Este é o tipo de avaliação que normalmente utiliza referências de pessoas ou de organizações para determinar se o programa está, ou não, a ir de encontro aos resultados esperados. O tipo seguinte é a avaliação da eficácia, que tem como objetivo reportar até que ponto o programa está a atingir os objetivos. Outro tipo é a avaliação do impacto. Este tipo de avaliação estuda se o programa está ou não a fazer a diferença para as partes interessadas em comparação com os programas alternativos. Finalmente, temos a avaliação da política, que pesquisa sobre a equidade, eficiência ou eficácia da política dos resultados do programa (Schalock, 2001).

Avaliação do Programa

A avaliação do programa esforça-se tipicamente por responder à questão "Que resultados o meu programa produz para os destinatários deste serviço?". Mais uma vez, estes tipos de avaliações são mais frequentemente utilizadas na educação e na saúde. Os resultados do programa traduzem-se normalmente na organização dos resultados e dos valores da performance, e nos resultados individuais da performance e dos valores. Os resultados de organização lidam mais com o público alvo e com a sua competência e satisfação. Os resultados individuais estão mais relacionados com preferências pessoais, qualidades e valores.

A avaliação de um programa geralmente envolve três passos. Primeiro o programa deve estabelecer uma base que é composta pelos resultados que estão a ser usados no momento. Essa base é constituída pelos requisitos da avaliação. O passo seguinte envolve o recolher de informação para determinar qual o resultado pretendido. Isto irá dizer ao avaliador a direção da organização. Por fim, o avaliador tentará explicar como a organização pode orientar os seus serviços de encontro aos resultados desejados.

De forma a que uma avaliação de programa seja sucedida, a organização tem de estar disposta a adotar uma cultura de mudança e ser responsável por desenvolver estratégias e objetivos, bem como desenvolver um sistema de monitorização (Mika, 1996). O processo de avaliação tem mais impacto, se todos os interessados fizerem parte dos processos de tomada de decisão.

Contudo, existem algumas limitações na avaliação do programa (Mika, 1996), uma vez que esta consome demasiado tempo em analisar todos os resultados. Por isso, o tempo passado com cada resultado deve ser bem estipulado desde o início. Outra limitação é a utilidade do método que varia, ou seja, é um método que é tão útil quanto o interesse de todos os envolvidos.

Avaliação da Eficácia

A avaliação da eficácia tem como objetivo responder à pergunta “Estará o meu programa a atingir os seus objetivos?”. As suas utilizações principais são:

- Comparar os objetivos do programa com os alcançados.
- Reportar o desempenho do programa e seus resultados.
- Fornecer *feedback* que possibilite a mudança e melhoria do programa (Schalock, 2001).

Este tipo de avaliação é muito parecido ao da avaliação do programa, exceto no aspeto que esta estabelece uma condição de comparação dos resultados finais.

Para começar esta avaliação, começa-se por definir estratégias e planos de desempenho e, através dos resultados obtidos, especifica, antecipadamente, os resultados. Seguidamente, define-se o propósito juntamente com a condição de comparação com a qual comparar a significância dos resultados – é nesta fase que se especificam as categorias resultantes e as técnicas de análise. Os resultados são depois recolhidos e analisados. As avaliações de eficácia conseguem fornecer resultados fidedignos com a possibilidade de recomendações para a sua melhoria (Mika, 1996).

No entanto, implementar um tipo de avaliação que consegue fornecer a informação necessária para a constante melhoria do processo não é tarefa fácil. Alguns dos desafios que os avaliadores têm são, nomeadamente:

- O desenvolvimento de políticas, objetivos e termos de referência.
- O estabelecimento de uma estrutura de monitorização.
- A localização dos recursos.
- A gestão, análise, interpretação dos dados e apresentação dos resultados.
- O estabelecimento de compromissos cooperativos de longo termo entre as organizações e as partes interessadas.

- As recomendações de mudanças na gestão de estilos dentro da organização.

Avaliação do Impacto

A avaliação do impacto aponta para responder à questão: “O meu programa atingiu os objetivos?”. Este tipo de avaliação foca-se no impacto do programa e determina se esse impacto pode ser atribuído à avaliação realizada. Além disso, também dá *feedback* a todos os envolvidos em relação à melhoria dos planos definidos (Schalock, 2001).

De forma a realizar-se uma avaliação do impacto, é necessário que exista um outro grupo com quem se possa comparar os resultados. Existem seis tipos de design que são normalmente usados neste tipo de avaliações, a saber (Schalock, 2001):

1. Experiência/controlado.
2. Pares idênticos.
3. Um grupo hipotético de comparação.
4. Estatuto longitudinal e comparação.
5. Pré e pós comparação.
6. Pessoa como comparação própria.

Aquando da escolha do design, o avaliador deve saber qual o objetivo dos dados e da avaliação, de forma a conseguir determinar se um programa faz diferença (Mika, 1996). Como um tipo de OBE, os resultados podem não parecer tão apropriados para a contínua melhoria, contudo, se o avaliador souber identificar os fatores que são responsáveis pelas diferenças entre os programas e as variáveis que produzem programas bem-sucedidos, então os resultados podem ser benéficos.

Avaliação do Método

A avaliação do método procura responder à questão: “O método definido funciona?”. A avaliação do método foca-se no desempenho e nos valores (Schalock, 2001). Este tipo de avaliação emprega diversos métodos e tem a capacidade de se conseguir focar em algo individual, no programa ou no sistema como um todo. Para além disso, este método de avaliação inclui todas as técnicas referidas nos métodos anteriores e ainda a análise do custo-benefício. Esta análise identifica se os benefícios de um programa justificam

os seus custos (Priest, 2001). A avaliação do método envolve cinco passos. Primeiro, o avaliador deve descrever os objetivos do método definido, a intenção da análise e o seu conteúdo. Deve também identificar os valores que compõe a política. O foco deve ser na interação entre a política e os grupos do programa. Por fim, o avaliador fornece *feedback* aos participantes chave da avaliação. Para o sucesso desta avaliação, é necessária a participação das partes interessadas, incluindo o consumidor, o praticante, os supervisores, os gerentes e os responsáveis por definir a política (Mika, 1996). Esta avaliação lida com tópicos subjetivos e depende dos avaliadores para interpretar os dados.

Nos últimos anos, tem-se vindo a notar uma mudança nas avaliações aplicadas na educação, verificando-se uma mudança dos processos que se usavam para esta avaliação por resultados. Os responsáveis por definir o método a aplicar estão a olhar para os resultados como notas, taxas de proficiência dos professores, envolvimento dos pais, clima escolar, satisfação dos alunos, capacidades académicas e ajustamentos sociais (Kovalik & Dalton, 1998).

2.3.6. Vantagens e Desvantagens

Nenhum modelo de avaliação consegue ser tudo em todas as situações. O modelo Kirkpatrick aguentou o teste do tempo desde que foi demonstrado em 1959, mas não sem gerar alguma controvérsia. Os críticos argumentam que as reações e a aprendizagem avaliada no nível 1 e 2 perdem o valor, quando querem determinar a eficácia do treino, e fornecem pouca ou nenhuma explicação de como a aprendizagem aconteceu. De facto, algumas pessoas do ramo insistem que os dados não podem ser validados porque há pouca correlação entre aquilo que os participantes sentem e aquilo que realmente aprenderam (Birnbrauer, 1987). Os níveis 3 e 4 são considerados custosos e são frequentemente omitidos do resultado final. No entanto, cada nível pode ser administrado separadamente dada a flexibilidade do modelo Kirkpatrick.

Quanto ao modelo CIPP, este tem vindo a ganhar popularidade nos últimos anos. Este modelo foca-se em melhorar os processos e foi desenhado para ser usado nas fases de planeamento e implementação. É também útil quando se avalia o custo e utilidade em retrospectiva para determinar a eficácia de um

programa. Apesar da intenção do seu autor de fornecer flexibilidade, alguns avaliadores acreditam que é necessária a passagem por todas as fases. Contudo, outros letrados reportam o sucesso do uso de versões modificadas do modelo CIPP para necessidades específicas de informação.

O modelo de avaliação Outcome-based também tem vindo a aumentar a sua popularidade com a mudança dos tempos. Num passado recente, assistiu-se a uma mudança para a utilização deste modelo na área da educação e saúde. Este modelo pode ser muito benéfico ao cliente e é relativamente fácil de adaptar, dependendo do resultado final desejado. Pode também melhorar um programa quase imediatamente, se as partes interessadas estiverem disponíveis para mudar e dispostas a fazerem melhorias no programa.

Capítulo 3

Profiling

Um outro objetivo deste trabalho é criar uma aplicação com a capacidade de se adaptar ao utilizador, ou seja, ter a aptidão para construir um teste de avaliação integrando as maiores dificuldades e facilidades de cada aluno. Em termos técnicos, isto traduz-se por *profiling*. O perfil de um utilizador é a visualização gráfica dos seus dados pessoais, a criação de um conjunto de características que determinem o comportamento de aluno. Um perfil refere-se à representação digital específica da identidade de uma pessoa. O perfil de um utilizador também pode ser considerado a representação informática do modelo do comportamento de um utilizador. Um perfil pode ser usado para armazenar a descrição e as características de uma pessoa. Esta informação pode ser usada por sistemas que tenham, posteriormente, em conta as suas características e preferências pessoais. Estes perfis de utilizadores podem ser encontrados em sistemas operativos, programas informáticos, sistemas de recomendação ou sites dinâmicos.

Profiling, em ciências da computação, refere-se ao processo de construção e aplicação de perfis gerados por dados analisados informaticamente. Isto envolve a utilização de algoritmos ou outras técnicas matemáticas capazes de detetar padrões e correlações em grandes quantidades de dados. Quando estes padrões e correlações são usados para identificar ou representar pessoas, são usualmente designados por perfis. Um dos grandes desafios na informação disponível da sociedade é ter que lidar com o constante aumento do volume de dados disponível. Hoje em dia, com a capacidade de digitalização de qualquer tipo de conteúdo, a quantidade de informação disponível tem se tornado de difícil manipulação e análise. E está constantemente a aumentar. Assim, é extremamente importante conseguir distinguir informação útil da descartável. As técnicas de *profiling* são vistas como tendo a capacidade de conseguirem recolher, analisar e encontrar conhecimento nos padrões estatísticos da informação inserida. Este processo, denominado como *Knowledge Discovery in Databases* (descoberta de

informação nas bases de dados) (Usama Fayyad., 1996) fornece ao *profiler* conjuntos de dados correlacionados que podem ser usados como perfis.

3.1. O Processo de *Profiling*

Um processo técnico de *profiling* pode ser separado em sete etapas diferentes, nomeadamente:

- **Base preliminar** - o processo de *profiling* começa com uma especificação do domínio do problema e a identificação dos objetivos de análise.
- **Recolha de dados** - os dados do alvo são formados filtrando apenas os dados úteis da base de conhecimento existente.
- **Preparação dos dados** - os dados são processados de forma a eliminar a existência de dados redundantes.
- **Mineração dos dados** - os dados são analisados com um dado algoritmo desenvolvido para atingir o objetivo.
- **Interpretação** - os padrões resultantes são avaliados por relevância e validados por especialistas no domínio da aplicação.
- **Aplicação** - Os perfis obtidos são utilizados numa qualquer aplicação.
- **Decisão institucional** - a instituição decide que ações ou políticas se irão aplicar a grupos ou indivíduos cujo os dados se relacionem com o perfil.

As etapas de recolha, preparação e mineração dos dados pertencem todos à fase em que o perfil está em construção.

3.2. Riscos e Problemas

Existem vários riscos e problemas associados à execução de um processo de *profiling*. Um deles é a privacidade. As técnicas de *profiling* tornam possível o acompanhamento duradouro do comportamento e preferências de um indivíduo. Os perfis gerados podem revelar informação pessoal ou privada de uma pessoa sem que ela esteja sequer consciente que essa informação existe. (Hildebrandt & Gutwirth, 2008). As técnicas de *profiling* são, também, por natureza, discriminatórias, uma vez que permitem a segmentação de comportamentos sociais que, em alguns casos, poderão não ser os mais justos.

Outro grande problema é o roubo de identidade, isto acontece quando um determinado perfil é usado por pessoas que não estão autorizadas a usá-lo.

3.3. Motor de Inferência

A parte mais complexa de se realizar na construção de um perfil de um aluno, principalmente no de uma criança de 5 ou 6 anos, é a forma como se consegue obter informação útil a partir de dados em bruto. É nesta parte que entra a avaliação dos dados disponíveis.

Um motor de inferência, é um subcampo da ciência da computação que evoluiu do estudo de reconhecimento de padrões e da teoria da aprendizagem computacional no ramo da inteligência artificial (Hosch, 2016). Em 1959 Arthur Samuel definiu motor de inferência como “um campo de estudo que permite aos computadores a habilidade de aprender sem terem sido previamente programados” (Simon, 2013). Tal habilidade tem como base de funcionamento um modelo baseado em *inputs* e com isso tentam fazer previsões ou tomar decisões, em vez de seguirem instruções previamente programadas. Um motor de regras é também uma das ferramentas mais utilizadas no ramo da inteligência artificial por ter a capacidade de materializar métodos, teorias e vários domínios de aplicações no ramo.

3.3.1. Tarefas de um Motor de Inferência

Dependendo da natureza do "sinal" ou "*feedback*" de aprendizagem disponível no sistema, as tarefas de um motor de inferência podem ser classificadas em três categorias, nomeadamente:

- **Aprendizagem supervisionada**, na qual são apresentados exemplos de inputs e outputs desejados ao computador, tendo como objetivo aprender a regra geral de mapeamento de inputs para outputs.
- **Aprendizagem não supervisionada**, uma tarefa na qual o algoritmo de aprendizagem não recebe qualquer tipo de dica, ficando ao seu encargo criar uma estrutura para o input; a aprendizagem não supervisionada pode ser um objetivo por si só, pelo facto de permitir descobrir padrões escondidos nos dados.
- **Aprendizagem reforçada**, um tipo de aprendizagem na qual o programa interage com um ambiente dinâmico a fim de atingir um determinado objetivo sem que haja um professor a dizer-lhe se ficou perto ou não de o conseguir; um exemplo disto é um indivíduo aprender a jogar um jogo, jogando contra um adversário.

Entre várias categorias de tarefas de motores de inferência, encontramos uma em que se considera o output desejado de um destes motores, nomeadamente em:

- **Classificação**, em que os inputs são divididos em duas ou mais classes, na qual a máquina terá de produzir um modelo que atribui inputs desconhecidos a estas classes; um exemplo disto é o caso do spam, no qual o input é o email e as classes o "spam" e o "não spam".
- **Regressão**, em que os outputs são contínuos, o que faz com que isto seja uma tarefa supervisionada.
- **Segmentação (*Clustering*)**, em que o conjunto de inputs é dividido em grupos; ao contrário da classificação, os grupos não são conhecidos de antemão, o que faz disto uma tarefa não supervisionada.

3.3.2. Abordagem

Existem várias técnicas inteligentes para traçar o perfil de um utilizador. A utilização de cada uma delas vai sempre depender da forma como se quer abordar o problema. De seguida, apresentaremos e discutiremos algumas dessas técnicas.

As Árvores de decisão

Nesta abordagem são utilizadas árvores de decisão como modelo de previsão.

O objetivo é criar um modelo que prevê o valor de uma variável com base em vários inputs. Um exemplo de uma árvore de decisão está apresentado na Figura 5, na qual cada elemento do domínio da classificação é denominado por classe e o nodo pai é etiquetado com características do input. Os arcos que saem deste nodo são por sua vez etiquetados com essa mesma característica. Cada folha da árvore é etiquetada com uma classe ou uma probabilidade de distribuição sobre as classes.

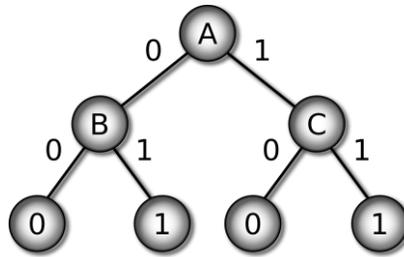


Figura 5: Exemplo de uma árvore de decisão - Imagem extraída de (htt).

Uma árvore de decisão é uma simples representação de um modelo de classificação, sendo uma das técnicas de supervisão mais bem-sucedidas. A forma de aprendizagem de uma árvore passa por subdividir o conjunto inicial de dados em subconjuntos baseados num atributo com um valor de teste. Este processo é repetido para cada subconjunto, de uma forma recursiva, que é denominada por particionamento recursivo. Este processo dá-se por completo quando um subconjunto num determinado nodo tem o mesmo valor que a variável alvo.

Redes Neurais Artificiais

As redes neurais artificiais, Figura6, integram uma família de técnicas de aprendizagem inspiradas nas redes neurais biológicas, sendo usualmente utilizadas para estimar funções que dependam de uma grande quantidade de dados desconhecidos. Uma rede deste tipo é geralmente apresentada como uma rede de "neurónios" interligados que conseguem calcular os valores dos inputs, e são capazes de aprender com o reconhecimento de padrões devido à sua capacidade de adaptação por natureza. A rede aprende a ajustar os seus resultados melhorando os pesos, consoante o que a rede acha que o resultado deve ser e o que realmente é.

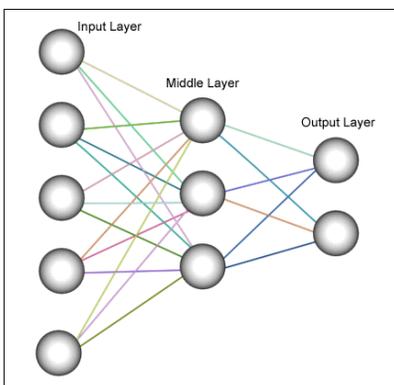


Figura 6: Exemplo do funcionamento de uma rede neuronal artificial – imagem extraída de (Resumekiduit.dvrlists.com, 2016) .

Quanto aos perfis de utilizador, as redes neuronais poderão ser usadas na classificação do utilizador em estereótipos, com base na informação inserida pelo utilizador ao longo da utilização da aplicação. Através da aplicação de estereótipos, a rede vai fazendo suposições do perfil do utilizador até que lhe provem o contrário.

Clustering

Clustering é a ação de agrupar um conjunto de objetos de uma maneira em que os objetos do mesmo grupo (*cluster*) são mais similares entre si do que outros noutros grupos (*clusters*). O *Clustering* não é por si só uma técnica específica, mas sim a tarefa geral a ser resolvida. Esta consegue ser resolvida por vários algoritmos que diferem significativamente da noção do que constitui um *cluster* e como o encontrar eficientemente. A técnica de *clustering* a ser usada depende de cada conjunto de dados e do que se pretende dos resultados. Durante o processo de *clustering* é necessário mudar o conjunto de dados e os parâmetros do modelo até que o resultado atinja as propriedades desejadas. É, por isso, um processo de tentativa erro. De seguida, apresenta-se uma lista com uma seleção dos algoritmos mais relevantes em termos de *clustering* (existem mais de 100 algoritmos distintos publicados):

- *Clustering* Baseado em Conectividade. Este tipo de clustering também é conhecido como *clustering* hierárquico. O *clustering* baseado em conectividade, tem como ideia base que os objetos que estão mais próximos uns dos outros estão mais relacionados entre si do que aqueles que estão mais longe. Estes algoritmos inter-

relacionam objetos de forma a criar *clusters* baseado nas suas distâncias. Um *cluster* pode ser descrito pela distância máxima necessária para relacionar partes de um *cluster*. Estas técnicas não fornecem um único particionamento do conjunto de dados, pelo contrário, fornecem uma extensa hierarquia de *clusters*, na qual os *clusters* se relacionam uns com os outros a certas distâncias.

- Clustering Baseado num Ponto Central. Neste tipo de *clustering*, os *clusters* são representados por um vetor central que poderá não pertencer necessariamente ao conjunto de dados. Quando o número de *clusters* é organizado para "*k*", *k-means clustering* (técnica), representado na Figura 7, dá uma definição formal como uma otimização do problema: encontra o "*k*" no *cluster* central e atribui os objetos ao centro do *cluster* mais perto. A maioria dos algoritmos *k-means* requerem que se defina *a priori* o número de *clusters* - "*k*" - o que é considerado uma das grandes desvantagens destes algoritmos.
- Clustering baseado em distribuição. Os modelos de *clustering* mais relacionados com métodos estatísticos são baseados em modelos de distribuição. Os *clusters* poderão então ser facilmente definidos como objetos que pertencem mais facilmente à mesma distribuição. Uma propriedade conveniente nesta abordagem é que se assemelha à forma como os conjuntos de dados artificiais são gerados: tirando amostras de objetos aleatórios de uma distribuição. Apesar da fundamentação teórica destes métodos ser excelente, eles sofrem de um problema fundamental denominado por sobre ajuste. Este problema acontece quando o modelo estatístico se ajusta demasiado ao conjunto de dados. Um dos métodos mais relevantes é conhecido como mistura de modelos *Gaussiana*. Aqui, os dados são modelados com um número definido de distribuições *Gaussianas* para evitar situações de sobre ajuste. Estas distribuições são inicializadas aleatoriamente e os seus parâmetros otimizados para se adaptarem melhor ao conjunto de dados. Isto permitirá convergir para uma solução ótima. Como as distribuições *Gaussianas* são inicializadas aleatoriamente, poderemos obter resultados diferentes a cada tentativa.

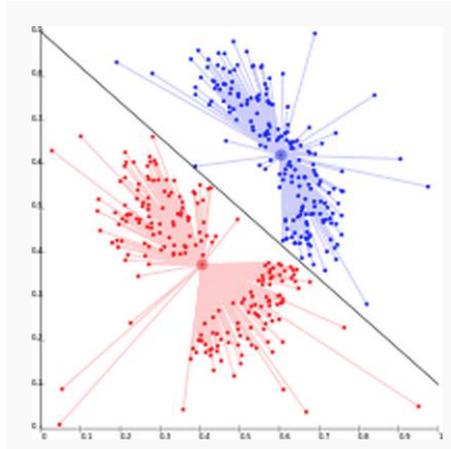


Figura 7: Conjunto de dados após a utilização do algoritmo *K-means* – imagem extraída de (Wikipedia, 2016).

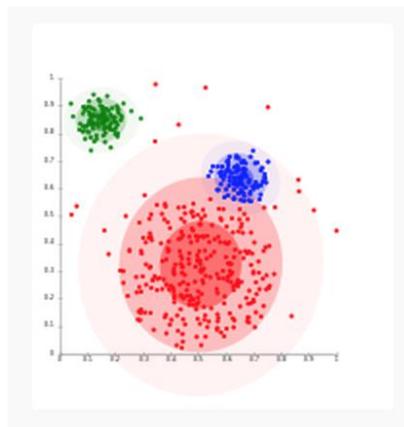


Figura 8: *Cluster* baseado em dados com distribuição *Gaussiana* – imagem extraída de (Wikipedia, 2016)

Redes Bayesianas

Uma rede *Bayesiana*, Figura 9, é um modelo gráfico probabilístico que representa um conjunto de variáveis aleatórias e as suas dependências condicionais. Um exemplo disto seria a representação de uma relação probabilística entre doenças e sintomas que, dados os sintomas, poderia ser representada por uma rede para calcular as probabilidades da presença de outras doenças. Estas redes têm a capacidade de organizar o conhecimento numa dada área através de um mapeamento causa-efeito. Os sistemas que utilizam estas redes conseguem tomar decisões mesmo quando algumas

partes da informação são inexistentes. A sua representação é feita utilizando grafos acíclicos direcionados, nos quais os nodos representam variáveis aleatórias no sentido *Bayesiano*, isto é, podem ser quantidades observáveis, variáveis de latência, parâmetros desconhecidos ou hipóteses. Por sua vez, as arestas representam dependências condicionais, e os nodos que não estão conectados representam variáveis que estão condicionalmente independentes umas das outras. Cada nodo está associado com uma função de probabilidade que recebe como input o conjunto de valores para as variáveis do nodo pai, dando como output a probabilidade da variável representada pelo nodo.

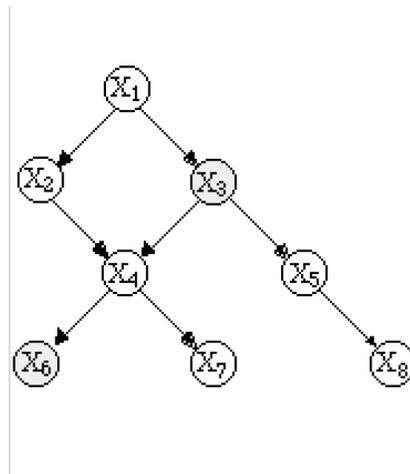


Figura 9: Exemplo de uma rede *Bayesiana* com 8 variáveis - imagem extraída de (htt1).

Numa aplicação como a deste trabalho, uma rede *Bayesiana* permitiria classificar os utilizadores, agrupando-os em diferentes categorias com base nas características principais que os distinguem. Ao contrário das outras técnicas já analisadas, que são muito detalhadas nos critérios utilizados, uma rede *Bayesiana* baseia-se na generalização para classificar os objetos que, num caso como o nosso, seriam os utilizadores.

Case-Based Reasoning

Case-based reasoning (CBR), é um paradigma de resolução de problemas que em muitos aspetos é totalmente diferente de outras abordagens no domínio da inteligência artificial. Ao invés de se apoiar apenas no conhecimento geral de um problema, ou de fazer associações ao longo de relações generalizadas entre descritores de problemas e conclusões, esta técnica é capaz de utilizar conhecimento específico de experiências anteriores, de casos anteriores. A

resolução de um novo problema passa por encontrar, e reutilizar, similaridades em casos anteriores. Outra grande diferença, é a abordagem que esta técnica faz ao problema, partindo de uma perspectiva de aproximação incremental, com conhecimento sustentado, já que uma nova experiência é retida sempre que um problema é resolvido. Isto faz com que esteja de imediato disponível para problemas futuros. Em termos gerais, esta técnica atua num ciclo de 4 fases (Figura 2). Essas fases são:

- **Recuperação**, em que, dado um novo problema, a técnica recupera casos que lhe pareçam relevantes para a solução; Um caso é consistido pelo problema inicial, pela solução alcançada e, tipicamente, pelas anotações de como se chegou a essa solução.
- **Reutilização**, em que se mapeia a solução desde um caso anterior até ao novo problema; isto poderá envolver a adaptação da solução para ir de encontro ao novo problema.
- **Revisão**, tendo-se mapeado a solução anterior para o novo problema, aqui realizam-se testes, nos quais se vê se a solução se adapta ao "mundo real" e, se necessário, faz-se uma nova revisão.
- **Retenção**, após a solução ter sido adaptada com sucesso ao novo problema, nesta fase guarda-se a nova solução como um caso na memória do sistema.



Figura 10: Fases de atuação de uma técnica de CBR – imagem extraída e adaptada de (Cbrwiki, 2012).

Os trabalhos de investigação sobre esta técnica aumentaram exponencialmente nos últimos anos. Como prova disso temos o constante

crescimento de conferências, trabalhos e aplicações usadas no dia a dia no domínio esta técnica.

No nosso trabalho, a aplicação desta técnica poderia ser interessante, na medida em que começaria por gerar um novo caso consoante a informação inicial do utilizador. Este novo caso seria utilizado para recuperar casos de memórias anteriores na base de casos disponíveis. O caso recuperado seria então combinado com o novo caso (reutilização), o que resultaria numa possível solução para o problema inicial. Posteriormente, o caso entraria na fase de revisão, na qual seria testado para se saber se seria bem-sucedido. EM caso contrário, seria novamente revisto. Após a revisão, o novo caso é guardado na base de casos, preparando melhor o sistema para uma situação futura.

Esta técnica apresenta claras vantagens em relação às restantes técnicas que podem ser utilizadas em profiling, nomeadamente (Kolodner, 1992):

- permite ao sistema propor soluções de uma forma rápida;
- permite ao sistema propor soluções em domínios que não estão completamente entendidos pelo mesmo;
- fornece ao sistema recursos que o permitem avaliar uma solução quando não existe nenhum algoritmo disponível para avaliação;
- os casos são úteis quando se está a interpretar conceitos em aberto ou, ainda, mal definidos;
- o facto de ter uma base de casos com experiências anteriores, faz com que o sistema não cometa constantemente os mesmos erros.

Capítulo 4

O Desenvolvimento do ITS

Basicamente, o ITS que temos em desenvolvimento é um assistente pessoal para suportar o ensino da língua portuguesa a crianças entre os 5 e os 6 anos, segundo a metodologia da Cartilha Maternal João de Deus (Deus, 2008). Neste sistema o “livro” passa para o mundo digital, no qual se privilegia a interação do sistema com o aluno. Assim, o aluno é avaliado durante a sua aprendizagem, estudo, utilização da aplicação, e avaliação, e, depois, consoante os resultados obtidos, o sistema vai-se adaptando ao perfil do utilizador. De forma a personalizarmos o assistente pessoal de acordo com o perfil dos seus utilizadores, de forma a que o aluno possa aprender ao seu ritmo e não ao ritmo da turma na qual está inserido. A obtenção do perfil do utilizador é, assim, fulcral neste trabalho, traduzindo-se, também, em inúmeras vantagens no processo de ensino, uma vez que o tornam mais simples, intuitivo, atrativo e personalizado de acordo com as necessidades de cada aluno.

Nesta dissertação já apresentámos e avaliámos as estratégias mais comuns para abordar este tipo de problemas. A avaliação é uma fase muito importante, uma vez que permite a consolidação do estudo dos métodos. Neste capítulo descreve-se com detalhe a aplicação e a avaliação experimental da técnica escolhida e que foi implementada no sistema. Para testar a aplicação, utilizámos um *tablet Nexus 9*, dual-core com 2.3GHz, 32Gb de memória, ecrã de 8.9 polegadas com uma resolução de 2048x 1536 e com o sistema operativo *Android 6.0.1 Marshmallow*.

4.1. O Modelo da Base de Dados

O sistema de ensino está a ser desenvolvido para o sistema *Android*. O sistema de gestão de bases de dados que foi escolhido para manter a base de dados de suporte foi o *SQLite*. O modelo de dados definido para suporte de integração dos dados provenientes da atividade do aluno é constituído, essencialmente, por 3 tabelas principais, nomeadamente: “Difficulty”, “StepTypeHistory” e “StepsHistory”. Por sua vez, o modelo de dados definido para suporte à integração dos dados relativos à avaliação do aluno é constituído por outras 4 tabelas, de referir: “Test”, “TestResultSentences”, “TestResults” e “Grades”. Na Figura 11 podemos ver o modelo lógico que foi construído para o sistema de ensino, que inclui outras tabelas do modelo de dados global e que foram incluídas em outras fases de desenvolvimento do próprio sistema. Estas tabelas – “Step”, “StepType”, “Task” e “TaskType” - atuam como tabelas auxiliares para suporte a todo o processo de integração e transformação dos dados.

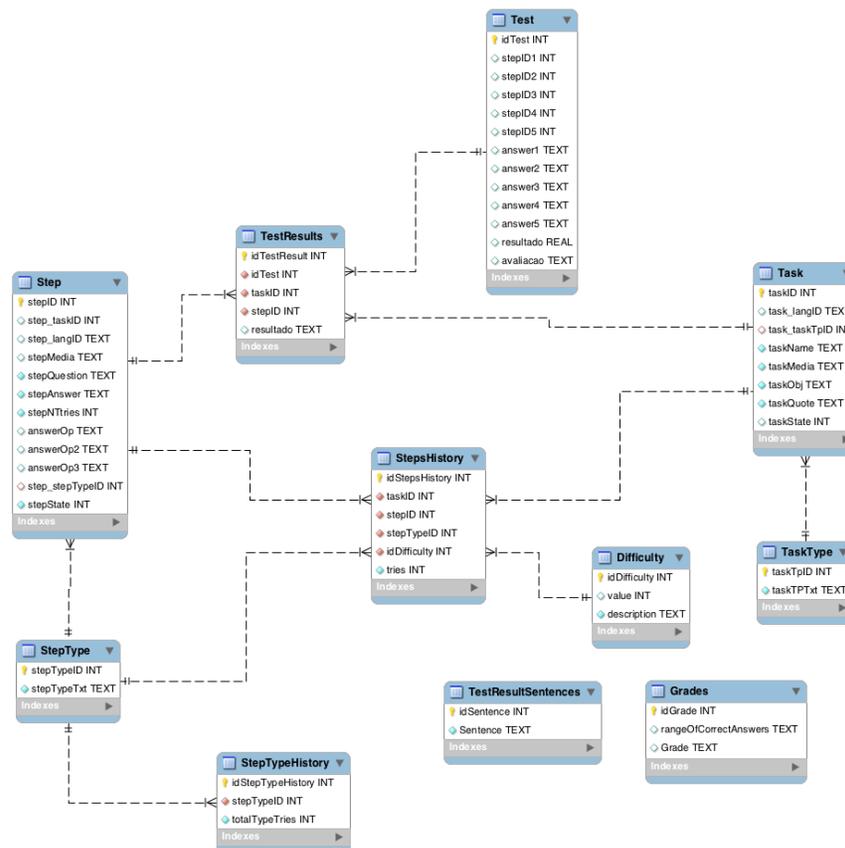


Figura 11: Esquema lógico da base de dados do sistema.

Vejam os então a descrição e caracterização das tabelas e atributos que estão integrados no esquema lógico que acabamos de apresentar. Começamos pela tabela “Task” (Tabela 2). Esta acolhe toda a informação relativa às tarefas, que no contexto da aplicação, representam uma lição ou um exercício prático. Como podemos observar na Tabela 2, esta tabela é constituída por uma chave primária, “taskID”, que serve para identificar cada tarefa inserida na tabela, por duas chaves estrangeiras, a “task_TaskTpID”, que tem como propósito identificar o tipo de tarefa - lição ou exercício escrito - e a “task_langID”, que identifica a linguagem escrita da tarefa, atualmente a aplicação suporta português e inglês. A primeira chave é relativa à tabela “TaskType” e a segunda relativa à tabela “Language”. Esta última tabela não está representada no modelo lógico. Como referido anteriormente, esta é uma tabela auxiliar no processo que já fazia parte do modelo relacional definido para o protótipo, que, como tal, não é detalhado nesta dissertação. Esta tabela é também constituída por mais 5 atributos, sendo eles o “taskName” que representa o nome da tarefa, o “taskMedia”, que contém a identificação do conteúdo multimédia da tarefa, o “taskObj”, que representa o objetivo a ser atingido com a tarefa, ou seja, o texto que irá surgir aquando do início de uma lição a explicar o que é esperado da mesma. Finalmente, os dois atributos finais representados por “taskQuote” e “taskState”, onde o primeiro representa a explicação do que a tarefa representa e o segundo o estado em que a mesma se encontra.

Atributo	Descrição	Chave	Tipo
taskID	Identificador da tarefa	P	INT
task_langID	Identificador da linguagem EN ou PT	E(“Language”)	TEXT
task_taskTpID	Identificador do tipo de tarefa	E(“TaskType”)	INT
taskName	Nome da tarefa		TEXT
taskMedia	Identificação do conteúdo multimédia da tarefa		TEXT
taskObj	Objetivo a atingir com a tarefa		TEXT
taskQuote	Explicação do que a tarefa representa		TEXT
taskState	Estado da tarefa		INT

Tabela 2: Caracterização da tabela “Task”.

A tabela “TaskType”, representada na Tabela 3, acolhe a informação que caracteriza o tipo de tarefa, uma lição ou um exercício escrito. Como podemos

observar na Tabela 3, esta tabela é constituída por uma chave primária, “taskTpID”, que tem como propósito identificar cada tipo de tarefa, e por um atributo, “taskTPTxt”, que consiste na descrição do tipo de tarefa que pode ser assumida pela aplicação. Até ao momento existem duas entradas na tabela, sendo elas “Lição” e “Exercício”.

Atributo	Descrição	Chave	Tipo
taskTpID	Identificador do tipo de tarefa	P	INT
taskTPTxt	Descrição do tipo de tarefa		TEXT

Tabela 3: Caracterização da tabela “TaskType”.

A tabela “Step”, representada na Tabela 4 aloja a informação que caracteriza uma dada etapa de uma lição. Como podemos observar na Tabela 4, esta tabela é constituída por uma chave primária, “stepID”, cujo o objetivo é identificar cada etapa inserida na tabela, por três chaves estrangeiras, a “step_taskID”, que tem como propósito identificar a tarefa a que uma etapa pertence, e a “step_langID”, que identifica a linguagem escrita da etapa, e tal como na tabela “Task”, até ao momento suporta português e inglês, e a “step_stepTypeID”, que identifica o tipo de etapa que será apresentada, podendo esta ser do tipo “DrawRecognition”, “VoiceRecognition”, “Intro”, “QUIZZ” ou “ExRepeat”. A primeira chave é relativa à tabela “Task” e a segunda relativa à tabela “Language”. Esta última tabela, como já referido, não está representada no modelo lógico. Como referido anteriormente, esta é uma tabela auxiliar no processo que já fazia parte do modelo relacional definido para o protótipo, que, como tal, não é detalhado nesta dissertação. Os restantes 8 atributos complementam a caracterização de uma etapa, sendo eles o “stepMedia”, que serve para identificar o conteúdo multimédia relativo a esta etapa, o atributo “stepQuestion”, que contém a descrição da etapa, o atributo “stepAnswer” que contém o identificador da resposta correta relativa a esta etapa, o atributo “stepNTries”, onde é estabelecido o número permitido que o aluno tem para responder a esta etapa, os atributos “answerOP”, “answerOP2” e “answerOP3” que contém o identificador das respostas alternativas quando o tipo de tarefa representa uma pergunta com escolha múltipla (QUIZZ). Por fim, o último atributo desta tabela é o “stepState”, que representa o estado da etapa.

Atributo	Descrição	Chave	Tipo
stepID	Identificador da etapa	P	INT
step_taskID	Identificador da tarefa	E("Task")	INT
step_langID	Identificador da linguagem EN ou PT	E("Language")	TEXT
stepMedia	Identificador do conteúdo multimédia da etapa		TEXT
stepQuestion	Descrição da etapa		TEXT
stepAnswer	Resposta certa da etapa		TEXT
stepNTries	Número máximo de tentativas da etapa		INT
asnwerOP	Resposta opcional da etapa		TEXT
asnwerOP2	Resposta opcional da etapa		TEXT
asnwerOP3	Resposta opcional da etapa		TEXT
step_stepTypeID	Tipo de etapa	E("StepType")	INT
stepState	Estado da etapa		INT

Tabela 4: Caracterização da tabela "Step".

A tabela "StepType", representada pela Tabela 5 acolhe a informação que caracteriza o tipo de etapa, Como podemos observar na Tabela 5, esta tabela é constituída por uma chave primária, "stepTypeId", que tem como propósito identificar cada tipo de etapa, e por um atributo, "stepTypeTxt", que consiste na descrição do tipo de etapa que pode ser assumida pela aplicação. Até ao momento existem cinco entradas na tabela, sendo elas "DrawRecognition", que consiste no reconhecimento da escrita, "VoiceRecognition", que consiste no reconhecimento de voz, "Intro", que identifica quando uma etapa é a introdução com a explicação da mesma, "QUIZZ", que consiste num tipo de pergunta em que permite escolha múltipla e, por fim, "ExRepeat", que consistem em exercícios de repetição de escrita.

Atributo	Descrição	Chave	Tipo
stepTypeId	Identificador do tipo de etapa	P	INT
stepTypeTxt	Descrição do tipo de etapa		TEXT

Tabela 5: Caracterização da tabela "StepType"..

A tabela "StepsHistory", representada na Tabela 6, armazena toda a interação do aluno com a aplicação durante as lições. Cada linha desta tabela corresponde ao comportamento do aluno em cada etapa de cada lição. Como podemos observar na Tabela 6, esta tabela é constituída por uma chave primária, "idStepsHistory", que identifica o histórico de uma etapa, e por quatro chaves estrangeiras sendo elas, a "taskID", identificador da tarefa, a "stepID", que identifica a etapa, a "stepTypeID", que identifica o tipo da etapa, e a "idDifficulty", que identifica o grau de dificuldade que um aluno apresenta

para uma determinada etapa. A primeira chave é relativa à tabela “Task”, a segunda à tabela “Step”, a terceira à tabela “StepType” e, por fim, a quarta é relativa à tabela “Difficulty”. O restante atributo desta tabela “Tries”, que representa o número de tentativas que o aluno necessitou para responder a uma determinada etapa.

Atributo	Descrição	Chave	Tipo
idStepsHistory	Identificador do histórico de uma determinada etapa	P	INT
taskID	Identificador da tarefa	E(“Task”)	INT
stepID	Identificador da etapa	E(“Step”)	INT
stepTypeID	Identificador do tipo de etapa	E(“StepType”)	INT
idDifficulty	Identificador do grau de dificuldade do aluno em relação a uma determinada etapa	E(“Difficulty”)	INT
Tries	Número de tentativas que o aluno precisou para responder a uma determinada etapa		INT

Tabela 6: Caracterização da tabela “StepsHistory”.

A tabela “Difficulty”, representada na Tabela 7, permite acolher o estado do aluno em relação a uma determinada etapa. Como podemos observar na Tabela 7, esta tabela é composta por uma chave primária, que identifica o grau de dificuldade do aluno em relação a uma determinada etapa, e por dois atributos sendo eles, o atributo “Value”, que contém o valor atribuído consoante o grau de dificuldade do aluno numa determinada etapa, e o atributo “Description”, que representa a descrição da dificuldade do aluno numa determinada etapa. Até ao momento esta tabela contém cinco registos: ‘Aprendido’, ‘Pode Melhorar’, ‘Alguma Dificuldade’, ‘Dificuldade’ e ‘Muita Dificuldade’. A dificuldade é atribuída conforme o valor obtido pelo aluno em cada etapa.

Atributo	Descrição	Chave	Tipo
idDifficulty	Identificador do grau de dificuldade do aluno em relação a uma determinada etapa	P	INT
Value	Valor atribuído consoante o grau de dificuldade do aluno numa determinada etapa		INT
Description	Descrição da dificuldade do aluno numa determinada etapa		TEXT

Tabela 7: Caracterização da tabela “Difficulty”.

A tabela “StepTypeHistory” representada na Tabela 8, é utilizada na definição do tipo de etapa em que o aluno tem mais dificuldade. Como podemos

observar na Tabela 8, esta tabela é constituída por uma chave primária, “idStepTypeHistory”, que identifica o registo do histórico do tipo de uma determinada etapa, e por uma chave estrangeira, “stepTypeID”, que identifica o tipo de etapa. Esta chave é relativa à tabela “StepType”. O restante atributo da tabela, o atributo “totalTypeTries”, tem como objetivo somar o campo “Tries” da tabela “StepsHistory” por cada tipo de etapa, isto é, faz a soma do número de tentativas que o aluno necessitou em cada tipo de etapa, por exemplo, no tipo de etapa “1”, correspondente a “DrawRecognition”, o aluno, de todas as perguntas que já realizou deste tipo, precisou, no total de 5 tentativas. Com isto, conseguimos perceber qual o tipo de etapa em que o aluno tem mais dificuldade.

Atributo	Descrição	Chave	Tipo
idStepTypeHistory	Identificador do registo do histórico do tipo de uma determinada etapa	P	INT
stepTypeID	Identificador do tipo de etapa	E(“StepType”)	INT
totalTypeTries	Soma do campo <i>Tries</i> da tabela ‘StepsHistory’ por cada tipo de etapa		INT

Tabela 8: Caracterização da tabela “StepTypeHistory”

A tabela “Test”, representada na Tabela 9, acolhe a informação do teste de avaliação que o aluno irá realizar. Como podemos observar na Tabela 9, esta tabela é constituída por uma chave primária, “idTest”, que identifica cada teste da tabela, e por cinco chaves estrangeiras, “stepID1”, “stepID2”, “stepID3”, “stepID4” e “stepID5”. Estas chaves são todas elas relativas à tabela “Step” e representam o identificador de cada etapa que constitui um teste. Um teste é composto por um máximo de cinco perguntas. Estas chaves representam as diferentes etapas nas quais o aluno tem mais e menos dificuldades. Cada entrada nesta tabela representa o teste mais recente que o aluno poderá realizar. É uma tabela que será atualizada consoante as lições que o aluno realizar. As perguntas do teste podem ser as etapas de uma só lição ou de diferentes lições, dependendo sempre das dificuldades e facilidades que o aluno encontrou. Um teste tem que ser obrigatoriamente constituído por cinco perguntas. Os restantes atributos desta tabela representam a resposta que o aluno deu a uma determinada pergunta, atributos “answer1”, “answer2”, “answer3”, “answer4” e “answer5”, o resultado numérico obtido no teste, representado pelo atributo “Resultado” e, por fim, o atributo “Avaliação” que representa o resultado em texto obtido no teste.

Atributo	Descrição	Chave	Tipo
idTest	Identificador do teste	P	INT
stepID1	Identificador de uma determinada etapa	E("Step")	INT
stepID2	Identificador de uma determinada etapa	E("Step")	INT
stepID3	Identificador de uma determinada etapa	E("Step")	INT
stepID4	Identificador de uma determinada etapa	E("Step")	INT
stepID5	Identificador de uma determinada etapa	E("Step")	INT
answer1	Resposta do aluno à primeira pergunta do teste		TEXT
answer2	Resposta do aluno à segunda pergunta do teste		TEXT
answer3	Resposta do aluno à terceira pergunta do teste		TEXT
answer4	Resposta do aluno à quarta pergunta do teste		TEXT
answer5	Resposta do aluno à quinta pergunta do teste		TEXT
Resultado	Resultado numérico obtido no teste		REAL
Avaliação	Resultado em texto obtido no teste		TEXT

Tabela 9: Caracterização da tabela "Test".

A tabela "TestResults", representada pela Tabela 10, é uma tabela auxiliar que permite relacionar as perguntas de um teste, que são as etapas de uma lição, com a sua respetiva lição e o respetivo resultado no teste. Isto é, cada entrada desta tabela representa uma, e uma só, pergunta de um determinado teste. Como podemos observar pela Tabela 10, esta tabela é constituída por uma chave primária, "idTestResult", que tem como objetivo identificar cada resultado obtido pelo aluno numa determinada pergunta, e por três chaves estrangeiras, sendo elas a "idTest", que identifica a que teste o resultado desta pergunta pertence, a taskID, que identifica a tarefa a que a pergunta pertence e a "stepID" que identifica a etapa a que a pergunta pertence. Estas chaves são relativas às tabelas "Test", "Task" e "Step" respetivamente. O atributo "Resultado", corresponde ao resultado obtido numa determinada pergunta do teste, que poderá ser do tipo "Correto" ou "Errado".

Atributo	Descrição	Chave	Tipo
idTestResult	Identificador do resultado de uma pergunta do teste	P	INT
idTest	Identificador do teste	E("Test")	INT
taskID	Identificador da tarefa	E("Task")	INT
stepID	Identificador da etapa	E("Step")	INT
Resultado	Resultado obtido numa determinada pergunta de um teste		TEXT

Tabela 10: Caracterização da tabela "TestResults".

A tabela "TestResultSentences", representada na Tabela 11, é, também, uma tabela auxiliar, que tem como propósito armazenar as frases que vão ser apresentadas pela aplicação ao aluno, quando este conclui um teste. As frases

apresentadas dependem da classificação obtida pelo aluno. Como podemos observar na Tabela 11, esta tabela é constituída por uma chave primária, “idSentence”, que identifica a frase a ser apresentada, e pelo atributo “Sentence” que contém as frases predefinidas pelo sistema. Este atributo é relativo apenas à frase inicial que o sistema apresenta após um teste, o resto da frase é originada dinamicamente pelo sistema como iremos explicar a seguir. Até ao momento, a tabela “TestResultSentences” tem cinco frases possíveis, sendo elas:

- **“Estou a ver que não andas a estudar, precisas de rever”** - Que é apresentada quando o aluno falha todas as perguntas do teste.
- **“Não foi desta. Antes de voltar a tentar precisas de estudar mais”** - Que é apresentada quando aluno acerta uma ou duas perguntas do teste.
- **“Boa! Mas precisas de estudar mais”** - Que é apresentada quando o aluno acerta três perguntas do teste.
- **“Muito bem! Estás no bom caminho, acertaste quase tudo! Precisas de rever”** - Que é apresentada quando o aluno acerta quatro perguntas do teste.
- **“Parabéns!! Acertaste todas as perguntas, estás pronto para aprender lições novas.”** - Que é apresentada quando o aluno responde corretamente a todas as perguntas.

Todas estas frases com exceção da última são complementadas com as perguntas que o aluno precisa de rever.

Atributo	Descrição	Chave	Tipo
idSentence	Identificador da frase	P	INT
Sentence	Frase		TEXT

Tabela 11: Caracterização da tabela “TestResultSentences”

Por último, temos a tabela “Grades”, representada na Tabela 12, uma outra tabela auxiliar. Esta tabela tem como objetivo suportar a atribuição de uma nota consoante o número de respostas certas num determinado teste. Como podemos observar na Tabela 12, esta tabela é constituída por uma chave primária, “idGrade”, que identifica a nota de um determinado teste, e por dois atributos. O primeiro, “rangeOfCorrectAnswers”, corresponde ao intervalo de respostas certas que um aluno obteve num teste, e o segundo, “Grade” é a nota que um aluno obteve num determinado teste.

Atributo	Descrição	Chave	Tipo
idGrade	Identificador da nota de um teste	P	INT
rangeOfCorrectAnswers	Intervalo de respostas certas para atribuir uma nota		TEXT
Grade	Nota de um teste		TEXT

Tabela 12: Caracterização da tabela “Grades”

4.2. O Algoritmo de Profiling e O Método de Avaliação

Apesar de não ser nenhum dos algoritmos já existentes e já explicados nesta dissertação, o algoritmo de profiling desenvolvido pareceu-nos o mais adequado ao caso de estudo. Além disso segue as regras base para um processo de *profiling* típico. Como já tivemos a oportunidade de referir anteriormente, no capítulo 3, um processo de *profiling* considera 7 etapas diferentes. Nesta secção iremos descrever a sua aplicação ao nosso caso em particular.

4.2.1. Base Preliminar

O processo de *profiling* começa com uma especificação do domínio do problema e a respetiva identificação dos objetivos de análise. No caso deste trabalho, o objetivo inicial passava por conseguir identificar o perfil do aluno para mais tarde o podermos avaliar. Para isso, tivemos que ser capazes de perceber o comportamento do aluno perante certos cenários de aprendizagem e, com isso, entender quais as suas maiores dificuldades e facilidades. O propósito de conseguirmos caracterizar o perfil de um aluno, é para conseguirmos ter a capacidade de criar um teste especialmente orientado ao aluno, ou seja, um teste constituído pelas suas maiores dificuldades e as suas maiores facilidades. Com a introdução de perguntas nas quais o aluno tem mais facilidade, pretende-se manter o aluno motivado durante o teste para não ser só constituído por perguntas que o aluno tenha mais dificuldade em responder. Uma forma de incentivar o aluno e dar-lhe confiança. É importante.

4.2.2. Recolha de Dados

Na parte da recolha de dados são armazenados os dados que se consideram importantes para o processo. No nosso caso, os dados considerados relevantes, que suportam o algoritmo, são aqueles que são recolhidos durante a realização das lições. Cada uma das lições é realizada por etapas, tal como

está definido no método de João de Deus. Para aprender os conteúdos de uma lição, o aluno tem de ser capaz de conseguir desenhar, pronunciar e identificar uma determinada letra. De forma a conseguirmos armazenar esta informação, foram criadas tabelas específicas na base de dados do sistema, nomeadamente: “Difficulty”, “StepTypeHistory” e “StepsHistory”. Na tabela “Difficulty” definimos os diversos níveis de dificuldade que o aluno pode enfrentar ao longo de uma lição. Cada um destes níveis de dificuldade tem um valor associado. Os dados contidos nesta tabela estão organizados de forma crescente, do menor nível de dificuldade para o maior nível de dificuldade. O nível de dificuldade ‘1’ corresponde ao estado ‘Aprendido’ e o nível de dificuldade ‘5’, o mais alto da tabela, ao estado ‘Muita Dificuldade’.

	idDifficulty	value	description
1	1	-1	Aprendido
2	2	0	Pode Melhorar
3	3	1	Alguma Dificuldade
4	4	2	Dificuldade
5	5	3	Muita Dificuldade

Figura 12: Extrato do conteúdo tabela “Difficulty”.

O nível de dificuldade é atribuído consoante o número de tentativas que o aluno usa para responder a uma determinada etapa de uma tarefa. Isto é, se o aluno acertar à primeira tentativa é-lhe atribuído o nível ‘1’, à segunda tentativa, o nível ‘3’, na terceira tentativa o nível ‘4’ e caso não consigo responder dentro das 3 tentativas é-lhe atribuído o nível 5. Estes níveis aplicam-se caso seja a primeira vez que o aluno está a responder a uma determinada etapa, caso contrário verificar-se-á o histórico relativo a essa mesma etapa. Neste contexto, aparece a tabela “StepsHistory”, que tem como objetivo guardar todos s registos relativos à atividade do aluno durante a realização das suas várias lições. Desta forma, conseguimos ter um histórico das várias tentativas realizadas pelo aluno e, com isso, melhorar os seus resultados finais. Cada registo na tabela “StepsHistory” corresponde a uma tentativa de execução de uma etapa, ou seja, ao registo daquilo que o aluno está a responder numa dada etapa. Durante esse processo verificamos se é a primeira vez que a se está a executar e, se for, atribui-se o nível de dificuldade consoante o número de tentativas, tal como já foi referenciado anteriormente. Porém, se não for a primeira vez, apenas mudamos o nível de dificuldade. Aqui, o que fazemos, é ver o último registo efetuado dessa etapa e guardamos

o respetivo valor do nível de dificuldade. Seguidamente, verificamos na tabela “Difficulty” a que valor corresponde esse nível e, a ao valor do nível de dificuldade obtido, somamos-lhe o valor do nível correspondente ao número de tentativas da nova tentativa de execução. Com isto, conseguimos fazer com que, por exemplo, um aluno acerte à primeira, na primeira vez que tenta fazer um exercício (valor -1), e, quando o fizer pela segunda vez, só acertará à segunda (valor 1) - assim, incrementamos -1 com 1 e ficamos com o valor 0 que corresponde ao nível de dificuldade 2 (‘Pode Melhorar’). Com isto, queremos dizer que, o aluno poderá ter alguma sorte na sua primeira vez, tendo a segunda tentativa de execução do exercício um efeito de equilíbrio nos seus resultados. Ao valor incrementado, o algoritmo não permite que este seja superior a 3, valor este que corresponde ao nível 5 da tabela. Por fim, temos a tabela “StepTypeHistory”, que nos permite saber qual o tipo de etapa que o aluno tem mais dificuldade, ou seja, indica-nos se o aluno tem mais dificuldade em desenhar, pronunciar ou identificar as letras. Para que isto seja possível, o que fazemos sempre que um aluno termina a execução de uma etapa é consultar a tabela “StepsHistory” e verificamos qual o último “idStepType” e somamos o número de tentativas que o aluno necessitou para completar a etapa com o campo “totalTypeTries” da tabela “StepTypeHistory” cujo “idStepType” seja igual ao último realizado pelo aluno. Desta forma mantemos o número total de tentativas por cada tipo de etapa sempre atualizado conseguindo perceber em qual tipo de etapa o aluno tem mais dificuldade.

4.2.3. Preparação dos Dados

Nesta fase de preparação faz-se a eliminação de dados redundantes. No método que utilizamos, quando preparamos os dados para serem analisados temos que garantir que estes são os mais recentes. Isto significa que, quando construímos o teste, procurando as perguntas em que o aluno tem mais dificuldade, começamos pelo nível 5, o nível máximo. Depois, no conjunto de resultados obtidos, poderá estar um registo de nível 5, que já não seja o mais recente, isto poderá significar que o aluno poderá já ter feito aquela pergunta, tendo obtido outro tipo de rendimento e, com otal, poderá este resultado não equivaler a um caso de nível 5. Nestes casos, o método que adotámos trabalhará sobre o conjunto de resultados obtidos e verificará se tal resultado

corresponde à entrada mais recente da tabela. Se for esse o caso, o resultado mantém-se. Caso contrário, remove-se o resultado da lista final de possíveis resultados. O mesmo se aplica sobre as perguntas com o maior grau de facilidade.

4.2.4. Mineração dos Dados

Nesta fase, os dados são analisados com um algoritmo que foi desenvolvido especificamente para este objetivo. Para este trabalho, definimos que um teste seria constituído por 3 perguntas, nas quais o aluno sentiria mais dificuldade, e por mais 2 perguntas nas quais se sentiria mais confortável. De forma a conseguirmos criar um teste de avaliação baseado nas maiores dificuldades e facilidades do aluno, temos que analisar os dados contidos na tabela “StepsHistory”. O que o algoritmo escolhido começa por fazer é uma consulta à tabela “StepsHistory” onde recolhe todos os registos em que o campo “idDifficulty” tem o valor “5”, que corresponde ao nível máximo de dificuldade que o aluno pode ter. De seguida, como já explicado anteriormente, o algoritmo verifica se os resultados são os mais recentes, ao aplicar este filtro verificamos o tamanho da lista obtida, se for igual a 3, já temos as perguntas com maior dificuldade do aluno e passamos a procurar pelas de maior facilidade. Se o tamanho da lista obtida for maior que 3, então temos que ficar com as de maior dificuldade, para isso, fazemos uma consulta à tabela “StepTypeHistory” de forma a perceber qual o tipo de etapa em que ele tem mais dificuldade, e aí filtramos de forma a perceber das perguntas com mais dificuldade quais são os tipos de perguntas em que ele tem mais dificuldade, até obter o tamanho 3.

Se o tamanho da lista obtida for inferior a 3 então, fazemos uma nova pesquisa à tabela “StepsHistory” desta vez com o nível de dificuldade 4, voltamos a filtrar esses dados para verificar se são os mais recentes, e verificamos se o tamanho total da lista com as perguntas de nível 5 e 4 é igual, superior ou inferior a 3. Caso seja igual, termina o processo, caso seja superior, voltamos a filtrar com recurso à tabela “StepTypeHistory” e caso seja inferior votamos a fazer uma pesquisa à tabela “StepsHistory” desta feita com o nível 3. Este processo repete-se até obtermos o tamanho total desejado, 3.

O processo é o mesmo na procura das perguntas com mais facilidade, mas começa-se a pesquisa à tabela “*StepsHistory*” pelo nível de dificuldade mínimo, 1.

4.2.5. Interpretação

Nesta fase, os padrões resultantes são avaliados por relevância e validados por especialistas no domínio da aplicação. Na interpretação dos resultados finais obtidos, o que o método desenvolvido faz é verificar se alguma das perguntas que formam o teste, já saiu num outro teste que o aluno tenha realizado. Para isto temos a tabela “*Test*”, tabela que é armazenada todos os testes que o aluno já realizou. Caso alguma pergunta seja repetida, substituímos por uma com o mesmo nível de dificuldade.

Depois, na fase de aplicação, os vários perfis desenvolvidos são aplicados, o que origina a geração do teste de avaliação final. Na última fase, a decisão institucional, a instituição decide quais as ações ou políticas que se irão aplicar aos grupos ou aos indivíduos cujos dados se relacionem com o perfil em questão. Porém, dado que esta fase não está incluída no âmbito dos trabalhos desta dissertação não nos alargaremos mais quanto a ela.

Em relação ao modelo de avaliação desenvolvido, a estrutura do modelo iria de encontro ao modelo desenvolvido por Kirkpatrick, modelo este que já foi descrito nesta dissertação. Este modelo, numa primeira fase, começa por avaliar as reações dos participantes, de seguida avalia o conhecimento antes e depois de se aplicar o programa de treino, posteriormente avalia a mudança no comportamento dos participantes e, por fim, avalia os resultados. De forma a conseguirmos perceber os resultados do modelo aplicado, teríamos de utilizar não só um conjunto de dados maior, como pôr em prática o modelo desenvolvido numa sala de aula, por exemplo. O modelo desenvolvido foca-se na primeira fase do modelo de avaliação, reações, ou seja, apresentámos uma possível forma de avaliar o aluno, em que as reações ao mesmo teriam que ser analisadas para posteriormente podermos tirar conclusões dos resultados obtidos para perceber se o aluno está a progredir no seu percurso académico e, também avaliar os resultados finais de forma a perceber se estamos a conseguir tirar proveito do método de avaliação definido e se este é de facto, o melhor método a ser utilizado.

Como já referido, o teste criado é constituído por cinco perguntas, sendo três delas a representação das maiores dificuldades do aluno, e duas delas as suas maiores facilidades. Este modelo de teste faz parte do modelo de avaliação criado. Foi decidido, após um estudo intensivo, que a melhor forma de avaliar um aluno, principalmente uma criança nos seus 5/6 anos, era abordar as suas dificuldades e as suas facilidades. Um teste constituído só pelas suas maiores dificuldades acabaria por desanimar o aluno ficando este desinteressado e desmotivado para completar o teste com a devida concentração. A integração das perguntas em que o aluno se sente mais confortável faz com que este se sinta motivado, e que mantenha o raciocínio desperto e a concentração máxima de forma a terminar o teste nas melhores condições. Como não deixa de ser um exercício de avaliação com o objetivo de melhorar a aprendizagem do aluno, pareceu-nos mais adequado que o teste fosse maioritariamente constituído pelas as maiores dificuldades.

O teste de avaliação tem, também, como objetivo orientar o estudo do aluno. Para suportar esse fim foram criadas na base de dados várias tabelas, em particular: "Test", "TestResultSentences", "TestResults" e "Grades". A tabela "Test", como já referido, armazena todos os testes realizados pelo aluno juntamente com avaliação que este obteve em cada um deles. A tabela "TestResults" serve como apoio à tabela "Test". Nesta tabela, cada entrada corresponde a uma pergunta de um teste e à resposta que o aluno deu a essa pergunta. A tabela "TestResultSentences" contém uma frase genérica por cada avaliação que o aluno possa ter obtido, que será apresentada ao aluno no final de cada teste. Por fim, a tabela "Grades" é composta pelas diversas avaliações que o aluno poderá obter no final de cada teste. Durante o teste, à medida que o aluno vai respondendo, um objeto, "teste", vai sendo atualizado, isto é, a resposta que um aluno dá a uma pergunta é armazenada neste objeto. O aluno tem a possibilidade de mudar a resposta que deu a uma pergunta, ou seja, o aluno pode voltar para trás e mudar a resposta que deu numa pergunta. O teste termina quando o aluno responder às 5 perguntas e carregar no botão finalizar. O sistema não permite respostas em branco. Ao clicar no botão finalizar, o sistema guarda os resultados na tabela "TestResults", na figura 13 podemos ver um exemplo de um teste resolvido. Como se pode observar, temos cinco registos nesta tabela todos com o mesmo "idTest". Optamos por apenas guardar o resultado da resposta como um "Correcto" ou "Errado"

porque para resultados finais precisamos apenas da contagem do número de respostas certas.

	idTestResult	idTest	idLesson	idStep	resultado
1	1	1	1	3	Errado
2	3	1	3	3	Correcto
3	6	1	2	3	Correcto
4	10	1	3	2	Correcto
5	15	1	3	4	Correcto

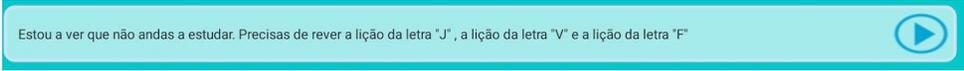
Figura 13: Registos relativos a um teste armazenado na tabela “TestResults”.

Também, ao selecionarmos o botão finalizar, o sistema faz a contagem do número de perguntas corretas. Com esse número realiza uma pesquisa sobre a tabela “TestResultSentences” para recolher a frase que se adequa ao resultado final. Na figura 14, podemos ver um fragmento da tabela “TestResultSentences”. Se o aluno acertar 3 perguntas, por exemplo, a frase “Boa! Mas precisas de estudar mais” vai-lhe ser apresentada no ecrã juntamente com as perguntas que precisa de rever.

	idSentence	Sentence
1	0	Estou a ver que não andas a estudar, precisas de rever
2	1	Não foi desta. Antes de voltar a tentar precisas de estudar mais
3	2	Boa! Mas precisas de estudar mais
4	3	Muito bem! Estás no bom caminho, acertaste quase tudo! Precisas de rever
5	4	Parabéns!! Acertaste todas as perguntas, estás pronto para aprender lições novas.

Figura 14: Excerto da tabela “testResultSentences”.

Como um dos objetivos da avaliação é orientar o aluno no seu estudo, foi desenvolvido também um sistema de construção de frases que atua consoante as perguntas ou lições que vão aparecendo no teste e as respetivas respostas que o aluno vai dando. Se o aluno não acertar nenhuma pergunta ou se só acertar apenas uma, para além da frase genérica, é-lhe recomendado que estude as lições das perguntas que falhou, como representado na Figura 15.



Estou a ver que não andas a estudar. Precisas de rever a lição da letra "J", a lição da letra "V" e a lição da letra "F"

Figura 15: Exemplo de um teste em que o aluno não acertou nenhuma pergunta

Assim, na realização de um teste sobre a pronúncia da letra J, pela identificação da letra V, pela identificação da letra F, pela pronúncia da letra V e pela pronúncia da letra F, se o aluno errar todas as perguntas, além da frase genérica, o sistema avisa o aluno de que ele precisa fazer a revisão completa das lições correspondentes, nomeadamente as referentes às letras J, V e F. Caso o aluno acerte pelo menos em duas perguntas, então o sistema apresenta, para além da frase genérica, as etapas específicas que ele precisa de voltar a estudar.



Não foi desta. Antes de voltar a tentar precisas de estudar mais como se escreve a letra "J", como se lê a letra "V" e como se lê a letra "F"

Figura 16: Exemplo de um teste em que o aluno acertou duas perguntas

Como podemos observar na figura 16, um aluno que erre especificamente a escrita da letra J, como se lê a letra V e a letra F o sistema informa aluno da necessidade da repetição das lições relativas a essas letras. Caso o aluno acerte todas as questões, o sistema informa o aluno sobre o seu resultado e sugere-lhe a realização de lições mais avançadas (figura 17)



Parabéns!! Acertaste todas as perguntas, estás pronto para aprender lições novas.

Figura 17: Exemplo de um teste em que o aluno acertou todas as perguntas.

Para além da frase final, o sistema complementa a sua informação com alguns elementos gráficos adequados aos resultados obtidos. No caso do aluno ter acertado em tudo, por exemplo, o aluno percebe isso através de um conjunto de 5 estrelas, completamente preenchidas, e a informação de uma avaliação de "Excelente", como demonstrado na Figura 18.



Figura 18: Visualização gráfica da nota para o exemplo do "Excelente"

Neste caso, apenas demonstrámos o caso relativo à obtenção de uma classificação de “Excelente”. O número de estrelas preenchidas e o desenho da avaliação variam consoante a classificação obtida. Analisaremos mais à frente os diferentes cenários que esta imagem pode apresentar. O aluno ao terminar um teste (figura 19) o sistema volta ao seu ambiente principal (figura 20).

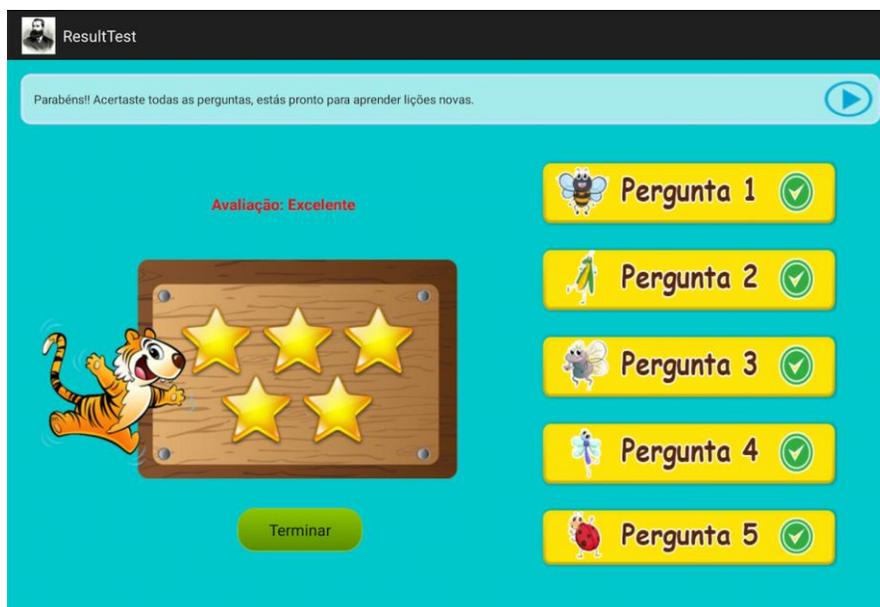


Figura 19: Finalização da realização de um teste.

4.3. Demonstração das Funcionalidades do Sistema

De forma a podermos demonstrar as funcionalidades desenvolvidas, vamos analisar cada uma das opções apresentadas no menu inicial do sistema (Figura 20). A partir deste menu, temos acesso aos serviços que o sistema tem atualmente implementados, nomeadamente:

1. **Cartilha** – em que o utilizador pode ler e consultar uma cópia digital, em formato .pdf, da Cartilha Maternal João de Deus.
2. **Lições** – na qual o sistema apresenta as lições da cartilha que estão disponíveis - até ao momento foram preparadas as cinco primeiras lições: “Vogais”, “A letra V”, “A letra F”, “A letra J” e “A letra T”.
3. **Exercícios** – opção em que temos acesso a um conjunto de atividades complementares ao processo de aquisição de conhecimentos, como, por exemplo, a escrita do ditongo “ai” ou a repetição da vogal “i”.
4. **Plano** – opção do sistema que permite aos professores criarem planos de trabalho para que, por exemplo, os alunos possam realizar algum trabalho em casa.
5. **Estatísticas** – esta funcionalidade permite ao aluno visualizar o seu progresso no processo de aprendizagem ao longo das tarefas que vão realizando.
6. **Perfil** – em que podemos observar a cadeia de *Markov* desenvolvida, de acordo com o perfil de aprendizagem do aluno.

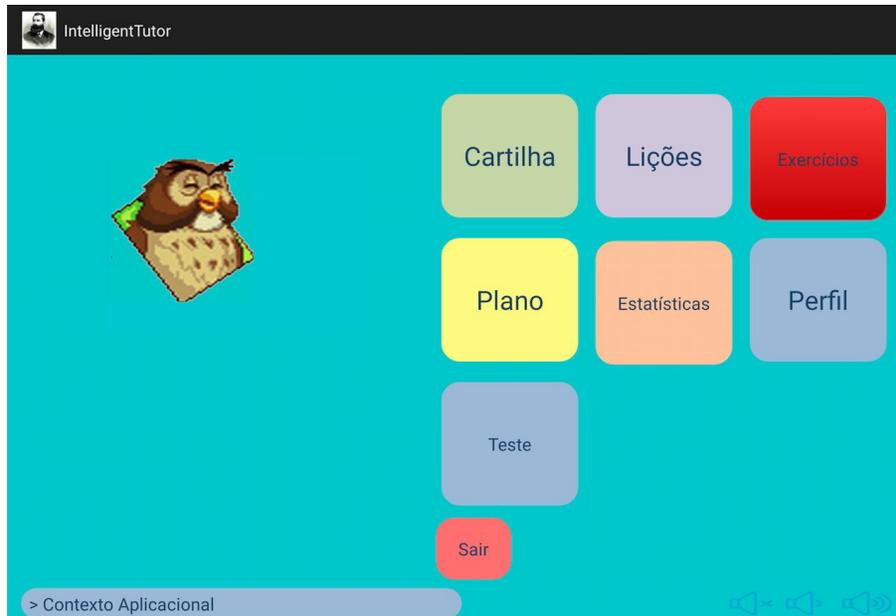


Figura 20: Menu das funcionalidades até agora implementadas no sistema

As opções disponíveis no menu principal do sistema não sofreram alterações em relação ao protótipo que inicialmente foi desenvolvido. De seguida, expomos a opção do sistema que demonstra o trabalho realizado no âmbito desta dissertação. Para melhor percebermos o funcionamento da construção do teste final, vamos também demonstrar o menu das lições. Os menus apresentam cores fortes e vivas de forma a conseguir captar a atenção da criança. Também para esse efeito, durante toda a utilização da aplicação fez-se com que a animação da “coruja” estivesse sempre presente. Esta lê os textos e interage com o aluno ao longo da sua utilização do sistema.

4.3.1. As Lições

A opção “Lições” representa as várias lições descritas na Cartilha João de Deus. Ao optarmos por esta opção o sistema apresenta-nos um submenu das várias lições disponíveis no sistema (figura 21).

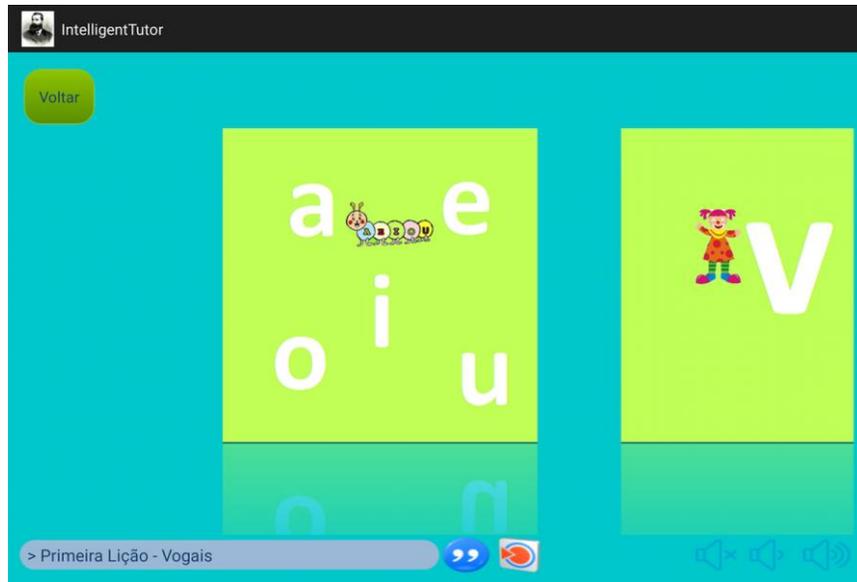


Figura 21: Menu das lições

Se, por exemplo, escolhermos a lição “V”, o sistema faz-nos uma pequena apresentação sobre a lição escolhida (figura 22). Uma lição é sempre constituída por uma parte de pronúnciação e outra de identificação e escrita da letra escolhida. Quando o aluno acerta uma pergunta, o sistema apresenta um símbolo de “certo” em cor verde, de forma a que o aluno perceba de forma clara que acertou na resposta (figura 24). No caso de errar, é apresentado ao aluno um símbolo com uma “cruz” a vermelho (figura 25). Sempre que um aluno acerta ou erra uma pergunta, o sistema guarda essa informação na tabela “*StepsHistory*” do sistema. Como já referido, esta tabela serve de apoio à construção dos testes de avaliação.

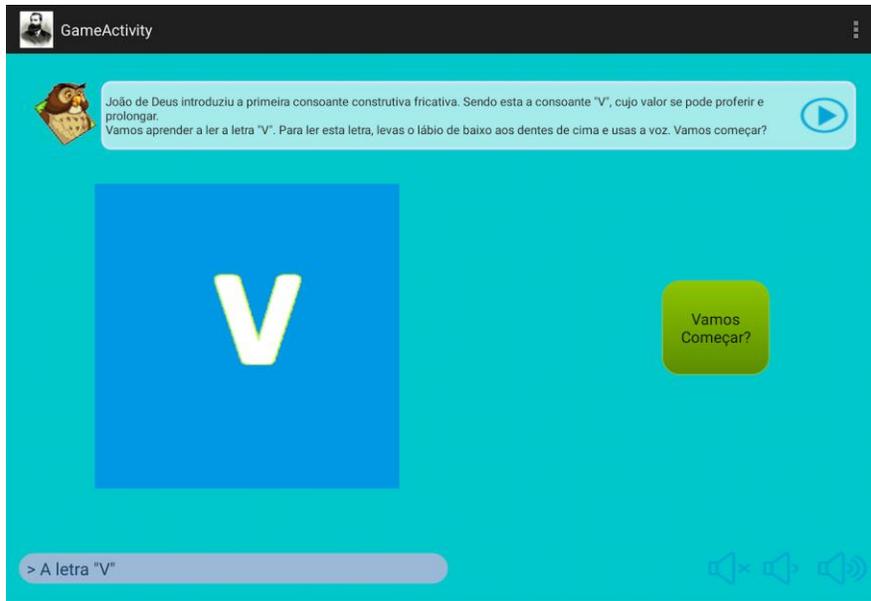


Figura 22: Apresentação inicial da lição da letra "V"

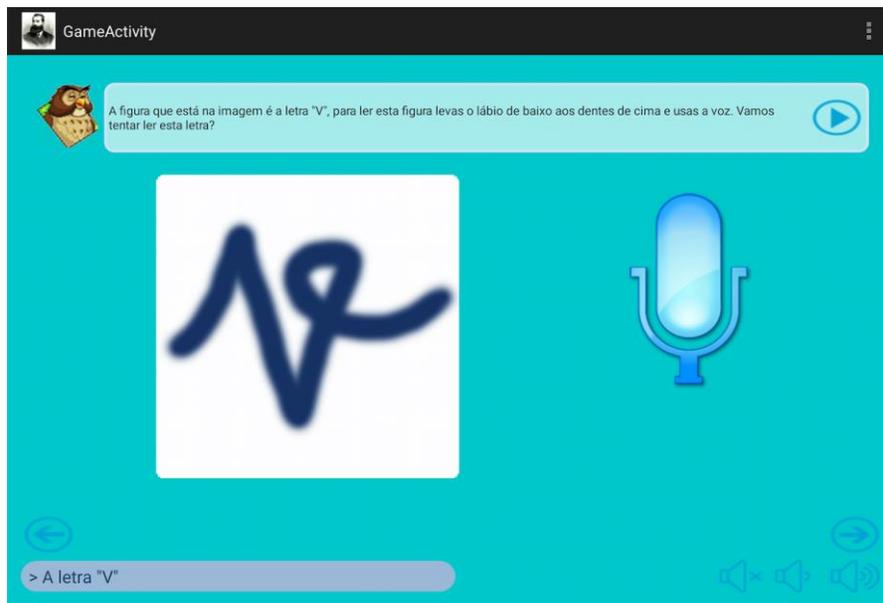


Figura 23: Exemplo de uma pergunta de pronúncia



Figura 24: Exemplo a resposta do sistema quando o aluno acerta uma pergunta

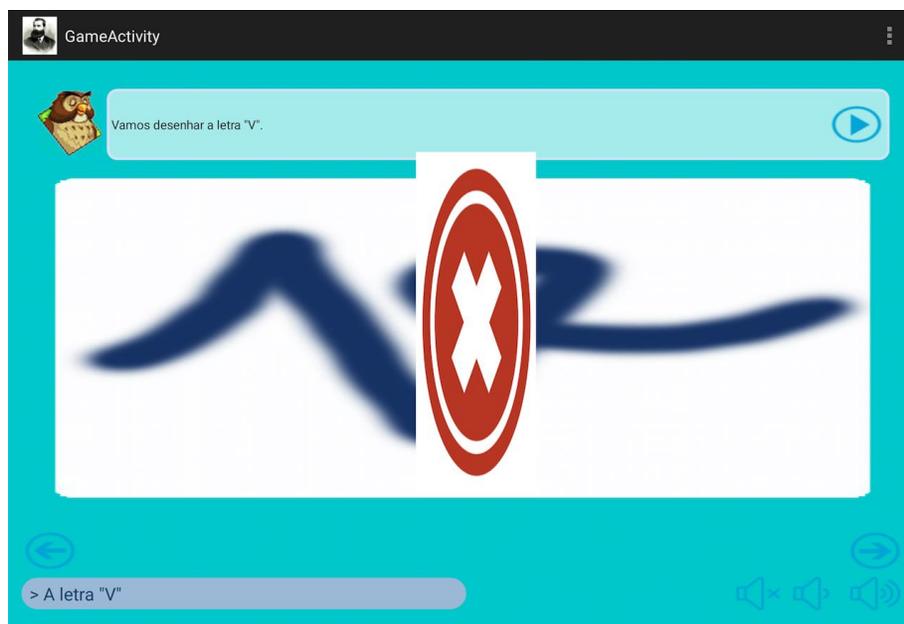


Figura 25: Exemplo da resposta do sistema quando o aluno erra uma pergunta

4.3.2. Os Testes de Avaliação

A opção “Teste”, tal como já foi explicado neste documento, dá-nos acesso a um conjunto de cinco perguntas baseadas nas maiores dificuldades e

facilidades que o aluno demonstrou ao longo da execução das lições que foi realizando. Ao selecionarmos a opção “Teste”, o sistema conduz-nos para o ambiente de realização do teste de avaliação no qual podemos iniciar a realização do teste pretendido (figura 26). Através de um botão específico – “Iniciar Teste” – o sistema faz a criação do teste, passando de seguida para o ambiente de realização, em concreto, do teste de avaliação. Caso o aluno saísse deste ambiente de criação do teste e voltasse ao ambiente das lições para a realização de uma ou mais lições, qualquer solicitação para a realização de um teste ia provocar a geração de um teste diferente, uma vez que as perguntas que constituem o teste mudam consoante os resultados obtidos pelo aluno nas lições que realizou.

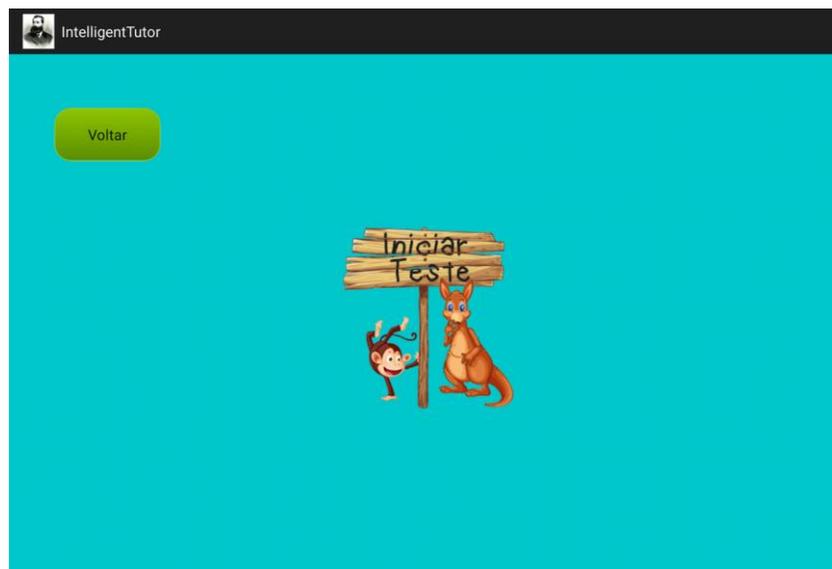


Figura 26: Início do teste de avaliação

Após a solicitação do teste ter sido concretizada, o teste é apresentado num formato idêntico ao das lições. Porém, a forma como as perguntas aparecem não acompanham as lições que foram realizadas. Um teste é um aglomerado de questões das várias lições, ou seja, dificilmente encontramos num teste a matéria de uma lição completa. Usualmente, um teste é constituído por cinco perguntas de cinco lições diferentes, podendo contudo, mediante a história da realização das lições, incluir várias perguntas de uma mesma lição. Durante a realização do teste o aluno só vê uma pergunta de cada vez. Ao responder à questão apresentada, independentemente da correção da sua resposta, o sistema passa para a pergunta seguinte. Como já foi referido, o aluno poderá saltar a resposta a uma pergunta ou voltar a responder a uma pergunta já

respondida anteriormente. Quando o aluno tiver respondido às cinco perguntas do teste, o sistema ativa o botão “Finalizar” que surge no ecrã quando o aluno der por terminado o teste. Nessa altura ser-lhe-á apresentada a avaliação do teste que acabou de realizar. Na figura 27 podemos ver uma representação gráfica do conjunto de perguntas de um dos testes que o sistema gerou, após terem sido realizadas várias lições. O teste em questão envolveu questões relativas à identificação e à escrita da letra “J”, e à pronúncia e escrita da letra “V”.

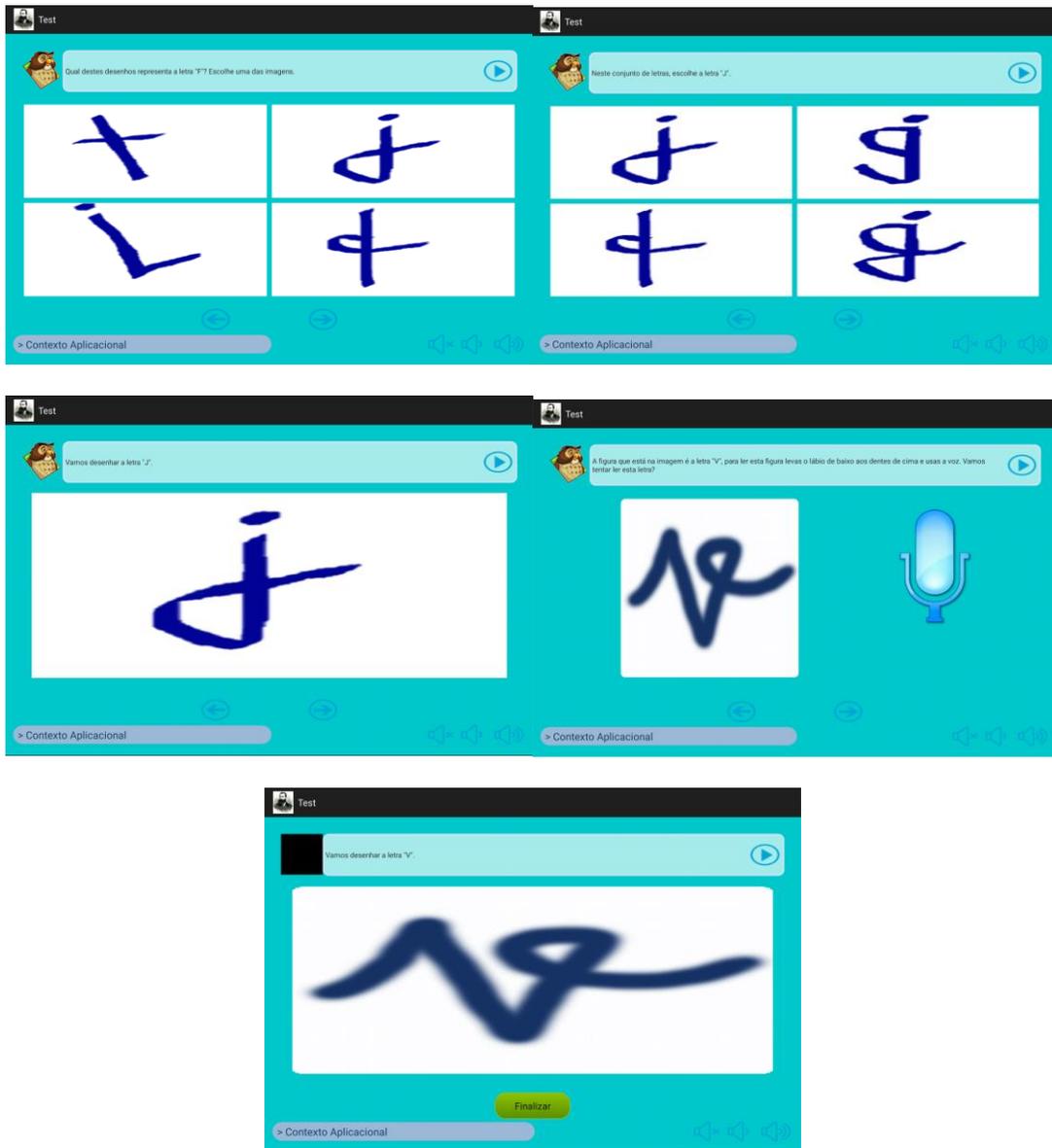


Figura 27: Representação de um teste gerado pelo sistema

4.3.3. Avaliação

Na apresentação da avaliação ao aluno, optámos pela utilização de várias cores, de tons fortes e garridas, e pela utilização de animações, como cartoons, e mostrando a classificação através de um sistema de estrelas, para complementar o texto apresentado. Na figura 28 podemos ver os diferentes tipos de avaliação que o aluno pode obter.

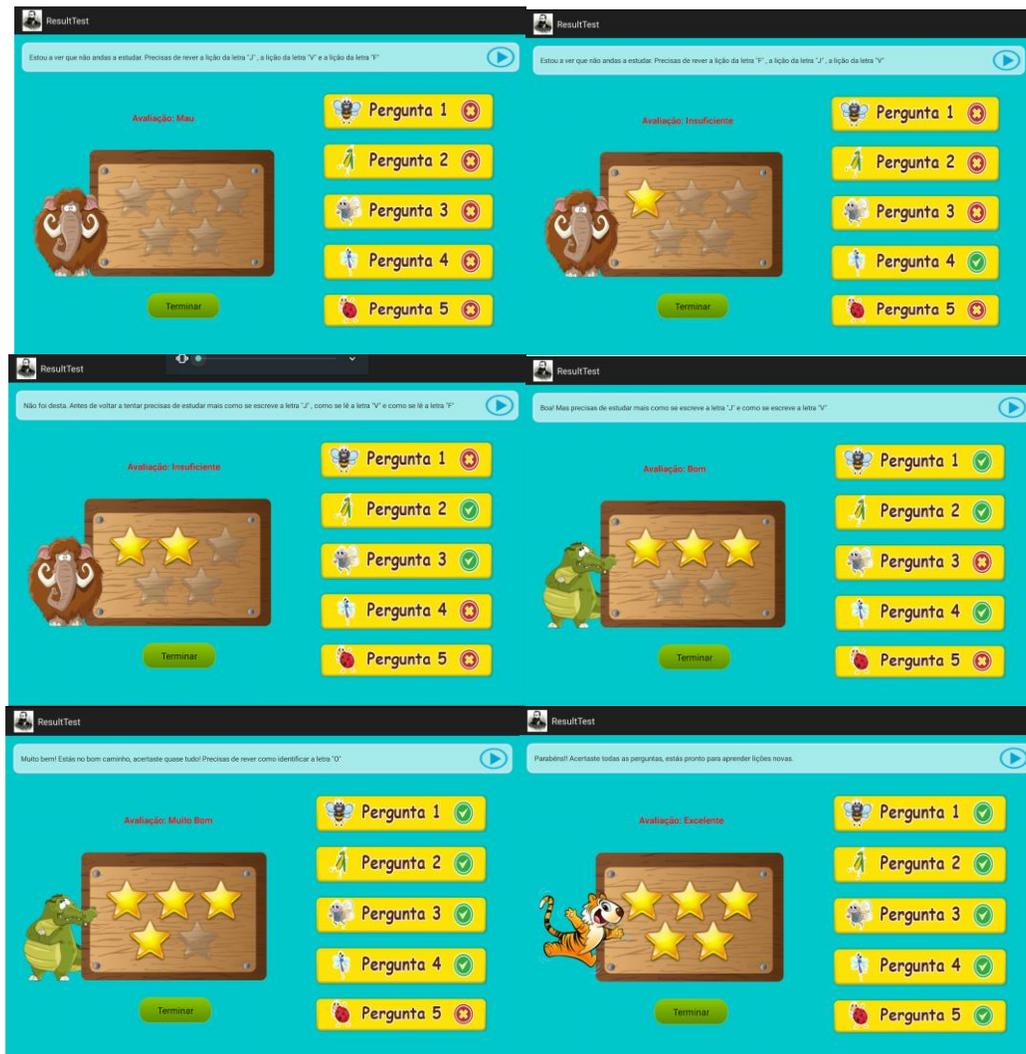


Figura 28: Representação dos diferentes tipos de avaliação que o aluno pode obter

Aquando da apresentação da avaliação que o aluno obteve num dado teste de avaliação, este também pode consultar todas as perguntas que o teste de avaliação inclui, tanto as que errou como as que acertou. As perguntas que o

aluno acertou são assinaladas com o símbolo do “certo”, a verde (figura 29), e as que errou com uma “cruz”, a vermelho (figura 30).



Figura 29: Exemplo de uma pergunta que o aluno acertou



Figura 30: Exemplo de uma pergunta que o aluno errou

Após a fase de validação e teste do sistema de avaliação incorporado, achámos que seria importante o aluno poder ver novamente as perguntas do teste que realizou e verificar aquilo que errou ou acertou. Para que isso fosse possível, foi implementado uma funcionalidade que permite ao aluno clicar na pergunta que quer consultar para se poder lembrar de qual era a pergunta apresentada. Na figura 31, podemos ver um exemplo em que o aluno decidiu consultar qual era a pergunta 1 do teste e, como podemos observar, a imagem que mostrava a classificação por estrelas é substituída por um modelo mais pequeno da pergunta em questão. Se o aluno quiser ver novamente a classificação pode fazê-lo através do botão “Classificação” que fica disponível quando o aluno decide consultar uma, ou mais, perguntas do teste. Se o aluno quiser terminar de visualizar a avaliação do teste, basta clicar no botão “Terminar” que é direcionado para o menu inicial da aplicação.

ResultTest

Boa! Mas precisas de estudar mais como se escreve a letra "J" e como se escreve a letra "V"

Avaliação: Bom

Qual destas desenhos representa a letra "J"? Escolhe uma das opções.

« A letra "J" »

Classificação Terminar

- Pergunta 1  
- Pergunta 2  
- Pergunta 3  
- Pergunta 4  
- Pergunta 5  

Figura 31: Demonstração da consulta de uma pergunta

Capítulo 5

Conclusões e Trabalho Futuro

5.1. Considerações, Contributos e Limitações

Nesta dissertação abordámos a problemática da aplicação de um ITS numa instituição de ensino básico, dando particular atenção à incorporação de um modelo de ensino para a aprendizagem da Língua Portuguesa utilizando a Cartilha de João de Deus. Durante o desenvolvimento do trabalho desta dissertação, as metodologias seguidas para a obtenção do perfil e o método de avaliação escolhido seguiram todas as fases de desenvolvimento consideradas apropriadas, nomeadamente o pré-processamento de dados, a observação do algoritmo de mineração de dados e o consequente estudo dos padrões gerados para a identificação do perfil. A fase de investigação e estudo dos diferentes algoritmos permitiu-nos adquirir o conhecimento para identificar as três etapas principais do trabalho a realizar. O processo de mineração de dados pode ser feito através de várias técnicas e métodos. Contudo, nesta dissertação, descrevemos aqueles que nos pareceram ser os mais utilizados para a identificação de um perfil de um utilizador de um sistema de ensino inteligente. Apesar das várias técnicas analisadas, optámos por desenvolver uma técnica alternativa baseada em regras base de profiling especificamente orientada para a identificação de perfis. O protótipo do modelo de avaliação desenvolvido até ao momento ainda tem muito por onde evoluir. Porém, através do modelo de domínio desenvolvido, acreditamos ter as bases suficientes para a progressão do protótipo do modelo de avaliação desenvolvido. O passo natural seguinte para a melhoria do sistema desenvolvido passaria por uma fase de testes num ambiente real de sala de aula e com a utilização da aplicação por parte dos alunos, de forma a conseguirmos observar uma reação aos resultados obtidos e com isto progredir nas etapas seguintes do modelo de avaliação do modelo de Kirkpatrick.

Em termos gerais, podemos dizer que o trabalho realizado no âmbito desta dissertação acrescentou novo valor ao sistema que inicialmente recebemos

para incorporar um sistema de avaliação, particularmente nos seguintes aspetos:

- melhorou o ITS de forma a que este pudesse ser adaptado para um acompanhamento mais personalizado, eficaz e apropriada a cada aluno e ao seu próprio ritmo de aprendizagem;
- concebeu e integrou na estrutura base do sistema uma técnica alternativa para a construção de um teste de avaliação mais personalizado.

Ao longo do desenvolvimento desta dissertação fomos identificando algumas limitações que é importante realçar. As limitações encontradas estão, maioritariamente, relacionadas com questões funcionais, nomeadamente, com o desempenho das funcionalidades que agora foram desenvolvidas e integradas no módulo de avaliação. A utilização de maiores volumes de informação é algo que deveria ter sido realizado para podermos ter uma melhor perceção da eficiência do sistema. Para que a implementação do modelo de avaliação tivesse outro impacto, era fundamental que este tivesse sido testado num caso real. Além disso, a inevitabilidade de ter que existir uma ligação à internet para a execução de alguns serviços, nomeadamente o reconhecimento de voz, obriga à utilização do sistema em modo on-line. Por fim, importa também realçar que, por muito que estes sistemas tenham o objetivo de simular o comportamento humano aquando da interação do aluno, os diálogos acabam por ser bastante rígidos e sem qualquer tipo de entoação. É, também, importante mencionar que o uso da tecnologia tem as suas desvantagens e deve ser controlado, visto que o seu uso excessivo pode levar a situações de isolamento e cansaço visual, entre outros. É importante que não deixe de existir interação entre os alunos e entre os alunos e professores.

5.2. Trabalho Futuro

Como trabalho futuro, parece-nos essencial indicar algumas medidas que seria conveniente aplicar. Uma vez que a identificação de um perfil de utilização é um processo algo complexo, a inclusão do algoritmo desenvolvido num contexto real de sala de aula seria o ideal, de forma a conseguirmos perceber quais as suas fragilidades e, posteriormente, corrigi-las e melhorá-las. Seria necessário perceber se o sistema de identificação de um perfil de

utilização desenvolvido se comporta como deveria, ou seja, percebermos se realmente tem a capacidade perante um caso real de conseguir perceber quais as maiores dificuldades e facilidades de um aluno aquando da utilização completa do sistema. Outra medida importante, seria a capacidade de resposta do sistema perante a utilização de um maior conjunto de dados, não só com diferentes utilizadores, mas também com um maior conjunto de lições. Seria essencial perceber qual a capacidade do sistema de gerar testes diferentes, sempre consoante a condição base, maiores dificuldades e facilidades do aluno, mas perceber se o sistema conseguiria continuamente criar testes diferentes e englobar a matéria toda de forma a não se tornar simplesmente um processo monótono e repetitivo e assim conseguirmos estar constantemente a apresentar ao aluno novos desafios para o conseguirmos manter motivado nesta diferente forma de aprendizagem.

Outro aspeto a considerar, é identificar e perceber se a escolha do método de avaliação desenvolvido é realmente o mais indicado para o caso de estudo. De forma a percebermos isto, e tal como já referido relativamente à identificação de um perfil, o ideal seria aplicar o método desenvolvido num contexto real de sala de aula e com isto perceber primeiro, se é realmente a forma mais indicada de o fazer e, segundo, conseguirmos observar qual a reação aos resultados obtidos (nível 1 do método de avaliação de Kirkpatrick) para com isto progredir nas etapas seguintes do método de avaliação do modelo de Kirkpatrick. Para isto, era também necessário utilizar um maior conjunto de dados, pois se as conclusões fossem positivas a uma primeira aplicação do nível 1 do modelo de avaliação de Kirkpatrick passaríamos aos níveis seguintes. Sendo que no nível seguinte, aprendizagem, tentaríamos perceber, qual o nível de aprendizagem de um aluno, ou seja, perceber se realmente houve uma melhoria no conhecimento adquirido do estudante. Para isto teríamos que comparar os resultados iniciais dos primeiros testes do aluno com os resultados dos testes mais recentes e perceber o comportamento do mesmo. Posteriormente, no nível três, transferência, tentaríamos perceber se o conhecimento adquirido pelo aluno é estável, ou seja, se quando realiza as lições já não comete sistematicamente os mesmos erros que cometia numa fase inicial do estudo e se consegue transferir esse conhecimento aos seus colegas de turma. Por fim, no nível quatro e último deste método, tentaríamos perceber se o aluno consegue ser um aluno produtivo, se retém o conhecimento adquirido e se tem a capacidade de continuar a adquirir novos

conhecimentos. Após a aplicação deste método de avaliação num caso real era importante conseguir perceber se o aluno tem a capacidade de perceber e interpretar a matéria tão bem quanto, ou melhor do que, se a aprendizagem fosse feita pelos métodos tradicionais para, com isto, conseguirmos entender a capacidade e utilidade do sistema desenvolvido, e se realmente atinge o que é esperado de um ITS.

Um outro aspeto a melhorar e, já mencionado, é a capacidade de interação do sistema com o aluno. O facto de não ter qualquer entoação na voz reproduzida pelo sistema poder-se-á tornar monótono para o aluno que também poderá não ter a melhor perceção se o que está a fazer e a forma como está a interagir com o sistema é a mais correta. Tendo em conta o estudo que foi realizado em relação a motores de voz para o sistema *Android*, o motor atual implementado pareceu-nos o mais atual e indicado, querendo com isto dizer, que com a tecnologia atual destes motores parece-nos uma limitação do sistema, não querendo com isto dizer que não poderá ser algo a ser melhorado futuramente com o desenvolvimento e melhoria destes motores.

Outro aspeto que poderia melhorar no sistema seria a forma como atualmente o sistema reconhece a voz do aluno, nos exercícios que o requerem. De forma a conseguirmos um sistema que conseguisse funcionar em pleno em modo *offline* o ideal seria a criação de um sistema próprio semelhante e que fosse uma característica que fizesse parte da aplicação. Para isto era necessário analisar os prós e os contras de tal implementação, porque enquanto que por um lado, provavelmente, deixaríamos de necessitar de uma constante ligação à internet, por outro, a aplicação poderia vir a tornar-se mais lenta e, com isso, perder capacidade de resposta.

Uma outra limitação do atual sistema é o facto de toda a base de dados ser local, o que faz com que um aluno tenha que utilizar sempre o mesmo dispositivo para que o sistema funcione corretamente, caso o aluno faça login num outro dispositivo senão o que utilizou previamente fará com que o sistema interprete como sendo um novo utilizador e recomeça o processo de identificação do perfil do atual utilizador. Uma forma de contornar esta situação seria a criação de uma base de dados num servidor e que quando o utilizador fizesse a ligação à internet, o sistema fizesse a sincronização da base de dados local com a do servidor. Isto possibilitaria a que um utilizador

pudesse fazer login a partir de qualquer dispositivo e que os seus dados fossem os mais atualizados.

Por último, um aspeto igualmente importante seria entender qual o interesse dos professores na utilização deste sistema na sala de aula, e perceber o que os professores esperariam de um sistema como este e quais as sugestões pelos mesmos de forma a conseguirmos chegar a um sistema que agradasse às duas partes. Para isto, sugeriria que se fizesse um estudo, na forma de inquérito, com um número reduzido de professores para perceber qual o nível de interesse e, se houvesse, por onde é que seria sugerido melhorar o sistema até agora desenvolvido.

Nesta altura, quando estamos a encerrar mais um capítulo no desenvolvimento desta ferramenta, gostaríamos de terminar esta dissertação dizendo que acreditamos que este sistema trará vantagens acrescidas no processo de ensino individual e orientado a cada aluno no contexto de sala de aula em processos de ensino que sigam a metodologia da Cartilha Maternal de João de Deus.

Referências

- (s.d.). Obtido de <http://www.cs.bris.ac.uk/~kovacs/publications/gbml-survey/html-version/shadow-decision-tree-3mux.png>
- (s.d.). Obtido de <http://personales.unican.es/gutierjm/main/ai.html>
- Allinger, G., & Janak, E. (1989). Kirkpatrick's levels of training criteria: Thirty years later.
- Arnold, E. (2003). *Mass High Tech*. Obtido de How can businesses measure the impact of training? *Mass High Tech: The Journal of New England Technology*: <http://masshightech.bizjournals.com/masshightech/stories/2003/12/01/focus6.html>
- Arthur, K., Gage-Little, R., & Munson, D. (2002). *Dalemunson*. Obtido de Training Kirkpatrick's four levels of evaluation: <http://www.dalemunson.com/products/Kirkpatrick%20Presentation.pps>
- Baskin, C., Barker, M., & Woods, P. (2005). When group work leaves the classroom does group skills development also go out the window? *British Journal of Educational Technology*, 36, 19-31.
- Bates, R. (2004). *A critical analysis of evaluation practice: The Kirkpatrick model and the principle of beneficence*. (Vol. 27). Evaluation and Program Planning.
- Beal, C., Beck, J., & Woolf, B. (1998). Impact of intelligent computer instruction on girls math self concept and beliefs in the value of math. *American Educational Research Association*.
- Birnbrauer, H. (1987). *Evaluation techniques that work*. Training and Development.
- Boyle, M., & Crosby, R. (1997). Academic program evaluation: Lessons from business and industry. *Journal of Industrial Teacher Education*, 81-85.

- Cbrwiki. (23 de Setembro de 2012). *Cbrwiki*. Obtido de http://cbrwiki.fdi.ucm.es/mediawiki/index.php/Main_Page
- Cedefop. (2002).
- Deus, G. d. (2008). *Associação de Jardins Escolas João de Deus*. Obtido de Associação de Jardins Escolas João de Deus: <http://www.joaodeus.com/associacao/detalhe.asp?id=6>
- Fitzpatrick, J., Sanders, J., & Worthen, B. (2004). *Program evaluation: Alternative approaches and practical guidelines* (3^a ed.). Boston, Pearson.
- Freedman, R. (. (2000). What is an intelligent tutoring system? Em *Intelligence*.
- Gertner, A., Conati, C., & and VanLehn, K. (1998). Procedural Help in Andes. Em A. Gertner, C. Conati, & K. and VanLehn, *Generating Hints Using a Bayesian Network Student Model*. *Artificial Intelligence*.
- Ghaoui, C. (2004). *E-Education Applications: Human Factors and Innovative Approaches*. Liverpool, England.
- Haulmark, M. (2002). *Accommodating cultural differences in a web-based distance education course: a case study*. In *Proceedings of Ninth Annual International Distance Education Conference*. (C. f. Reserch, Produtor) Obtido de World Wide Web.
- Hildebrandt, M., & Gutwirth, S. (. (2008). Profiling the European Citizen. Cross Disciplinary Perspectives. Em *Springer, Dordrecht*.
- Hofsted, G. (1986). Cultural differences in teaching and learning. *International Journal of Intercultural Relations*, 10, 301-320.
- Hosch, W. L. (2016). *Britannica*. Obtido de Britannica: <https://www.britannica.com/technology/machine-learning>
- Keles, A., Oca, R., & Gulcu, A. (2009). ZOSMAT: Web-based Intelligent Tutoring System for Teaching-Learning Process. Em *Expert Systems with Applications* (pp. 1229-1239).
- Kinshuk. (1996). *Computer aided learning for entry level Accountancy students*. Montford University, England: PhD Thesis.

- Kolodner, J. L. (1992). An introduction to case-based reasoning. Em *Artificial Intelligence Review*,
- Koper, R., & Tattersall, C. (2005). *Learning Design: A Handbook on Modelling and Delivering Networked Education and Training*. Berlin: Springer.
- Kovalik, C., & Dalton, D. (1998). The process/outcome evaluation model: a conceptual framework for assessment. *Journal of Education Technology Systems*, 27, 183-194.
- Kruse, K. (n.d.). *Beyond Kirkpatrick*. Obtido de <http://www.e-learningguru.com/articles/art51.html>
- Learnnovators. (27 de Junho de 2013). *Learnnovators*. Obtido de Learnnovators: <http://learnnovators.com/blog/kirkpatrick-fours-levels-of-evaluation/>
- McNamara, C. (2000). *Mapnp*. Obtido de The field guide to nonprofit program design, marketing, and evaluation.: http://www.mapnp.org/library/evaluation/fnt_eval.html
- Mika, K. (1996). *Program outcome evaluation: A step by step handbook*. Milwaukee: Families International.
- Nickols, F. (2004). A stakeholder approach to evaluation training. (<http://www.nickols.us>, Ed.)
- Nkambou, R. M. (2010). Advances in intelligent tutoring systems. Em *Heidelberg: Springer*.
- Nkambou, R. M. (2010). Advances in intelligent tutoring systems. Em *Heidelberg: Springer*.
- NUA. (2002a). *How many online?* Obtido de World Wide Web: http://www.nua.ie/surveys/how_many_online/index.html
- Nwana, H. S. (1990). Intelligent tutoring systems: An overview. Em *Artificial Intelligence Review*.
- Posavac, E., & Carey, R. (2007). *Program evaluation: Methods and case studies* (7^a ed.). Upper Saddle River, NJ: Prentice Hall.

- Priest, S. (2001). A program evaluation primer. *Journal of Experimental Education, 1*, 34-40.
- Reigeluth, C. (1999). *A new paradigm of instructional theory*. . London: Lawrence Erlbaum.
- Resumekiduit.dvrlists.com. (2016). Obtido de Resumekiduit.dvrlists.com: <http://resumekiduit.dvrlists.com/neural-networks.html>
- Riotto, J. (2004). Model for calculating ROI of training/learning initiatives. *Journal of Interactive Instruction Development, 18-21*.
- Sakthiyavathi, K., & Palanivel, K. (2009). A generic architecture for agent based E-learning system. *Intelligent Agent & Multi-Agent Systems*.
- Schalock, R. (2001). Outcome-based evaluation. *New York*. Kluwer Academic/Plenum Publisher.
- Schiaffino, S., Garcia, P., & Armandi, A. (2008). eTeacher: Providing personalized assistance to e-learning students. Em *Computers & Educational* (p. 51).
- Schumann, P. L., P.H. Anderson, T. S., & Lawton, L. (2001). *A framework for evaluating simulations as educational tools. Developments in Business Simulation and Experiential Learning*.
- Scott, G. (2006). *Accessing the Student Voice: Using CEQuery to identify what retains students and promotes engagement in productive learning in Australian higher education*. . Canberra: Department of Education, Science and Training: .
- Simon, P. (2013). *Too Big to Ignore: The Bussiness Case for Big Data*.
- Smith, A. (1997). *Human Computer Factors: A Study of Users and Information Systems*. Maidenhead, UK: McGraw Hill.
- Stats, I. L. (2016a). *Internet Live Stats*. Obtido de World Wide Web: <http://www.internetlivestats.com/internet-users/>
- Stats, I. L. (2016b). Obtido de World Wide Web: <http://www.internetlivestats.com/internet-users/>

- Stufflebeam, D. (2003). *Wmich*. Obtido de The CIPP model for evaluation.:
<http://www.wmich.edu/evalctr/pubs/CIPP-ModelOregon>
- Usama Fayyad., G. P.-S. (1996). From Data Mining to Knowledge Discovery in Databases. *AI Magazine*, pp. 37-54.
- Webmaster. (2002). *Informatik*. Obtido de Informatik:
<http://www.informatik.uni-bremen.de/gdpa/gdpa06.htm>
- Webmaster. (01 de Janeiro de 2002). *Informatik*. Obtido de Informatik:
<http://www.informatik.uni-bremen.de/gdpa/gdpa06.htm>
- Wikipedia. (30 de Agosto de 2016). *Wikipedia*. Obtido de
https://en.wikipedia.org/wiki/Cluster_analysis
- Wikipédia, a. e. (21 de Junho de 2016). *Wikipédia*. Obtido de Wikipédia:
https://pt.wikipedia.org/wiki/Cartilha_Maternal
- Wilson, M., Sahay, S., & Calhoun, C. (2014). *Aritzhaupt*. Obtido de aritzhaupt.com:
http://www.aritzhaupt.com/eBook_ADDIE/evaluation.html