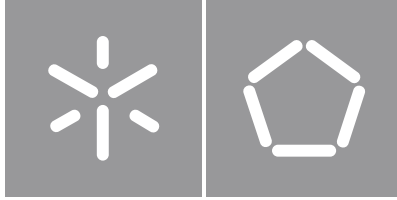


Universidade do Minho

Escola de Engenharia

Mariana de Oliveira Pereira

**Avaliação Automática de Testes de
Atenção e de Acuidade Visual**



Universidade do Minho

Escola de Engenharia

Mariana de Oliveira Pereira

**Avaliação Automática de Testes de
Atenção e Acuidade Visual**

Dissertação de Mestrado

Mestrado Integrado em Engenharia Informática

Trabalho efetuado sob a orientação do(a)

Professor Doutor Pedro Rangel Henriques

Professora Doutora Maria João Pereira

AUTHOR COPYRIGHTS AND TERMS OF USAGE BY THIRD PARTIES

This is an academic work which can be utilized by third parties given that the rules and good practices internationally accepted, regarding author copyrights and related copyrights.

Therefore, the present work can be utilized according to the terms provided in the license bellow.

If the user needs permission to use the work in conditions not foreseen by the licensing indicated, the user should contact the author, through the RepositóriUM of University of Minho.

License provided to the users of this work



Attribution-NonCommercial

CC BY-NC

<https://creativecommons.org/licenses/by-nc/4.0/>

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity.

I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the University of Minho.

Mariana de Oliveira Pereira

AGRADECIMENTOS

Em primeiro lugar, gostaria de agradecer aos meus orientadores, Professor Doutor Pedro Rangel Henriques e Professora Doutora Maria João Pereira pelo apoio e disponibilidade durante a realização deste projeto. Foram essenciais para a conclusão do mesmo.

Em segundo, agradecer a Rafaela Pinho do Pinto, pela prestabilidade e por disponibilizar a dissertação, que serviu de base para esta exposição.

A análise de resultados não seria possível sem a cooperação de algumas pessoas. Gostaria de reservar um agradecimento especial à professora Lurdes Oliveira, por, enquanto intermediária, ter tornado possível a realização da análise experimental no *Agrupamento de Escolas Camilo Castelo Branco*. Gostaria de reconhecer também o contributo do *Centro Neurosensorial de Braga* no levantamento de requisitos e testes preliminares da plataforma.

Em seguida, deixar uma nota de carinho especial à minha família, por todo o apoio e suporte que me deram ao longo desta caminhada. Sem eles não teria conseguido chegar aqui.

Agradecer também aos amigos, pelas muitas memórias da qual eles fizeram parte durante este percurso, em especial à Helena Martins e Maria Dias. ...

E por último agradecer ao meu querido Rui, que durante este caminho nunca me deixou desistir e me motivou sempre a continuar para a finalização deste ciclo. Sem o teu apoio e suporte não seria possível.

Muito obrigada.

ABSTRACT

In recent years, our research group on Language Processing, GEPL, has been collaborating with *Centro Neurosensorial de Braga*, led by Dr. Ana Paula Azevedo.

In this context, some serious games were developed and installed for recognizing shapes, emotions and training central and peripheral vision. They are used in memory therapy, deconcentration, dyslexia, and other problems that affect the acquisition of knowledge in learning processes. The ideas that rose up along the literature review done on those areas, will be exposed along the state-of-the-art chapter in this report.

This thesis proposes a system that will implement an error detection algorithm based on speech-to-text analysis to check whether the spoken sequence contains errors or not. As the system is intended to be installed in the Neurosensory Center, the results will be presented visually to help the therapist in their day-to-day work and monitor the actual use of the system.

Keywords: Serious Games, Speech Recognition

RESUMO

Nos últimos anos, o nosso grupo de investigação em Processamento de Linguagens, GEPL, tem vindo a colaborar com o *Centro Neurosensorial de Braga*, liderado pela Dr^a Ana Paula Azevedo.

Neste contexto, foram desenvolvidos e instalados alguns jogos sérios para reconhecimento de formas e de emoções e para treino da visão central e periférica. Estes são usados na terapia da memória, desconcentração, dislexia e outras perturbações que afetam a aquisição de conhecimentos em processos de aprendizagem. As ideias que surgiram, no decorrer da revisão, de literatura feita sobre essas áreas serão expostas ao longo do capítulo de estado da arte deste relatório.

Esta tese propõe um sistema que implementará um algoritmo de deteção de erros baseado na análise de fala para texto de modo verificar se a sequência falada contém erros ou não. Como o sistema se destina a ser instalado no Centro Neurosensorial, os resultados serão apresentados visualmente para auxiliar o terapeuta no seu dia-a-dia e monitorar o real uso do sistema.

Palavras-Chave: Jogos Sérios, Reconhecimento de Fala

CONTEÚDO

1	INTRODUÇÃO	1
1.1	Motivação	2
1.2	Objetivo	2
1.3	Hipótese de pesquisa	3
1.4	Metodologia de Pesquisa	3
1.5	Estrutura do Documento	4
2	CONCEITOS ESSENCIAIS	5
2.1	Jogos Sérios	5
2.1.1	Jogos Sérios na saúde e educação	5
2.2	Reconhecimento de fala	6
2.2.1	Arquitetura de um sistema de reconhecimento de fala	6
2.2.2	Ferramentas Reconhecimento de Fala	7
2.2.3	Reconhecimento de fala e dislexia	8
3	DAISY, ARQUITETURA DO SISTEMA	9
3.1	Arquitetura do Sistema	9
3.1.1	Realização de um Teste	10
3.2	Algoritmo de Correção	12
3.2.1	Tipo de Erros	13
3.2.2	Algoritmo de detecção	14
4	DAISY, IMPLEMENTAÇÃO DO SISTEMA	16
4.1	Implementação do Sistema	16
4.1.1	Modelo de Base de Dados	17
4.1.2	Servidor <i>Web</i>	18
4.2	Plataforma Final	19
4.2.1	Pacientes	19
4.2.2	Testes	20
4.2.3	Realização de Testes	22
5	ANÁLISE DE RESULTADOS DE TESTES	26
5.1	Análise de resultados	27
5.1.1	Precisão do algoritmo	27
5.1.2	Tipos de erros	29
5.1.3	Provas mal classificadas	29
5.1.4	Erros de omissão	30

5.1.5	Tipos de falha	30
5.1.6	Vocabulário utilizado	31
5.1.7	Conclusão	32
6	CONCLUSÃO	33
6.1	Trabalho futuro	33

LISTA DE FIGURAS

Figura 1	Arquitetura de Reconhecimento de Fala	6
Figura 2	Arquitetura do sistema	10
Figura 3	Diagrama de Sequência	11
Figura 4	Daisy: esquema físico e tecnologias usadas	17
Figura 5	Esquema da Base de Dados	18
Figura 6	Página com a lista dos pacientes	20
Figura 7	Página de adicionar um paciente	20
Figura 8	Página com informação de um paciente específico	21
Figura 9	Página com a lista dos testes	21
Figura 10	Página de adicionar um teste	22
Figura 11	Página com informação de um teste específico	22
Figura 12	Página de escolha do teste a realizar	23
Figura 13	Aviso de informação quando inicia o teste	23
Figura 14	Página de realização de teste	24
Figura 15	Página de escolha do teste a realizar	24
Figura 16	Página de apresentação de resultados da correção	25
Figura 17	Sexo dos alunos	27
Figura 18	Idade dos alunos	27
Figura 19	Gráfico com a percentagem de acerto do sistema	28
Figura 20	Exemplo de omissão	28
Figura 21	Contabilização dos erros e omissões detetadas	29
Figura 22	Erro bem identificado, de erro mal identificado e de falso positivo	30
Figura 23	Percentagem de omissões detetadas e não detetadas	31
Figura 24	Percentagem de falha na forma, cor ou ambos	31
Figura 25	Percentagem de uso vocabulário correto e de vocabulário diferente	32

ACRONYMS

A

ADHD Attention-deficit hyperactivity disorder.

B

BANC Bateria de Avaliação Neuropsicológica de Coimbra.

H

HMMS Hidden Markov models.

INTRODUÇÃO

Na sequência da colaboração com o *Centro Neurosensorial de Braga*, surgiu a necessidade de criar uma avaliação automática de jogos sérios. O Centro concentra-se no tratamento de atividades sensoriais neurológicas que são fundamentais para o desenvolvimento de capacidades de linguagem, leitura e escrita para uma ampla gama de idades, de crianças a idosos.

Este projeto de mestrado tem como foco o uso de jogos sérios na área de transtornos do neurodesenvolvimento. Os transtornos do neurodesenvolvimento são um grupo de condições que ocorrem tipicamente nos estágios iniciais do desenvolvimento e são caracterizados por deficits de desenvolvimento, variando de limitações muito específicas de aprendizagem ou habilidades comunicativas a deficiências globais de interação social ou aptidões intelectuais (Ahn, 2016). Os distúrbios comuns do neurodesenvolvimento incluem:

- Transtorno de deficit de atenção e hiperatividade (TDAH), definido por uma persistente falta de concentração ou alta impulsividade de hiperatividade que interfere no funcionamento ou desenvolvimento de um indivíduo (Ahn, 2016).
- Dislexia, que dificulta a aquisição das capacidades de leitura e escrita apesar de não estar relacionada às habilidades cognitivas da pessoa em questão (Gaggi et al., 2014).
- Memória de curto prazo e de trabalho, que envolve o esquecimento de informações às quais o indivíduo foi exposto recentemente. Este distúrbio pode ter diferentes tipos dependendo da causa subjacente da alteração da memória (Casella and Khalili, 2019).

Os jogos sérios são usados na saúde, educação e indústria com o objetivo de resolver um problema por meio de simulação. No que diz respeito à educação, o uso deste tipo de jogos na terapia de pacientes com problemas de aprendizagem vem desempenhando um papel cada vez mais importante, pois cativa as crianças por serem sistemas interativos (Gaggi et al., 2014).

1.1 MOTIVAÇÃO

Uma parte importante dos jogos sérios é a sua correção, pois permite observar a evolução dos pacientes ao longo de seu tratamento. O problema está no facto de ainda serem corrigidos manualmente pelos terapeutas, o que pode ser um processo exigente e demorado. Assim, criar uma solução automática para a correção de jogos sérios permitiria maior eficiência na deteção de erros cometidos, agilizaria e facilitaria o trabalho dos responsáveis por essas terapias. Com isso em mente, este projeto de mestrado tem como objetivo a criação de um sistema para Avaliação automática de testes de atenção e de acuidade visual. Sendo que o algoritmo deste sistema pretende corrigir um jogo baseado em uma matriz de cores ou formas e cores, utilizado para o diagnóstico de dislexia em indivíduos com idades entre 5 e 7 anos e 7 e 15 anos. Neste jogo, o paciente deve ler cada linha sequencialmente, ou seja, da esquerda para a direita e de cima para baixo. O objetivo do algoritmo seria detetar erros como a omissão, permuta, ou repetição de um elemento da matriz (Albuquerque and Simões, 2009).

A plataforma web criada para o sistema desenvolvido, inclui reconhecimento de fala. Esta ferramenta tem sido amplamente utilizada nesse tipo de terapia devido à sua praticidade para todos os envolvidos. Embora a última década tenha visto uma evolução crescente do reconhecimento de voz em adultos, este continua a ser um processo muito complexo. Algumas das dificuldades enfrentadas são a linguagem corporal, o ruído ao redor, as ambiguidades da linguagem. (Forsberg, 2003; Booth et al., 2020).

Além dos tópicos mencionados acima, o reconhecimento de voz em crianças representa uma dificuldade adicional. Após uma breve experiência no *Centro Neurosensorial de Braga* para testar este tipo de sistema, foi possível observar que os mesmos resultados podem não ser obtidos quando utilizados em crianças. Isso pode dever-se ao facto de que as vozes das crianças ainda estão em fase de desenvolvimento, têm uma dicção mais fraca e sofrem constantes mutações à medida que crescem (Booth et al., 2020). Nesta experiência também se observou que os pacientes podem usar palavras que nem sempre são sintaticamente as mais corretas, palavras fora do vocabulário pretendido para a realização da prova, apesar de estarem semanticamente de acordo com o pretendido.

1.2 OBJETIVO

Os principais objetivos são:

- Categorização de diferentes desafios terapêuticos ou jogos
- Escrita do algoritmo para correção automática de jogos selecionados
- Desenvolvimento da plataforma web do sistema

- Implementação do reconhecimento automático de fala para crianças

E como objetivos secundários:

- Caracterizar os diferentes desafios ou jogos terapêuticos.

1.3 HIPÓTESE DE PESQUISA

Uma aplicação web baseada em algoritmo de reconhecimento de fala treinada em vozes de crianças permitiria a correção automática do jogo escolhido. Além disso, a correção automática seria uma contribuição relevante para o trabalho dos terapeutas na realização desses testes. A criação de uma base de conhecimento para cada paciente permitirá também que os terapeutas consigam acompanhar mais facilmente a evolução do paciente ao longo do tratamento.

H1: É possível automatizar a correção de jogos sérios que utilizam reconhecimento de fala.

H2: É possível otimizar o processo de correção no caso de jogadores infantis.

Este documento descreve uma abordagem e a arquitetura de um sistema que se destina a apoiar a prova das hipóteses acima mencionadas.

1.4 METODOLOGIA DE PESQUISA

Para o desenvolvimento da dissertação, a metodologia adotada seguirá os seguintes passos:

- Pesquisar a bibliografia de jogos sérios e reconhecimento de voz
- Análise e resumo da bibliografia selecionada
- Testar e treinar API de reconhecimento de voz com crianças
- Definir algoritmo de correção
- Criar a plataforma web para o sistema implementado
- Análise dos resultados obtidos
- Testar o sistema desenvolvido

Sempre que foi necessário após os testes, repetiram-se as fases anteriores.

1.5 ESTRUTURA DO DOCUMENTO

O documento está estruturado em quatro capítulos. O primeiro capítulo aborda a motivação para a realização desta tese, bem como os seus objetivos e metodologias de trabalho. O segundo apresenta o estado da arte dos jogos sérios, correção automática e reconhecimento de voz. No terceiro, é descrito arquitetura do sistema implementado assim, *Daisy* e algoritmo de implementação. No quarto, é descrito implementação do sistema e demonstrado o resultado final do sistema. O quinto capítulo apresenta a análise de resultados dos testes realizados. E por último, o quinto sexto expõe a conclusão do trabalho desenvolvido bem como o trabalho futuro.

CONCEITOS ESSENCIAIS

Para cumprir os objetivos propostos, foi necessário pesquisar as áreas de jogos sérios na saúde e educação, bem como na área de reconhecimento de voz. Este capítulo serve para relatar os conceitos básicos e o estado da arte sobre esses dois tópicos.

2.1 JOGOS SÉRIOS

Como mencionado anteriormente, os jogos sérios têm uma ampla gama de aplicações, como saúde, educação, indústria e militar. Eles permitem simular situações da vida real e ajudam a treinar reações, tomar decisões rápidas ou melhorar e aprender certas habilidades.

Os jogos sérios consistem em realizar uma tarefa, por exemplo educativa, embora esta não seja considerada uma tarefa no verdadeiro sentido da palavra, pois está disfarçada em um jogo. Por ser mais interativo, interessante e viciante para quem o faz, os pacientes ficam motivados a fazê-lo, facilitando assim o trabalho das terapias educativas. Por norma, trata-se de terapias monótonas que não motivam as pessoas que as realizam, dificultando o tratamento. Ainda assim, os jogos sérios não são para fins de entretenimento, mas para fins educacionais.

2.1.1 *Jogos Sérios na saúde e educação*

Atualmente, os videogames não são apenas uma forma de entretenimento, mas também podem ser didáticos. Eles ajudam a motivar as pessoas durante o processo de aprendizagem, pois o nível sucessivo de progressão traz satisfação e bem-estar contínuos. Além disso, resulta em um aumento em seus conhecimentos gerais, que podem ser aplicados na vida real (Janarthanan, 2012).

Na área da saúde, os jogos sérios também têm um papel importante e podem ser aplicados na área de controle da dor ou treino. O controle da dor é uma aplicação inovadora desses jogos. A distração causada pelo jogo afeta a percepção da dor que o paciente sente quando sofre uma lesão ou doença (Janarthanan, 2012; Gaggi et al., 2014).

Como esses jogos simulam cenários reais, eles possibilitam que as pessoas treinem e se preparem para esses mesmos cenários (Janarthanan, 2012; Gaggi et al., 2014).

Em seguida, alguns exemplos de jogos sérios são descritos:

- *PlayWithEyes*, realizar triagem de acuidade visual de crianças muito pequenas (Bortoli and Gaggi, 2011).
- *DYSL-X*, detecção precoce do risco de dislexia em pré-escolares (Audenaeren et al., 2013).

No *Centro Neurosensorial de Braga* algumas das provas apresentadas são retiradas do BANC, incluindo as provas de nomeação rápida (psy).

2.2 RECONHECIMENTO DE FALA

Um sistema de reconhecimento fala converte uma sequência de palavras faladas em uma sequência de palavras escritas. Esse tipo de sistema possui diversas aplicações que facilitam a realização das tarefas do dia a dia.

A maioria desses sistemas modernos geralmente é baseada em modelos estatísticos, como modelos ocultos de Markov (HMMs). Uma razão pela qual os HMMs são populares é que seus parâmetros podem ser estimados automaticamente a partir de uma grande quantidade de dados, e são simples e computacionalmente viáveis.

2.2.1 Arquitetura de um sistema de reconhecimento de fala

Um sistema de reconhecimento de voz normalmente consiste em quatro componentes: um front-end acústico, um modelo acústico, léxico, modelo de linguagem e um decodificador, conforme mostra na Figura 1.

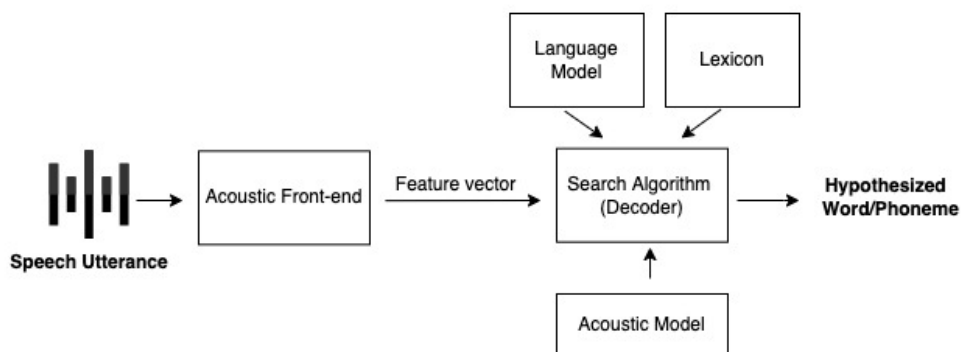


Figura 1: Arquitetura de Reconhecimento de Fala

A princípio, o front-end acústico converte o sinal de áudio em um conjunto apropriado de recursos. O sinal de entrada do microfone é convertido em uma sequência vetorial em um processo conhecido como extração de recursos.

Em seguida, os parâmetros do modelo são gerados usando os vetores obtidos anteriormente no processo de extração de características durante o processo de treino.

O decodificador visa explorar o espaço de busca para encontrar a sequência de palavras com maior probabilidade de corresponder à sequência de entrada. O reconhecimento de voz extrai um certo número de parâmetros de uma unidade de palavra ou sub-palavra. Esses parâmetros descrevem a palavra ou sub-palavra modelando a variação do sinal ao longo do tempo para criar um padrão que caracteriza a palavra ou sub-palavra.

Os padrões de palavras são armazenados e posteriormente usados como base de comparação para selecionar a melhor correspondência para uma determinada palavra. Essa técnica é frequentemente chamada de reconhecimento de padrões (Karpagavalli and Chandra, 2016).

2.2.2 Ferramentas Reconhecimento de Fala

Como anteriormente mencionado, o reconhecimento de fala é um problema muito complexo devido ao número de fatores envolvidos. Atualmente, os reconhecedores comerciais de última geração têm resultados próximos do ideal, mas apresentam algumas limitações que não permitem resolver totalmente o problema.

A tabela a seguir mostra uma comparação de três sistemas comerciais: Amazon, Google e Microsoft.

	Google Speech-to-Text	Microsoft	Amazon Transcribe
Idiomas & Variantes	125+	100+	30+
Suporte SDK	C#, Go, Java, Node.js, PHP, Python, Ruby	C#, C++, Go, Java, JavaScript, Swift, Python	.NET, Go, Java, JavaScript, PHP, Python, Ruby
Vocabulário Personalizado	Sim	Sim	Sim
Pagasse	Sim	Sim	Sim

Tabela 1: Comparação entre três sistemas comerciais de reconhecimento de voz (goo; Eric-Urban; Russo, 2003)

Conforme mostra na Tabela 1, os sistemas são bastante semelhantes nas funcionalidades que oferecem. Uma das principais preocupações na solução proposta é o suporte para o português, por ser a língua materna do paciente, sendo que os três sistemas mencionados suportam. Os sistemas comerciais de última geração vêm com um preço com limite de uso mensal. Como se destina a realizar várias avaliações experimentais e o sistema do Microsoft

oferece aos alunos um plano Estudante, sem custos, esse é o que vai ser usado ao longo deste projeto.

Durante esta pesquisa, não encontramos informações se estes sistemas são projetados para a voz das crianças ou não, o que é um fator importante, pois o foco dos pacientes serão as crianças.

2.2.3 *Reconhecimento de fala e dislexia*

Hoje em dia, o mundo digital está tão enraizado na nossa sociedade que ser excluído dele é uma clara desvantagem. As pessoas com dislexia podem sentir essa exclusão devido às dificuldades de leitura e escrita com as quais nasceram.

A inclusão de determinadas tecnologias no mundo da web auxilia pessoas com dislexia e outros transtornos de aprendizagem. Uma das tecnologias com poder para ajudar a lidar com essa condição é o reconhecimento de voz, pois permite a comunicação escrita em tempo real, eliminando o erro ortográfico. (Rauschenberger et al., 2019).

DAISY, ARQUITETURA DO SISTEMA

Este capítulo descreve etapas realizadas durante o processo de design do sistema *Daisy*. Começando por descrever a arquitetura do sistema, em mais detalhe a realização de um teste e por fim o desenvolvimento do algoritmo de correção.

3.1 ARQUITETURA DO SISTEMA

A plataforma de gestão e correção de jogos sérios foi desenvolvida em colaboração com os terapeutas do Centro Neurosensorial de Braga, pensada com o intuito de otimizar o trabalho dos terapeutas. Tendo isso em consideração, juntamente com os objetivos delineados no capítulo introdutório desta dissertação, os objetivos principais a atingir com o desenvolvimento da plataforma são:

- Disponibilização da plataforma como uma aplicação *web*, suportando todos os navegadores comumente utilizados, bem como ser responsivos e acessíveis de vários dispositivos.
- Inserção dos dados relevantes a cada paciente, bem como a sua gestão e visualização do histórico dos testes realizados pelos mesmos.
- Extensibilidade no que diz respeito ao número de testes disponíveis, ou seja, permitir a inserção de novos testes para além da visualização dos testes à disposição do terapeuta a cada momento.
- Capacidade de correção das provas realizadas pelo paciente em tempo real, começando pelo reconhecimento de fala, pré-processamento do texto transcrito, correção automática do jogo e apresentação dos resultados.

Tendo estes objetivos em consideração, a Figura 2 representa a arquitetura da plataforma, os seus principais componentes e interações entre os mesmos.

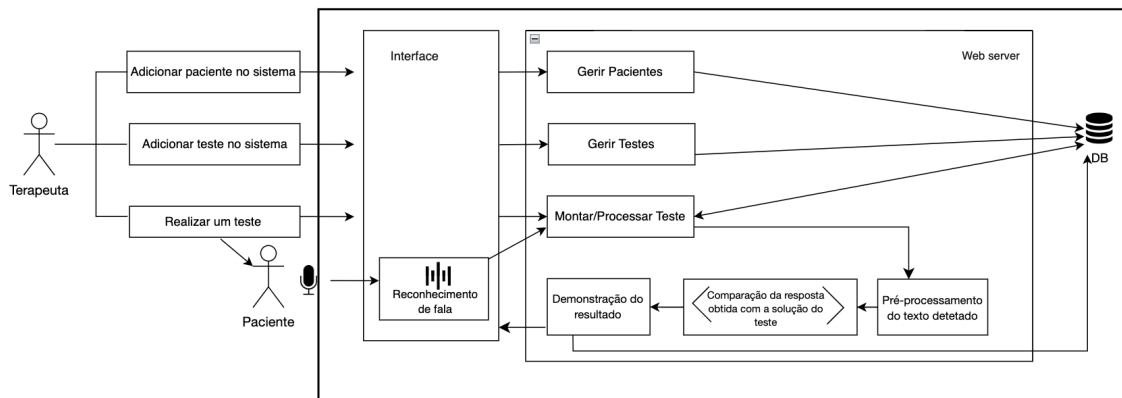


Figura 2: Arquitetura do sistema

O sistema é composto por uma única aplicação *web* que comunica com um servidor *backend* através de um serviço *web* RESTful. Este é composto por três módulos distintos com vista a responder aos objetivos delineados anteriormente:

- Módulo de Gerir Pacientes

Este módulo deve permitir a criação, manipulação e remoção de pacientes para uso futuro na realização de testes. Um paciente caracteriza-se pelas suas informações básicas, bem como os testes realizados até então.

- Módulo de Gerir Testes

Este módulo é responsável pela criação, manipulação e remoção de testes. Um teste é composto por uma imagem representativa do mesmo, observações sobre o seu funcionamento e respetiva solução.

- Módulo da Realização de Testes

Este é o módulo de maior relevância da plataforma, sendo responsável pela transcrição da fala do paciente em texto, o pré-processamento desse mesmo texto, a correção automática do teste e, por fim, a apresentação e associação dos resultados ao paciente.

3.1.1 Realização de um Teste

Pelo facto de o algoritmo de correção ser o principal foco do sistema, esta subsecção descreve em mais detalhe o funcionamento do mesmo, começando por apresentar as etapas da realização e correção do teste. De seguida, é descrito o comportamento do algoritmo desenvolvido.

Tal como representado na Figura 3, a realização e correção de um teste subdivide-se em cinco etapas distintas:

1. Seleção e execução do teste a realizar
2. Reconhecimento de voz
3. Pré-processamento de texto
4. Correção automática do teste
5. Apresentação dos resultados

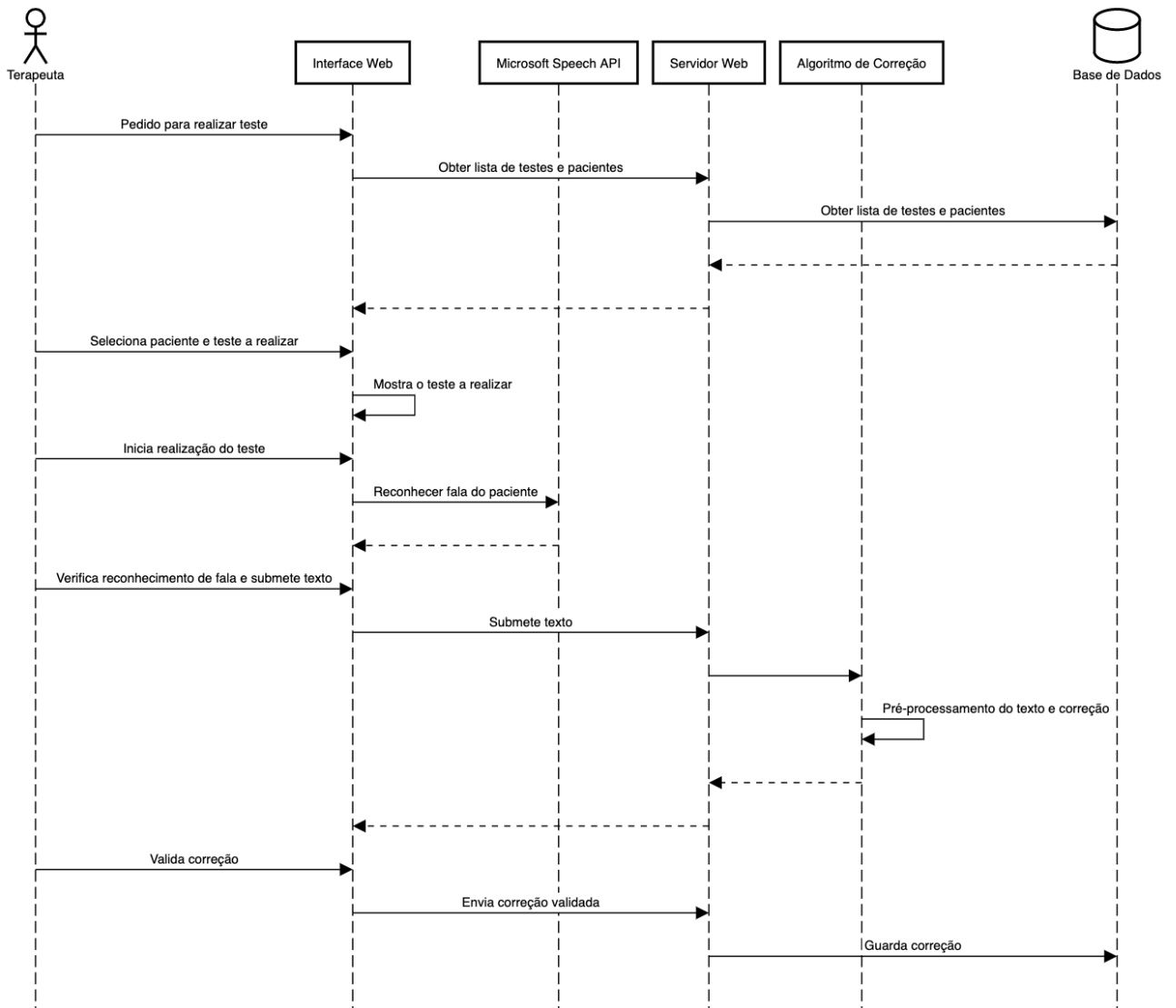


Figura 3: Diagrama de Sequência

Seleção e Execução do Teste a Realizar

Tendo várias opções de escolha, o terapeuta deve selecionar da lista existente o teste que deseja que o paciente realize. Feita a seleção, o jogo aparecerá no ecrã para o paciente realizar.

Reconhecimento de Fala

Enquanto o paciente realiza o teste, o reconhecedor de fala deteta o que está sendo dito e, posteriormente, a fala é transformada em texto para, depois, ser utilizada na correção da prova.

Pré-processamento de Texto

Como os sinónimos e a flexão de género podem interferir no resultado do teste, o vocabulário que deve ser usado durante o teste é apresentado ao paciente antes que ele inicie.

Além disso, os pacientes podem ter alguma hesitação, o que significa que podem dar a resposta errada, mas depois percebem o erro e tentam dar a resposta certa. O problema é que ambas as respostas seriam detetadas pelo reconhecedor de fala.

Diante desses dois problemas, é necessário fazer um pré-processamento para normalizar o vocabulário utilizado na resposta e eliminar a hesitação do paciente nas respostas, assim, o teste será corrigido automaticamente.

Correção Automática do Jogo

Após realizar o pré-processamento, o teste deve ser corrigido automaticamente, utilizando o algoritmo que compara a solução armazenada na base de dados com a resposta do paciente detetando se existe algum erro ou omissões na prova.

Apresentação dos Resultados

No final, o resultado da análise aparecerá no ecrã com detalhes do que foi detetado pelo algoritmo. Para que o terapeuta possa validar essa informação, guardando-a na base de dados. Além disso, informações sobre as sessões anteriores estarão acessíveis ao terapeuta permitindo fazer afirmações sobre a história do paciente e acompanhar a evolução do tratamento.

3.2 ALGORITMO DE CORREÇÃO

Um dos objetivos desta dissertação é a correção da resposta dada pelo paciente tendo por base um algoritmo. O intuito é saber quantos erros este deu durante a realização da prova.

Quando apresentado este problema, já haviam sido detetados certos tipos de erros que a correção tinha de ter em consideração, sendo estes a omissão, repetição, permuta, e falha ao dizer o elemento.

Após uma primeira proposta de algoritmo iniciou-se uma fase de teste. Estes testes consistiram em utilizar a resposta de 141 alunos na prova de Nomeação Rápida como input do algoritmo e analisar a correção dada por este, verificando se os erros detetados eram os esperados. Durante esta verificação foi detetado outro tipo de erro, o elemento extra.

3.2.1 Tipo de Erros

De seguida, são descritos os diversos tipos de erros usando a notação x_0, x_1, \dots, x_n , sendo n o número de elementos da solução e resposta do paciente.

Repetição: O paciente repete um elemento na sua resposta.

- Solução: x_0, x_1, x_2
- Resposta: x_0, x_0, x_1, x_2

Permuta: O paciente troca a posição de dois elementos.

- Solução: x_0, x_1, x_2
- Resposta: x_0, x_2, x_1

Elemento extra: O paciente diz a mais um elemento na sua resposta. O paciente dá uma respostas aleatória que não corresponde nem ao elemento anterior nem ao elemento seguinte da solução.

- Solução: x_0, x_1, x_2
- Resposta: x_0, x_p, x_1, x_2

Falha: O paciente não diz o elemento esperado na sua resposta.

- Solução: x_0, x_1, x_2
- Resposta: x_0, x_p, x_2

Omissão: O paciente omite um elemento da resposta, sendo que pode ser mais do que um seguido.

- Solução: x_0, x_1, x_2
- Resposta: x_0, x_2

3.2.2 Algoritmo de detecção

O algoritmo desenvolvido compara os elementos da solução com a resposta dada pelo paciente. A solução é a lista *sol[]*, cuja variável *i* é usada para a iterar. A resposta é a lista *input[]*, cuja variável *j* é usada para iterar. Estas duas listas são iteradas caso a resposta seja igual, senão o tipo de erro detetado é classificado através das condições de verificação definidas para cada tipo de erro.

Repetição:

```
input[j]==input[j-1] and (input[j+1]==sol[i] or input[j+1]==sol[i+1])
```

A análise da repetição de elementos é feito com base na conjunção de duas condições distintas:

- A primeira condição verifica se ocorre uma repetição de elementos na resposta do paciente.
- A segunda verifica se um determinado elemento na solução e o elemento na resposta do paciente da posição seguinte são equivalentes. Para além disso, como é possível que o paciente faça uma permuta no elemento seguinte verifica-se essa possibilidade com a comparação dos elementos seguintes das duas listas.

Permuta:

```
sol[i]==input[j+1] and sol[i+1]==input[j]
```

A detecção de uma permuta é feita com base na comparação do elemento da solução atual com o elemento seguinte da resposta, e vice-versa.

Omissão:

```
sol[i+1]==input[j] and input[j+1]==sol[i+2]
or flag!= '' and (flag==input[j-1] and input[j]==sol[i+1])
```

Uma vez que a verificação da omissão é feita com a comparação do elemento anterior e existe a possibilidade do paciente errar, acrescentou-se a variável *flag*, que vai conter o elemento que paciente disse erradamente, caso aconteça.

Elemento extra:

```
sol[i-1]==input[j-1] and sol[i]==input[j+1]
```

Compara-se o elementos da solução e da resposta anterior e o elemento atual da solução com o elemento seguinte da resposta.

Falha: Caso não se adeque aos casos anterior é considerado falha.

No final existe a possibilidade de uma das listas não ter sido percorrida até ao fim, havendo ainda assim hipótese de erros não detetados.

No caso de a solução ainda não ter sido toda percorrida, significa então que existem omissões por detetar.

No caso de ser a resposta do paciente que ainda se encontra por percorrer, pode significar que houve repetição do último elemento ou que este disse um elemento extra.

Erros seguidos

Além dos tipos de erros em cima enumerados, o algoritmo também foi concebido para a deteção de erros seguidos, mais concretamente omissões ou falhas seguidas. Nos erros seguidos não são considerados repetição de conjuntos.

```
sol[i+1]!=input[j] and sol[i+1]!=input[j+1] and sol[i+2]!=input[j+2])
```

Para detetar este caso, começa-se por selecionar na resposta dada os primeiros três elementos consecutivos, guardando-os na variável *procurar_seq*, com o intuito de definir a posição onde ocorre essa sequência na solução, tendo sido escolhida uma confiança de três em relação à sequência, ou seja, os três elementos consecutivos. Uma vez que existe a possibilidade de haver mais que uma ocorrência da sequência de procura, *procurar_seq*, é selecionada a primeira posição em que esta é encontrada. No final é verificado se o valor da variável *i* foi alterado, se não tiver sido quer dizer que não houve várias omissões seguidas mas sim erros seguidos.

Pré-Processamento

Antes do algoritmo de correção ser aplicado, é necessário fazer um pré-processamento ao texto detetado pelo reconhecimento de fala, para que este apresente o mesmo formato que a solução da prova, possibilitando assim a aplicação do algoritmo.

O formato da solução da prova implica que o texto esteja em minúsculas, singularidade e a coerência de escrita. Isto é, as palavras devem estar no masculino, os números escritos por extenso e, por último, deve ser usado o vocabulário da prova. Para satisfazer o último ponto, durante os testes acima referidos, foram apontadas as palavras utilizados pelos pacientes referentes ao vocabulário da prova.

Para efetuar a coerência de escrita, foram criados três dicionários: dicionário das formas, das cores e dos números.

Esta implementação foi dividida em duas partes. A primeira parte é referente à transformação do texto com as características acima referidas. A segunda consiste em verificar se as palavras do texto fazem parte do vocabulário de solução e, no caso da prova selecionada ser a das formas geométricas e das cores, verificar se o formato é [‘forma geométrica cor’] em vez de [‘forma geométrica’, ‘cor’].

DAISY, IMPLEMENTAÇÃO DO SISTEMA

Este capítulo descreve o sistema *Daisy*, a plataforma de gestão e correção automática dos jogos sérios desenvolvida ao longo desta tese. A plataforma é composta por uma única aplicação *web*, sendo as suas principais funcionalidades a gestão de pacientes, acompanhado pelo seu histórico de sessões, juntamente com a gestão e correção automática dos jogos sérios utilizados nestas mesmas sessões.

Nesse sentido, são descritas todas as etapas realizadas durante o processo de desenvolvimento da plataforma, desde os requisitos iniciais, os diferentes componentes que compõem a plataforma e a estratégia adotada para os implementar, terminando com uma apresentação do resultado final da mesma.

4.1 IMPLEMENTAÇÃO DO SISTEMA

Depois de estudo e discussão das tecnologias e ferramentas a escolher estas foram as escolhidas, (Figura 4):

- *Python*, uma linguagem de programação utilizada no desenvolvimento do algoritmo de correção de testes. Para além disso, foi utilizada a *framework Flask*, para o desenvolvimento do serviço *web* RESTful do sistema.
- *SQLite*, um sistema de gestão de base de dados relacional para guardar toda a informação necessária ao funcionamento do sistema.
- *React*, uma biblioteca de *JavaScript* utilizada para implementar a interface da aplicação.
- *Microsoft Speech API*, uma *API* desenvolvida pela *Microsoft* que permite o uso de algoritmos de reconhecimento de fala.

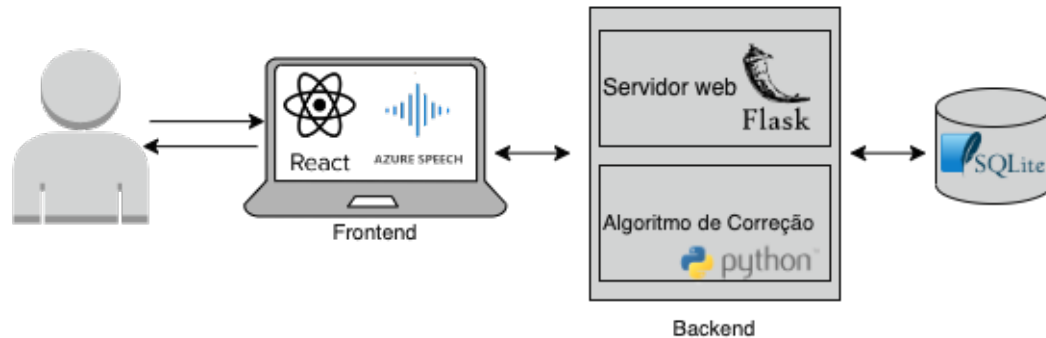


Figura 4: Daisy: esquema físico e tecnologias usadas

4.1.1 Modelo de Base de Dados

Para o correto funcionamento do sistema, é necessário o armazenamento da informação que diz respeito aos pacientes, testes e a realização dos testes. Sendo que o sistema de gestão *SQLite* armazena essa informação sob a forma de tabelas e a forma como se relacionam, foram definidas três tabelas distintas:

- Paciente

Representa toda a informação relativamente ao paciente. Um paciente é caracterizado pelas suas informações básicas, isto é, nome, data de nascimento e sexo, bem como um campo de observações que possam ser relevantes ao seu tratamento.

- Teste

Representa toda a informação relativamente a um teste, sendo descrita pelo nome, imagem e solução, assim como observações que possam ser relevantes à sua realização.

- Realização de Teste

Uma realização de teste corresponde à ligação entre a tabela do paciente e a tabela de teste, uma vez que cada paciente pode resolver quantos testes quiser e cada teste pode ser resolvido por vários pacientes. Contém informações como a sua data de realização, duração, a solução dada pelo paciente, e o tipo de erros dados durante a mesma (falhas, omissões, permutas, repetições e elemento extra).

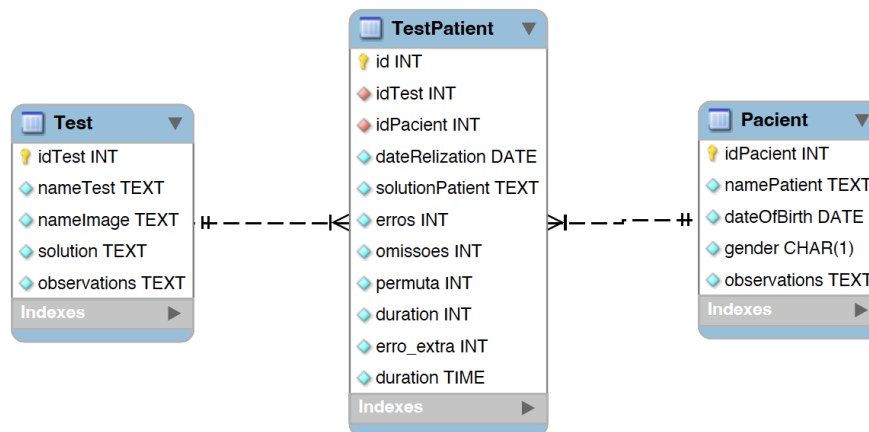


Figura 5: Esquema da Base de Dados

A Figura 5 resume o modelo relacional da base de dados, bem como as relacionamentos entre as três tabelas.

4.1.2 Servidor Web

Para desenvolver o servidor web, foi usado *Flask*, uma *framework* para *Python*, que oferece uma coleção de funcionalidades HTTP, o que facilita a criação de APIs robustas de forma simples e rápida.

Quando uma solicitação é enviada ao servidor, o *Flask* emprega mecanismos de roteamento para servir ao cliente o conteúdo que ele pediu. As rotas permitem mapear pedidos de HTTP para uma ação despoletada do lado do servidor, tendo sido definidas seguintes rotas:

- *POST /tests* -> Adicionar um teste novo na base de dados. Retorna uma string de sucesso ou não.
- *GET /tests* -> Lista os testes inseridos na base de dados até o momento.
- *GET /exams/:id* -> Dado um *id* é listada a informação do teste correspondente.
- *DELETE /exams/:id* -> Dado um *id*, é removida a informação do teste correspondente.
- *POST /patients* -> Adicionar um paciente novo na base de dados. Retorna uma string de sucesso ou não.
- *GET /patients* Lista os pacientes inseridos na base de dados até ao momento
- *GET /patients/:id* -> Dado um *id* é listada a informação do paciente correspondente.

- *DELETE /patients/:id* -> Dado um *id* é removido a informação do paciente correspondente.
- *POST /evaluation* -> Dado um *id* do exame e a resposta do paciente corrige esta.
- *POST /exam* -> Adicionar na base de dados a correção de exame realizado por um paciente.

A aplicação encontra-se hospedada num dos servidores da Universidade do Minho, através de um *container Docker*.

4.2 PLATAFORMA FINAL

Uma vez o sistema implementado, esta secção pretende demonstrar o resultado final desta, apresentado as interface que foram desenvolvidas, dividindo em três partes distintas:

- Gerir Pacientes
- Gerir Testes
- Realizar Teste

4.2.1 Pacientes

A Figura 6 demonstra a página que permite ver todos os pacientes listados no sistema, permitindo a partir dela adicionar mais pacientes e obter informações mais detalhadas sobre os mesmos.

A Figura 7 apresenta o formulário com a informação necessária para adicionar um novo paciente.



Figura 6: Página com a lista dos pacientes

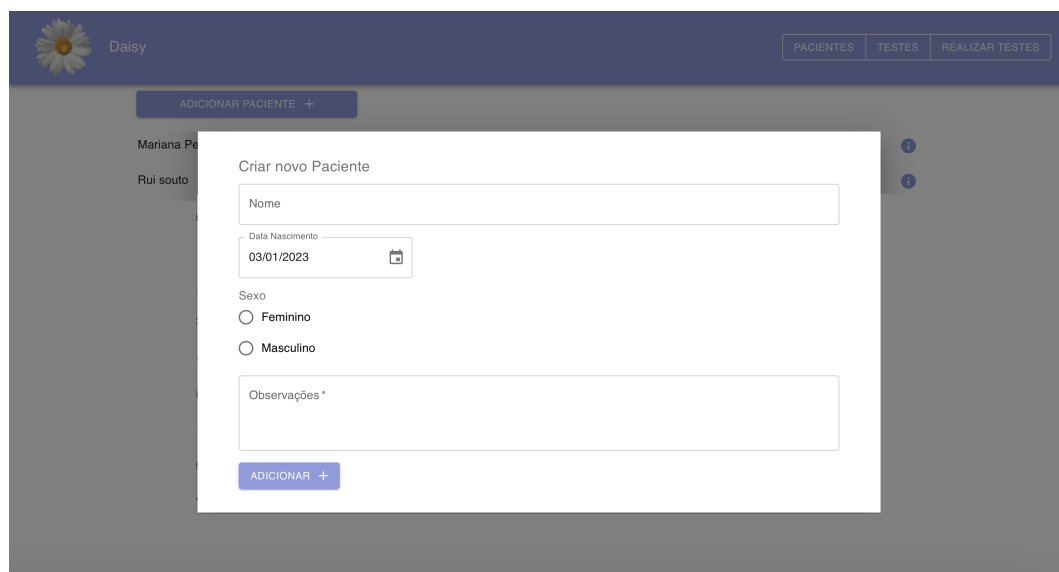


Figura 7: Página de adicionar um paciente

Para cada paciente, é possível ver as suas informações detalhadas assim como o histórico dos testes realizados por este, sendo possível ver essa página na Figura 8.

4.2.2 Testes

Tal como nos pacientes, também existe uma página com a lista dos testes existentes, sendo possível a partir dela adicionar mais testes e obter informações mais detalhadas dos mesmos, como demonstra a Figura 9.

Nome: Mariana Pereira
 Data Nascimento: 1998-12-22
 Sexo: feminino
 Observações:
 Histórico de testes

Nome do Teste	Data de Realização	Duração	Erros	Omissões	Permuta	Repetição	Elemento Extra
Teste das Cores	2022-09-21	0m:2s	0	49	0	0	0
Teste das Cores	2022-09-21	0m:2s	0	48	0	0	0
Teste das Cores	2022-09-21	0m:2s	0	48	0	0	0

Rows per page: 10 1-3 of 3 < >

Figura 8: Página com informação de um paciente específico

ADICIONAR NOVO TESTE +

Teste das Cores

Teste com formas

Figura 9: Página com a lista dos testes

A Figura 10 apresenta o formulário com a informação necessária para adicionar um novo teste.

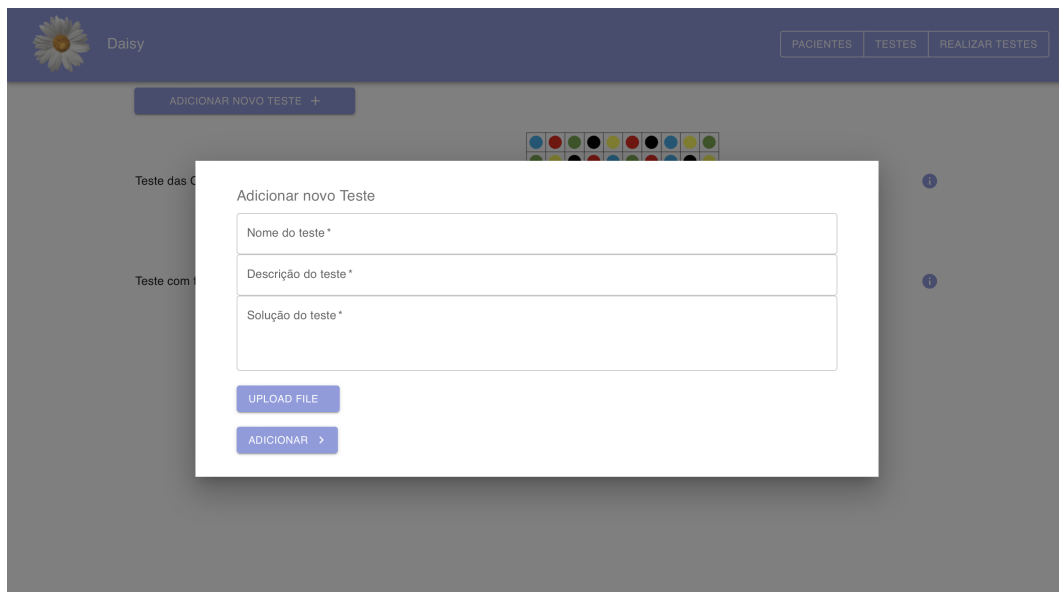


Figura 10: Página de adicionar um teste

Cada teste tem uma página com informação detalhada, sendo possível ver a solução do teste assim como a sua imagem associada. A Figura 11 apresenta essa página.



Figura 11: Página com informação de um teste específico

4.2.3 Realização de Testes

Para realização um teste existe 4 fases distintas.

- Seleção do teste:



Figura 12: Página de escolha do teste a realizar

A Figura 12 apresenta a página de seleção do teste, onde é preciso selecionar o paciente que vai realizar o teste e o teste a realizar.

- Realização do teste:

Após pressionar o botão *COMEÇAR O TESTE* aparece um aviso para informar que quando se carregar no *FECHAR* o microfone irá começar a detetar o que o paciente fala, (Figura 13).



Figura 13: Aviso de informação quando inicia o teste

Na Figura 14 apresenta a página de realização do teste onde aparece em grande a imagem associada a este.

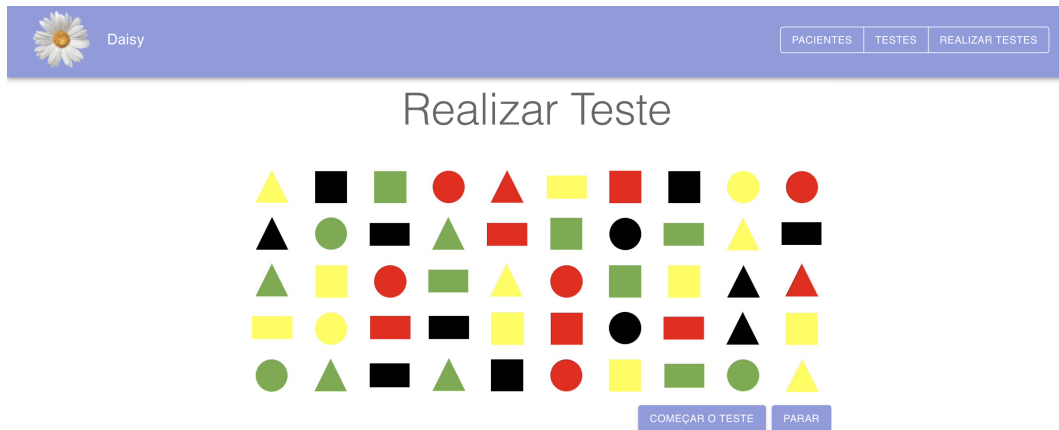


Figura 14: Página de realização de teste

- Validação da resposta:



Figura 15: Página de escolha do teste a realizar

Uma vez terminada a prova, é dada a hipótese do terapeuta corrigir algo que tenha sido mal detetado pelo reconhecedor de fala, tal como demonstrado na Figura 15.

- Correção do teste:

Por ultimo é apresentado uma página com o resultado da correção do teste, identificando a quantidade de cada tipo de erro, bem como a duração da prova.

The screenshot shows a user interface for a test result. At the top, there is a header with a Daisy logo and the name 'Daisy'. On the right, there are three buttons: 'PACIENTES', 'TESTES', and 'REALIZAR TESTES'. The main content area is titled 'Proposta de Solução' and displays a 5x10 grid of various colored shapes (triangles, squares, circles, rectangles). Below the grid, the text 'Resposta do Paciente' is followed by a list of 20 shape names: triângulo amarelo, quadrado preto, quadrado verde, círculo vermelho, triângulo vermelho, retângulo amarelo, quadrado vermelho, quadrado preto, círculo amarelo, círculo vermelho, triângulo preto, círculo verde, retângulo preto, triângulo verde, retângulo vermelho, quadrado verde, círculo preto, retângulo verde, triângulo amarelo, retângulo preto, triângulo verde, quadrado amarelo, círculo vermelho, retângulo verde, triângulo amarelo, círculo vermelho, quadrado verde, quadrado amarelo, triângulo preto, triângulo amarelo.

Below the list, the duration is shown as 'Duração: 0m:5s'. A table summarizes the error counts:

Erros	Omissões	Permutas	Repetições	Elemento Extra
0	0	0	0	0

At the bottom, there are two buttons: 'SUBMETER' and 'VOLTAR'.

Figura 16: Página de apresentação de resultados da correção

ANÁLISE DE RESULTADOS DE TESTES

Este capítulo expõe os resultados experimentais e a análise realizada sobre os mesmos com o intuito de avaliar o comportamento da plataforma. Esta análise tem como objetivo responder às seguintes questões:

- a percentagem de acerto do algoritmo, isto é, a percentagem de testes que foram corretamente classificados pelo algoritmo, classificados de forma incorreta e os invalidados
- a contabilização do tipo de erros nos testes realizados
- a percentagem de elementos que foram classificados como erro não o sendo, isto é, um falso positivo e:
 - dos corretamente classificados, a proporção de erros bem identificados e mal identificados
- a percentagem de omissões detetadas e não detetadas
- no que diz respeito ao tipo de falha, a proporção de erros cometidos no que diz respeito à forma e/ou cor do elemento
- o número de casos em que os alunos utilizam vocabulário diferente do expectável

Com o objetivo de ter um conjunto de dados mais diversificado e ter um conjunto de amostras mais alargados, optou-se por testar a plataforma a partir de casos reais, tendo como base 3 turmas do 3º ano do ensino primário do Agrupamento de Escolas Camilo Castelo Branco em Vila Nova de Famalicão.

A Figura 17 e a Figura 18 representa a distribuição dos alunos de acordo com o seu sexo e idade. Do total de 65 alunos, 36 alunos do sexo masculino e 29 alunos do sexo feminino, com idades entre os 8 e os 9 anos.

Comparativamente aos testes referidos no Capítulo 3, os descritos neste capítulo foram realizados de raiz, ou seja, a plataforma foi testada por completo. Para cada aluno, fez-se a inserção dos dados do aluno na plataforma e, posteriormente, a realização do teste propriamente dito.

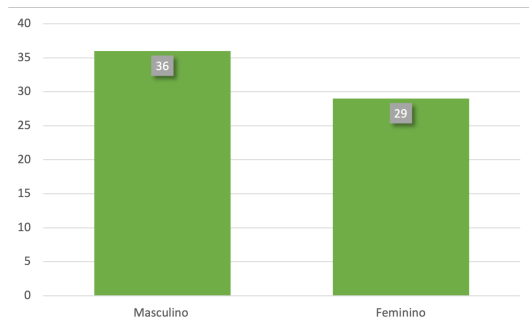


Figura 17: Sexo dos alunos

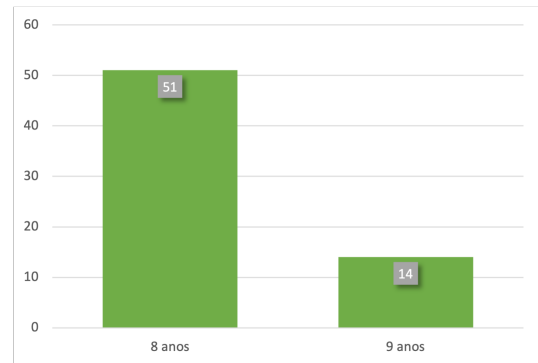


Figura 18: Idade dos alunos

Com o intuito de minimizar o ruído e possíveis perturbações no áudio captado pelo algoritmo de reconhecimento de fala, os testes foram realizados numa sala à parte, estando apenas presente o aluno a realizar a prova de cada vez. Para além disso, recorreu-se a um microfone externo de forma a captar de forma mais clara as palavras das crianças.

Antes de iniciar a prova era apresentado o vocabulário que os alunos deveriam utilizar:

- Números: um, dois, três, quatro, cinco, seis, sete, oito e nove
- Cores: azul, vermelho, amarelo, preto, verde
- Formas: quadrado, retângulo, triângulo, círculo

Tendo em conta que durante a experiência, os alunos usaram um vocabulário que não tinha sido inicialmente considerado (i.e., a forma paralelepípedo) e o reconhecer de fala detetar palavras e escrevê-las de uma forma inesperada como, por exemplo, $\hat{1}(2)$ em vez de quadrado e as palavras sem acento, houve a necessidade de acrescentar mais palavras aos dicionários do algoritmo mencionados no Capítulo 3.

5.1 ANÁLISE DE RESULTADOS

Esta secção descreve o parecer dos resultados propriamente de acordo com as questões mencionadas anteriormente.

5.1.1 Precisão do algoritmo

Numa primeira análise, avaliou-se a precisão do algoritmo de uma forma generalizada, isto é, a percentagem de testes que foram corretamente classificados pelo algoritmo, classificados de forma incorreta e os inválidos. Foi considerado inválido o caso em que durante a realização

da prova o aluno não conseguiu completar o que era pedido, tendo sido depois apontado que este tinha problemas de atenção.

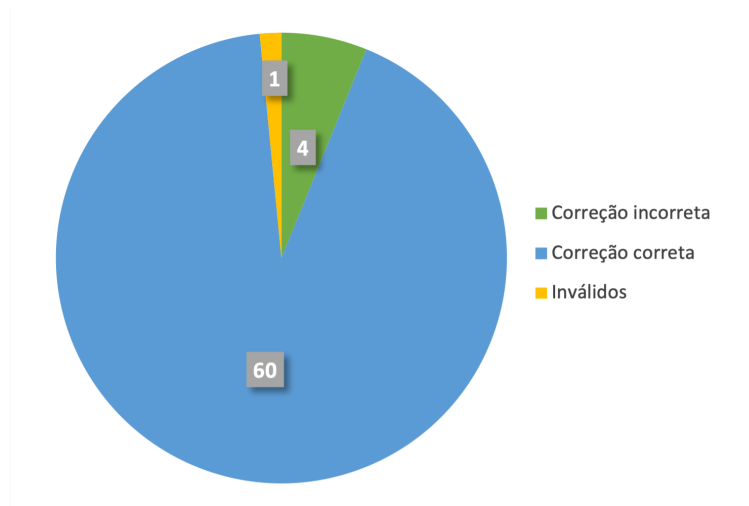


Figura 19: Gráfico com a percentagem de acerto do sistema

A Figura 19 demonstra que das 65 respostas analisadas, em 60 a correção do algoritmo estava correta, isto é, os erros detetados tinham sido efetivamente cometidos pelos pacientes. Tal como mencionado, apenas uma das provas foi considerada inválida.

Das 65 respostas, apenas em 4 a correção do algoritmo estava incorreta, isto é, foram detetados erros a mais ou a menos consoante os casos. De uma forma geral, estes quatro casos podem ser divididos da seguinte forma de acordo com a sua natureza:

- Dois dos casos devem-se ao facto de haver várias omissões seguidas de um erro, fazendo com que os pontos de comparação que existem no algoritmo sejam perdidos pelo número de erros seguidos. A Figura 20 seguinte demonstra um exemplo.

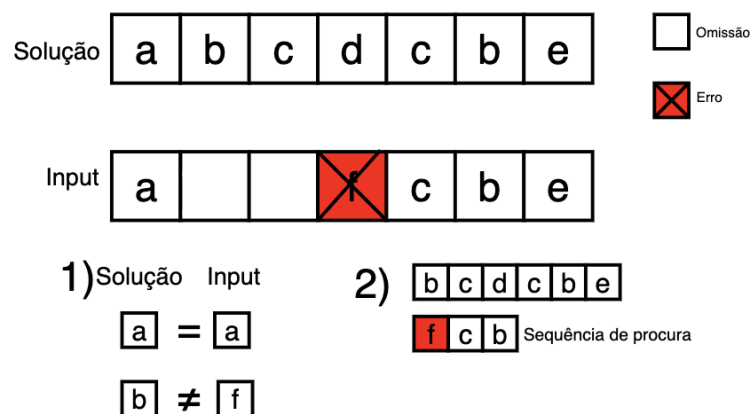


Figura 20: Exemplo de omissão

1. Depois de encontrar o erro, o algoritmo vai percorrer as condições definidas e ver qual é válida para o caso presente.
 2. Ainda que sejam omissões, por serem consecutivas, o algoritmo irá detetar a condição de erros seguidos. No entanto, logo após estas duas omissões, o paciente erra e, como tal, ao procurar a sequência na solução não encontra o elemento *f* que lhe dá início.
- Os restantes ocorrem devido a uma repetição de linha ou conjunto de elementos, fazendo com que o algoritmo perca as suas referências de correção, não conseguindo continuar a sua correção corretamente.

5.1.2 Tipos de erros

A segunda análise incide sobre a contabilização dos erros nos testes que foram realizados.

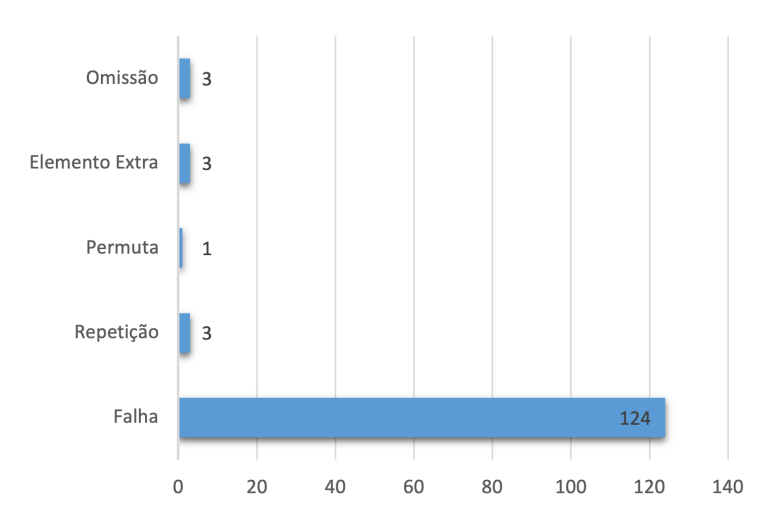


Figura 21: Contabilização dos erros e omissões detetadas

O gráfico representado na Figura 21 demonstra que das 64 provas, foram detetados 135 erros, dos quais 124 são falha, 3 são repetição, 1 é permuta e 3 são elementos extra. É relevante relembrar que se entende como falha quando o paciente se engana na forma, na cor ou em ambos. Por fim, foram detetadas 3 omissões.

5.1.3 Provas mal classificadas

Das provas em que o algoritmo avaliou de forma incorreta a prova, os erros detetados não foram bem classificados ou identificados, tendo ocorrido duas situações distintas:

- Falso positivo: elemento classificado como erro não o sendo.

- Erro mal identificado: o tipo de erro foi mal identificado.

A Figura 22 apresenta o gráfico referente à percentagem de erro bem identificado, mal identificado e de falsos positivos.

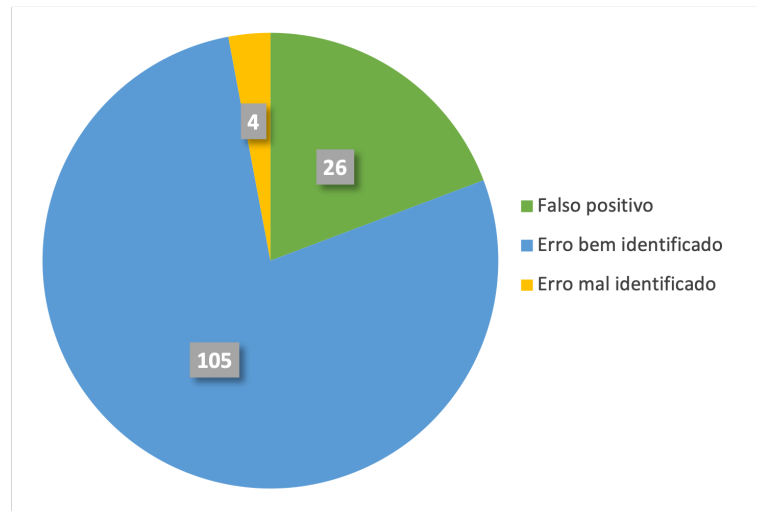


Figura 22: Erro bem identificado, de erro mal identificado e de falso positivo

Como é possível observar, a maior parte dos erros foram bem classificados, 105, 26 foram falsos positivos 4 erros mal classificados.

5.1.4 Erros de omissão

Sendo que uma boa parte dos casos mal classificados se devem ao facto de erros de omissão não terem sido detetados, prossegue-se à análise das omissões detetadas e não detetadas pelo algoritmo.

Nos casos descritos acima, durante a realização dos testes ocorreram 6 omissões mas o sistema detetou apenas 3. O gráfico da Figura 23 apresenta a percentagem das omissões detetadas e das omissões não detetadas.

5.1.5 Tipos de falha

Na Figura 24 os dados analisados são referentes ao tipo de falha, apresentados na Figura 21, verificando assim onde os pacientes mais erram, isto é, na forma, cor ou ambos.

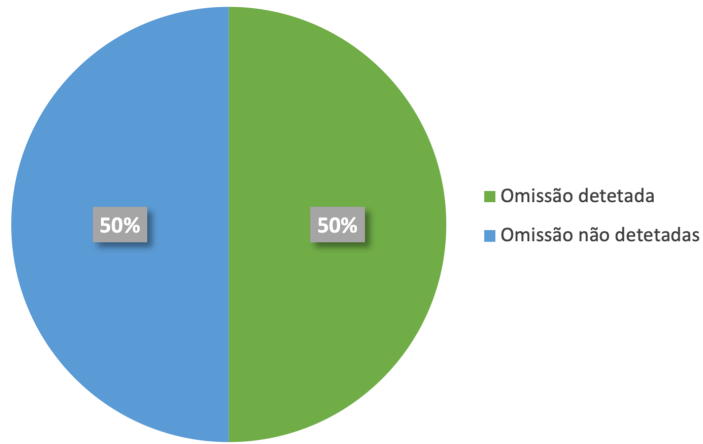


Figura 23: Percentagem de omissões detetadas e não detetadas

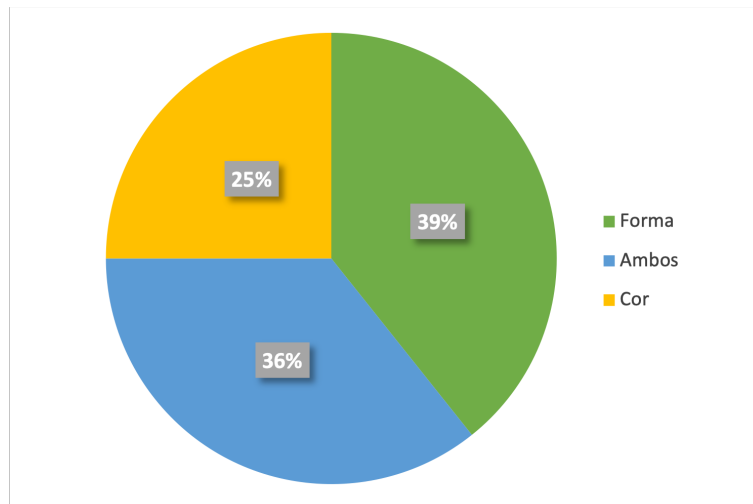


Figura 24: Percentagem de falha na forma, cor ou ambos

Durante a realização da experiência observou-se que quando os alunos erravam na forma do elemento por norma referiam a forma do elemento anterior.

5.1.6 Vocabulário utilizado

Apesar de ser apresentado aos alunos o vocabulário que estes deviam usar nem sempre isso acontecia. Na Figura 25 apresenta o gráfico com a percentagem de uso vocabulário correto e de vocabulário diferente.

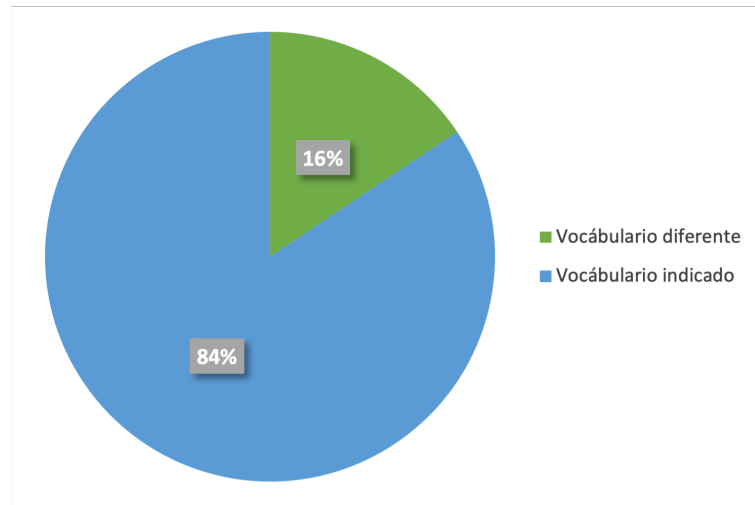


Figura 25: Percentagem de uso vocabulário correto e de vocabulário diferente

5.1.7 Conclusão

Após esta análise, conclui-se que o algoritmo teve uma percentagem de acerto de 94%, sendo que das 64 analisadas apenas 4 obtiveram resultados incorretos.

É importante ainda referir que a plataforma possibilita ao utilizador fazer um pré-processamento manual das respostas do paciente detetadas pelo algoritmo do reconhecimento de fala depois de realizada a prova. Isto permite assim colmatar falhas pontuais e otimizar o resultado do sistema em pacientes onde a dicção está subdesenvolvida, um caso comum em crianças.

CONCLUSÃO

A dissertação aqui descrita propôs a criação de um sistema de correção automática de jogos sérios, com integração de reconhecimento de fala durante a realização dos teste. Este tema surgiu no âmbito da realização da dissertação da Rafaela Pinto de Pinho, que teve como tema *Análise biométrica de comportamentos em jogos sérios* (de Pinho, 2022) também em parceria com o *Centro Neurosensorial de Braga*.

Inicialmente começou-se por definir os objetivos e motivação para o desenvolvimento deste trabalho. No estado da arte, uma análise de referências bibliográficas selecionadas sobre os Jogos Sérios, com foco na área da saúde e educação, e reconhecimento de fala. Em seguida, é apresentada a de arquitetura do sistema bem como o algoritmo de correção implementado. Posteriormente é apresentado a implementação do sistema e o resultado final da plataforma implementada. E por fim, é apresentado a análise de resultados dos testes realizados.

Inicialmente esta dissertação pretendia dar resposta a duas questões: (a) se seria possível automatizar a correção de jogos sérios que utilizam reconhecimento de fala (b) se seria possível otimizar o processo de correção no caso de jogadores infantis. A análise de resultados permitiu concluir que a solução proposta permite otimizar todo o processo de correção de jogos sérios com recurso a reconhecimento de fala, tendo o algoritmo obtido um grau de precisão de 94% nos testes realizados a crianças.

Apesar dos testes realizados terem tido como base apenas um tipo de jogo, a plataforma foi desenhada com o intuito de ser extensível a mais provas de nomeação rápida, podendo os pacientes realizarem qualquer tipo de prova de acordo com o que terapeuta pretende.

A plataforma encontra-se disponível no link <https://daisy.epl.di.uminho.pt>.

6.1 TRABALHO FUTURO

Ainda que tenha sido provada a viabilidade da plataforma no problema que se pretendia colmatar, para que a sua utilização fosse viável num cenário de vida real teriam de ser considerados detalhes adicionais.

Durante o planeamento deste projeto foi apontada a necessidade da plataforma ter um sistema de autenticação, mas a sua implementação foi adiada de forma a que o foco seja no desenvolvimento do algoritmo de correção.

Apesar da elevada percentagem de acerto do algoritmo de correção, existem algumas melhorias que se podem efetuar no mesmo: a correção de repetição de conjuntos de elementos, por exemplo repetição de uma linha, e a correção de omissões seguidas de erro.

No seguimento da extensibilidade do numero de testes disponíveis na plataforma, tem-se em vista a criação de novos dicionários e adição de novas palavras aos mesmos com o intuito de aumentar o universo de palavras.

BIBLIOGRAFIA

- Speech-to-text: Automatic speech recognition nbsp;|nbsp; google cloud. URL <https://cloud.google.com/speech-to-text>.
- Coimbra neuropsychological assessment battery (banc) / bateria de avaliação neuropsicológica de coimbra (banc). URL <https://psyassessmentlab.fpce.uc.pt/banc>.
- Donghyun Ahn. Introduction: Neurodevelopmental disorders. *Hanyang Medical Reviews*, 36: 1, 01 2016. doi: 10.7599/hmr.2016.36.1.1.
- Cristina Albuquerque and Mário Simões. Testes de nomeação rápida: Contributos para a avaliação da linguagem oral. *Análise Psicológica*, 27:65–77, 03 2009. doi: 10.14417/ap.183.
- Lieven Audenaeren, Véronique Celis, Vero Vanden Abeele, Luc Geurts, Jelle Husson, Pol Ghesquière, Jan Wouters, Leen Loyez, and Ann Goeleven. *DYSL-X: Design of a tablet game for early risk detection of dyslexia in preschoolers*, pages 257–266. 01 2013. ISBN 978-3-658-02896-1. doi: 10.1007/978-3-658-02897-8_20.
- Eric Booth, Jake Carns, Casey Kennington, and Nader Rafla. Evaluating and improving child-directed automatic speech recognition. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6340–6345, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL <https://aclanthology.org/2020.lrec-1.778>.
- Alberto Bortoli and Ombretta Gaggi. Playwitheyes: A new way to test children eyes. *2011 IEEE 1st International Conference on Serious Games and Applications for Health, SeGAH 2011*, 11 2011. doi: 10.1109/SeGAH.2011.6165458.
- Marco Cascella and Yasir Khalili. *Short Term Memory Impairment*. 08 2019.
- Rafaela Pinto de Pinho. Biometric analysis of behaviours in serious games. Master’s thesis, Minho University, Braga, Portugal, March 2022. MSc dissertation.
- Eric-Urban. Speech-to-text overview - speech service - azure cognitive services. URL <https://docs.microsoft.com/en-us/azure/cognitive-services/speech-service/speech-to-text>.
- Markus Forsberg. Why is speech recognition difficult?, 2003.

- Ombretta Gaggi, Claudio Palazzi, Matteo Ciman, Giorgia Galiazzo, Sandro Franceschini, Milena Ruffino, Simone Gori, and Andrea Facchetti. Serious games for early identification of developmental dyslexia. *Computers in Entertainment*, 15, 09 2014. doi: 10.1145/2629558.
- Vasudevan Janarthanan. Serious video games: Games for education and health. In *2012 Ninth International Conference on Information Technology - New Generations*, pages 875–878, 2012. doi: 10.1109/ITNG.2012.79.
- S. Karpagavalli and E. Chandra. A review on automatic speech recognition architecture and approaches. *International Journal of Signal Processing, Image Processing and Pattern Recognition*, 9:393–404, 2016.
- Maria Rauschenberger, Ricardo Baeza-Yates, and Luz Rello. *Technologies for Dyslexia*, pages 603–627. 06 2019. ISBN 978-1-4471-7439-4. doi: 10.1007/978-1-4471-7440-0_31.
- A. J. Russo. Pt, 2003. URL <https://aws.amazon.com/pt/transcribe/>.