

UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
CURSO SUPERIOR DE CIÊNCIA DA COMPUTAÇÃO

Augusto Moro

**APLICAÇÃO MOBILE PARA TRIAGEM FONOLÓGICA
INFANTIL.**

Santa Maria, RS
2018

Augusto Moro

APLICAÇÃO MOBILE PARA TRIAGEM FONOLÓGICA INFANTIL.

Trabalho de Conclusão de Curso apresentado ao Curso Superior de Ciência da Computação da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para a obtenção do grau de **Bacharel em Ciência da Computação**

Orientador: Prof. Dr. João Carlos Damasceno Lima

Augusto Moro

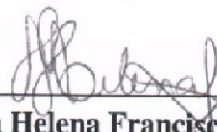
APLICAÇÃO MOBILE PARA TRIAGEM FONOLÓGICA INFANTIL.

Trabalho de Conclusão de Curso apresentado ao Curso Superior de Ciência da Computação da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para a obtenção do grau de **Bacharel em Ciência da Computação**

Aprovado em 04 de Dezembro de 2018:



João Carlos Damasceno Lima, Dr.
(Presidente/Orientador)



Maria Helena Franciscatto, M.^a (UFSM)



Vinícius Maranhão, Dr. (UFSM)

Santa Maria, RS

2018

AGRADECIMENTOS

RESUMO

APLICAÇÃO MOBILE PARA TRIAGEM FONOLÓGICA INFANTIL.

AUTOR: AUGUSTO MORO

ORIENTADOR: JOÃO CARLOS DAMASCENO LIMA

Ferramentas de avaliação fonológica infantil são sistemas que possuem por objetivo detectar e até mesmo recomendar uma reparação para distúrbios fonológicos, porém, tais ferramentas são, em sua maioria, utilizadas por profissionais com formação na área da fonoaudiologia, o que torna seu uso muito restrito à clínicas ou ao meio acadêmico. Além disso, a maioria dessas ferramentas dependem de um profissional com um alto grau de conhecimento para realizar as avaliações fonológicas. Nesse contexto, este trabalho tem por objetivo o desenvolvimento de uma aplicação *mobile* de triagem fonológica infantil automatizada e confiável, que seja capaz de detectar precocemente distúrbios fonológicos, realizando análise de áudio remotamente em um servidor. Além disso, que seja possível ser utilizada por pessoas com um menor grau de conhecimento acerca do assunto. O aplicativo funciona realizando uma conexão com um servidor e enviando os dados para serem analisados, ao final, o aplicativo recebe uma resposta que será apresentada ao usuário. Foram obtidos resultados satisfatórios e validados, uma vez que, os dados que foram utilizados para a obtenção dos resultados, foram avaliados por profissionais fonoaudiólogos, portanto, foi realizada uma comparação entre os resultados. Logo, os resultados obtidos mostraram que a ferramenta pode ser utilizada de forma efetiva no processo de triagem fonológica, apresentando uma alta confiabilidade nos testes realizados os quais foram utilizados dados de pacientes reais.

Palavras-chave: Avaliação Fonológica. Triagem Fonológica. Desvio Fonológico. Aplicação Móvel.

ABSTRACT

MOBILE APPLICATION FOR CHILDREN'S PHONOLOGICAL SCREENING.

AUTHOR: AUGUSTO MORO
ADVISOR: JOÃO CARLOS DAMASCENO LIMA

Children's phonological assessment tools are systems that aim to detect and even recommend a repairment for phonological disorders, however, such tools are mostly used by professionals trained in speech-language pathology, which makes their use considerably restricted to clinics or to the academia. In addition, most of these tools rely on a professional with a high degree of knowledge to perform phonological assessments. In this context, this work aims at the development of an automated and trustworthy phonological phone application, that is able to detect early phonological disorders, performing audio analysis remotely on a server, which is possible to be used by people with a lower degree of knowledge about the subject. The application works by establishing a connection to a server and sending the data to be analyzed, in the end, the application receives the response that will be presented to the user. Satisfactory and validated results were obtained, since the data that were used to obtain the results, were evaluated by speech-language pathologists, therefore, a comparison has been made between the results. Therefore, the results showed that the tool can be used effectively in the phonological screening process, presenting a high reliability of the tests performed which used data from real patients.

Keywords: Phonological Evaluation. Phonological Screening. Phonological Disorder. Mobile Application.

LISTA DE FIGURAS

Figura 2.1 – Imagem da Arquitetura de Schipor	16
Figura 2.2 – Imagem do INFONO	17
Figura 3.1 – Arquitetura do sistema (FRANCISCATTO, 2018)	23
Figura 3.2 – Diagrama de Classes do Sistema	24
Figura 3.3 – DER do sistema	26
Figura 3.4 – Diagrama de Atividades do aplicativo	27
Figura 3.5 – Sistema de sincronização	29
Figura 3.6 – Tela Inicial do Aplicativo	31
Figura 3.7 – Tela de Cadastro de Usuario	32
Figura 3.8 – Tela de listagem dos pacientes	33
Figura 3.9 – Tela de cadastro de paciente	34
Figura 3.10 – Tela de listagem de pacientes	35
Figura 3.11 – Tela de listagem de avaliações	36
Figura 3.12 – Tela de listagem de avaliações	37
Figura 3.13 – Tela de Avaliação	38
Figura 3.14 – Tela de resultado do aplicativo	39
Figura 3.15 – Fluxograma gráfico	40
Figura 3.16 – Fluxograma gráfico	41
Figura 3.17 – Fluxograma gráfico	42
Figura 3.18 – Script de Inserção	43
Figura 3.19 – Programa Java	44
Figura 3.20 – Consultas	45
Figura 3.21 – Resultados da palavra Anel	46
Figura 3.22 – Resultados da palavra Batom	46

LISTA DE ABREVIATURAS E SIGLAS

PET	<i>Phonological Evaluation Tool</i>
SLP	<i>Speech-Language Pathologist</i>
DER	<i>Diagrama Entidade-Relacionamento</i>
UML	<i>Unified Modeling Language</i>
PD	<i>Phonological Disorder</i>
PE	<i>Phonological Evaluation</i>

SUMÁRIO

1 INTRODUÇÃO	9
1.1 CONTEXTUALIZAÇÃO DO TEMA	9
1.2 OBJETIVO GERAL	10
1.3 OBJETIVOS ESPECÍFICOS	10
2 REFERENCIAL TEÓRICO	11
2.1 COMPUTAÇÃO MÓVEL E PERVASIVA	11
2.2 CONSCIÊNCIA DE SITUAÇÃO	12
2.3 DESVIO FONOLÓGICO	13
2.4 AVALIAÇÃO FONOLÓGICA	14
2.5 SISTEMAS DE TERAPIA FONOLÓGICA	15
3 APLICAÇÃO <i>MOBILE</i> PARA TRIAGEM FONOLÓGICA INFANTIL	19
3.1 METODOLOGIA	19
3.1.1 Ferramentas de Modelagem	19
3.1.2 Ferramentas de Desenvolvimento	21
3.1.3 Modelagem e Desenvolvimento	22
3.1.4 Sincronização	28
3.2 CENÁRIO DE AVALIAÇÃO	30
3.3 RESULTADOS	42
4 CONCLUSÃO E TRABALHOS FUTUROS	48
REFERÊNCIAS	50

1 INTRODUÇÃO

A grande popularização e difusão dos dispositivos móveis e sua redução de custo ao longo do tempo, permitiu que a área da computação móvel se desenvolvesse, possibilitando a criação de uma grande variedade de dispositivos e ferramentas. Com esse avanço tecnológico, estão surgindo também, ferramentas que fornecem auxílio à população, não apenas no dia-a-dia, mas também na área da saúde. Com base nisso, a proposta principal deste trabalho é desenvolver um aplicativo para dispositivos móveis voltada para a área da saúde, especialmente para a subárea da fonoaudiologia, utilizando o conceito de consciência de situação. Este conceito é definido por (ENDSLEY, 1995), como "a percepção dos elementos do ambiente dentro de um volume de tempo e espaço, a compreensão do seu significado e projeção de seus efeitos em um futuro próximo".

Este capítulo inicia contextualizando o tema no item 1.1 e apresenta o objetivo geral no item 1.2 e os objetivos específicos no item 1.3.

1.1 CONTEXTUALIZAÇÃO DO TEMA

No domínio de terapia da fala, percebe-se que existem poucas propostas que utilizam modelagem de conhecimento para melhorar tarefas como diagnóstico, planejamento de terapia e intervenção terapêutica (CHUCHUCA-MÉNDEZ et al., 2016). Ferramentas de avaliação fonológica (*Phonological Evaluation Tools* - PET) são ferramentas que visam auxiliar os SLPs na detecção e reparação de distúrbios fonológicos (*Phonological Disorders* - PD), especialmente na população infantil.

Como pode ser constatado em (CERON, 2015) e (O. A. SCHIPOR M. DANUBIANU; NESTOR, 2008), as PETs geralmente fornecem uma sequência de imagens e palavras, onde ambas são relacionadas, ou seja, cada palavra descreve uma figura correspondente. A ideia principal é que o usuário observe a imagem em questão e a elicite a palavra relacionada, enquanto isso, a ferramenta realiza a captura do áudio e o salva em um banco de dados. Posteriormente o fonoaudiólogo realiza a transcrição fonológica da palavra, com base no áudio capturado anteriormente, para que o sistema possa realizar as análises e detectar possíveis PDs.

Ainda existe uma considerável dificuldade para realizar uma avaliação fonológica (*Phonological Evaluation* - PE) atualmente: deslocamento, disponibilidade de profissionais, dispo-

nibilidade de recursos, entre outros fatores que acabam deixando tal assunto, muitas vezes, em segundo plano. Além disso, grande parte dos profissionais realizam o trabalho de forma manual, ou seja, sem o auxílio de uma ferramenta que possa auxiliá-los, o que torna o processo demorado e cansativo. Com base nesses problemas, a proposta do trabalho é desenvolver uma aplicação para dispositivos móveis que seja capaz de realizar uma triagem fonológica preliminar, isto é, que seja capaz de fornecer ao usuário um *feedback*, recomendando ao usuário a necessidade de um acompanhamento fonológico especializado.

1.2 OBJETIVO GERAL

Este trabalho tem por objetivo criar um aplicativo de triagem fonológica infantil, realizando a adaptação e implementação de uma parte do trabalho proposto por (FRANCISCATTO, 2018) em sua dissertação de mestrado. Além disso deve possibilitar a utilização do aplicativo tanto por SLPs como forma de auxílio, como por pessoas sem conhecimento na área.

1.3 OBJETIVOS ESPECÍFICOS

A partir do objetivo geral da seção anterior, o trabalho tem como alvo:

- Realizar uma revisão bibliográfica da literatura, com o objetivo de buscar trabalhos com propósitos semelhantes;
- Analisar ferramentas que podem fornecer algum auxílio no desenvolvimento do aplicativo;
- Desenvolver o aplicativo, com o objetivo de obter um protótipo funcional para a realização de testes;
- Realizar integração do aplicativo com o servidor que foi desenvolvido paralelamente por outro aluno de graduação, sincronizando os dados do aplicativo com o servidor, possibilitando a análise dos dados coletados, utilizando requisições *HTTP*;
- Efetuar testes com o objetivo de obter e validar os resultados, utilizando dados já previamente validados oriundos de um sistema de avaliação fonológico;
- Verificar a possibilidade de trabalhos futuros visando o aprimoramento do aplicativo;

2 REFERENCIAL TEÓRICO

Este capítulo tem como objetivo fornecer o referencial teórico necessário para compreender o trabalho desenvolvido abordando algumas das literaturas relacionadas, as quais abrangem sistemas de suporte à terapia fonológica. Deve também demonstrar alguns trabalhos anteriormente desenvolvidos que estão inseridos no contexto da terapia fonológica.

Assim sendo, a seção 2.1 deste capítulo descreve a computação móvel e pervasiva, a seção 2.2 descreve o conceito de consciência de situação, a seção 2.3 descreve desvio fonológico em si, a seção 2.4 apresenta a definição de PEs segundo alguns autores. Por fim, a última seção explana alguns trabalhos relacionados, além de softwares desenvolvidos para a detecção de PD.

2.1 COMPUTAÇÃO MÓVEL E PERVASIVA

Apresentar uma visão geral sobre a área da computação móvel e pervasiva é de extrema importância para o entendimento deste trabalho. Foi devido ao surgimento de tal área que tornou possível o desenvolvimento de uma série de tecnologias que possibilitou a criação de uma nova gama de *softwares*. Portanto, conhecer um pouco da história sobre a evolução da área é de grande importância para saber onde este trabalho se insere.

O avanço da tecnologia tem causado um impacto crescente na vida diária da população. Conforme a tecnologia avança e se desenvolve, aumenta a exigência por serviços inteligentes que possam ser personalizados, e, além disso, que forneçam os resultados de forma rápida e prática. Tal avanço fez com que a computação virasse seu foco a encontrar formas de integrar sistemas à vida diária da população, ou seja, adaptar sistemas ao cotidiano das pessoas de forma adequada.

A interação de sistemas computacionais com o usuário foi ganhando destaque e maturidade de tal maneira que a indústria voltou-se a formas de computação mais distribuídas e flexíveis: os computadores em grades e isolados mudaram-se para os nossos escritórios pessoais, para os nossos bolsos, roupas e corpos (LYYTINEN; YOO, 2002). Os grandes *mainframes* antes utilizados apenas em universidades e grandes empresas se converteram em computadores pessoais que atualmente são amplamente utilizados por toda a população. Além disso, nos últimos anos, a área da computação móvel vem ganhando um notável destaque. A cada dia sur-

gem diferentes tipos de dispositivos móveis aumentando sua presença na vida cotidiana, desde *smartphones* à dispositivos que podem ser incorporados ao corpo para monitorar dados vitais, entre outros. Essa evolução deu origem à área da computação móvel e pervasiva, que evoluiu de tal forma que levou a computação tradicional a um novo patamar, possibilitando o surgimento de novas tecnologias que podem impactar profundamente a vida da população.

Segundo Araujo (2003), a computação móvel “baseia-se no aumento da nossa capacidade de mover fisicamente serviços computacionais conosco”, ou seja, faz o uso intenso de comunicação sem fio, para que o usuário possa levar consigo serviços para utilizá-los quando desejar independentemente de sua localização, tais serviços que antes estavam disponíveis apenas em grandes máquinas, impossíveis de serem transportadas.

Na busca por automação e adaptação baseada em contexto, surge o conceito de Computação Pervasiva, a qual assume uma série de entidades computacionais ou sensoriais invisíveis, que interagem tanto com o usuário quanto com o ambiente em que operam (YE; DOBSON; MCKEEVER, 2012). Se trata de um modelo computacional totalmente distribuído, definido por aplicações presentes em todo lugar, adicionadas ao ambiente de forma transparente ao usuário.

2.2 CONSCIÊNCIA DE SITUAÇÃO

Situação é um termo que possui vastas definições na literatura. O termo é definido por (YE; DOBSON; MCKEEVER, 2012), como “uma abstração dos eventos que ocorrem no mundo real derivada de contexto e hipóteses sobre como os contextos observados se relacionam a fatores de interesse para projetistas e aplicações”. De forma semelhante, (BETTINI et al., 2010) definem situação como "abstrações semânticas provenientes de pistas contextuais de baixo nível, conhecimento humano e interpretação do mundo".

Por sua vez, Consciência de Situação (SA, ou *Situation Awareness*) é um termo que expressa, segundo Endsley (1995), “a percepção dos elementos do ambiente dentro de um volume de tempo e espaço, a compreensão do seu significado e projeção de seus efeitos em um futuro próximo”. Segundo (FRANCISCATTO, 2018) "se um sistema é consciente de situação, ele tem a capacidade de projetar respostas no ambiente de forma adaptativa, aprimorando a experiência do usuário em determinado meio e durante determinada atividade".

Portanto, o conceito de consciência de situação na área da fonoaudiologia mostra-se bastante eficaz, uma vez que em tal área, é necessário que o sistema seja capaz simular a autonomia humana de tomar decisões levando-se em consideração cada caso em específico.

O aplicativo alvo deste trabalho não usa em seu funcionamento o conceito de consciência de situação. Porém o servidor no qual o aplicativo realiza a comunicação utiliza este conceito para realizar as PEs. Portanto, é interessante conhecer o conceito para entender o contexto em que o sistema como um todo está inserido.

2.3 DESVIO FONOLÓGICO

A contextualização de desvio fonológico é muito importante para entender este trabalho, pois é o problema que este trabalho se propõe a resolver. Portanto, conhecer o problema e suas características é essencial para entender como detectá-lo e conhecer os impactos que tal problema pode causar na população infantil, além de seus danos a médio e a longo prazo.

O desvio fonológico é caracterizado por uma desorganização linguística do inventário de fonemas, sendo identificado por omissões e substituições destes na fala, especialmente consoantes e encontros consonantais (GRUNWELL, 1981; FERRANTE C. BORSEL J. V., 2009). Ou seja, quando um indivíduo possui algum tipo de desvio fonológico, ele pode substituir ou mesmo omitir fonemas de palavras, problema que pode ser corrigido com um acompanhamento fonológico adequado.

O desvio fonológico foi definido por (GRUNWELL, 1981) como uma desorganização, inadaptação ou anormalidade do inventário fonológico da criança em relação ao sistema padrão de sua comunidade linguística. Na classificação do desvio fonológico deve ser levado em consideração a comunidade linguística em que o indivíduo está inserido. Por questões de culturais, pessoas de diferentes regiões podem falar de maneiras diferentes e mesmo assim de forma correta, portanto, a detecção de PDs deve levar a região ou comunidade linguística como base para as análises fonológicas.

Outros autores (WERTZNER H. F., 2005; WERTZNER H. F. PAGAN L. O., 2007) definem o desvio fonológico como sendo uma alteração de fala, caracterizada pela produção inadequada dos sons, de acordo com a idade e as variações regionais, que podem envolver erros na produção, percepção ou organização dos sons.

Outro fator que deve ser levado em consideração durante a detecção de PD é a idade do indivíduo. Por exemplo, indivíduos com idades maiores possuem uma maior capacidade de pronunciar determinados fonemas se comparados com indivíduos com idade menor. Portanto, a escolha dos fonemas adequados com a idade, região e comunidade linguística em que determinado paciente vive é fundamental para uma eficiente detecção de PD.

2.4 AVALIAÇÃO FONOLÓGICA

Segundo (LOWE, 1996) uma PE possui quatro objetivos: (a) determinar se existe uma alteração fonológica; (b) se diagnosticada a alteração, possibilitar orientações e encaminhamentos; (c) proporcionar base para uma intervenção eficaz; e, (d) avaliar o progresso da criança durante o tratamento. Ainda, uma PE deve ser capaz de: (a) oferecer uma descrição dos padrões de produção da fala da criança em sua língua materna; (b) identificar as diferenças entre os padrões típicos da língua do falante e o sujeito que está sendo avaliado; (c) indicar que tipo de implicações comunicativas tem os padrões de fala alterados e examinar as consequências funcionais de sua fala (inteligibilidade) para priorizar no tratamento; (d) oferecer indicações sobre o nível alcançado no desenvolvimento da fala. Assim como, a PE deve dar conta do: (a) inventário fonético; (b) inventário fonológico; (c) distribuição contrastiva dos fonemas; e, (d) estrutura da sílaba (GALCERAN, 2005).

Uma PE deve ser capaz de indentificar se realmente existe alguma alteração fonológica. Portanto, a realização de triagem fonológica antes mesmo de identificar o qual o tipo de problema que pode existir pode ajudar na detecção do desvio fonológico. Logo, pode fornecer uma validação dupla, provendo uma confiabilidade maior nos resultados obtidos. Além disso, deve ser possível acompanhar o paciente durante todo o processo de tratamento, avaliando o progresso do indivíduo ao longo do tempo, além de proporcionar orientações conforme a evolução do tratemnto.

Como foi exposto na seção 2.3, a detecção de PD deve levar em consideração a comunidade linguística na qual o paciente está inserido. Além disso deve indicar o impacto que a alteração fonológica pode causar no paciente a curto e a longo prazo, para determinar quais as prioridades que devem ser levadas em consideração durante o tratamento do paciente.

Com base nisso, este trabalho tem o objetivo de determinar apenas se existe uma alteração fonológica, ou seja, realizar uma triagem fonológica, e efetar uma recomendação ao usuário para determinar se ele necessita de uma atendimento fonológico especializado para concluir as etapas restantes citadas.

Conhecer o funcionamento de uma PE é de grande importância para este trabalho, pois toda a coleta de dados do aplicativo proposto está inserido no conceito de PE. Portanto, fornecer uma interface amigável ao usuário que será avaliado, é de grande importância para o bom funcionamento, além disso, tal interface precisa cumprir seu papel com eficiência. Como ci-

tado anteriormente pelos autores, determinar se existe uma alteração fonológica é o principal objetivo deste trabalho. Coincidentemente, isto está inserido no conceito de PE.

2.5 SISTEMAS DE TERAPIA FONOLÓGICA

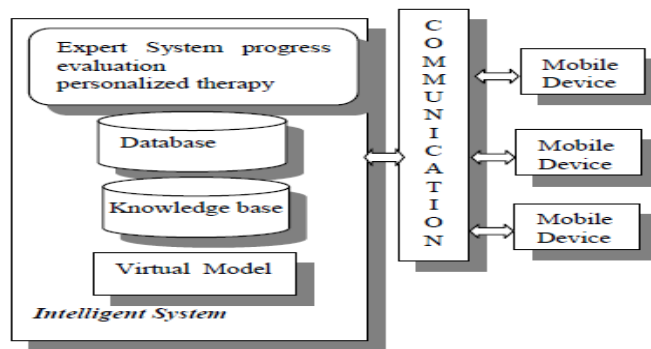
Para determinar o que deve ser projetado e desenvolvido, primeiramente é necessário realizar uma pesquisa dos trabalhos relacionados. Além disso, deve-se analisar cada trabalho, com o objetivo de verificar os pontos os quais a proposta estudada é eficaz, tendo como principal alvo, solucionar eventuais problemas ou pontos fracos encontrados. Também é necessário buscar em trabalhos relacionados, soluções para eventuais problemas que possam surgir durante o desenvolvimento deste trabalho, além de realizar uma comparação deste trabalho com os trabalhos pesquisados.

(ROBLES-BYKBAEV et al., 2016) apresentam um sistema para geração automática de instruções para PE. O trabalho foi proposto pela razão de que, várias vezes, os profissionais fonoaudiólogos (SLPs, ou *Speech-Language Pathologists*) ficam sobrecarregados durante o trabalho. Este problema poderia ser suavizado por com o auxílio de um sistema que possa absorver parte da carga de trabalho. O sistema proposto é baseado em ontologia e possui entidades que permitem realizar a modelagem dos padrões linguísticos obtidos durante as atividades terapêuticas dos pacientes. Assim, o sistema consegue sugerir as atividades ou estratégias que mais se adaptam ao perfil de determinado paciente, levando em consideração suas habilidades, limitações e necessidades. Além disso o sistema utiliza dados clínicos reais de pacientes para realizar sua validação.

(O. A. SCHIPOR M. DANUBIANU; NESTOR, 2008) descrevem uma arquitetura inteligente desenvolvida para atender crianças romenas com idade pré escolar composta por dois componentes: um inteligente instalado no computador de cada clínica fonológica e um componente *mobile* usado como um "terapeuta virtual" onde os dois componentes são conectados. Além disso, foi desenvolvido um sistema para funcionar sobre tal arquitetura, onde é feita uma triagem inicial de cada paciente. Os pacientes que possuem algum desvio fonológico mais grave são submetidos a uma avaliação mais aprofundada com a ideia de determinar o tipo de problema, onde o próprio sistema tenta realizar um diagnóstico. Ao final, um SLP analisa o resultado gerado pelo sistema e pode refutar a resposta do sistema ou aceitá-la. Após feito o diagnóstico, o sistema propõe ao paciente uma série de exercícios específicos voltados a corrigir o problema que foi detectado, ou seja, para cada tipo de problema, existe uma bateria de exercí-

cios. Todo o processo é acompanhado pelo terapeuta, isto é, todos os resultados da terapia são enviados pelo sistema ao terapeuta onde é feita a avaliação da evolução do paciente. A figura 2.1 mostra a arquitetura sobre o qual o sistema foi desenvolvido.

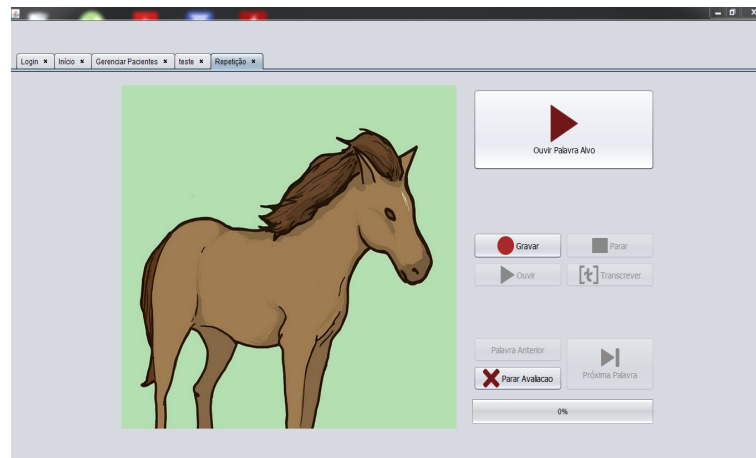
Figura 2.1: Imagem da Arquitetura de Schipor



Fonte: INFONO (2015)

Em CERON (2015) é descrita uma ferramenta de PE voltada para SLPs, denominada *INFONO*. O sistema facilita o trabalho e torna o processo de avaliação menos cansativo e mais atraente para o paciente. O sistema funciona apresentando uma sequência de imagens com suas respectivas palavras correspondentes, onde o usuário deve elicitá-las, o sistema grava o áudio das palavras e os salva no banco de dados. Após a realização da coleta, o SLP analisa os sons que foram salvos e os transcreve foneticamente. Isso torna o processo de PE menos cansativo para o paciente, uma vez que o sistema necessita apenas que o paciente fale as palavras, não necessitando que o SLP realize a transcrição manualmente no momento da avaliação. Ao final, após o SLP transcrever cada áudio que foi salvo anteriormente, o sistema faz uma análise das transcrições. Também são verificadas as substituições e omissões de cada fonema e fornece ao fonoaudiólogo um relatório completo, com todos os fonemas que apresentaram problema. Além disso, apresenta também um diagnóstico, com base nos resultados gerados. O único problema de tal ferramenta é que pode ser usada apenas por SLPs, pois seu uso é bastante complexo, isto é, o usuário da ferramenta necessita de um grau elevado de conhecimento na área da fonoaudiologia para utilizá-la. A figura 2.2 mostra uma tela da PET descrita.

Figura 2.2: Imagem do INFONO



Fonte: INFONO (2015)

(PARNANDI et al., 2015) apresentam um sistema para acompanhamento fonológico remoto pacientes com apraxia de fala¹. O sistema é composto de uma aplicação móvel que realiza a gravação da fala do paciente e em uma interface o terapeuta pode definir uma série de exercícios remotamente permitindo o mesmo acompanhar o desempenho do paciente. Além disso, é realizado um treinamento de modelos acústicos a partir de um conjunto de registros de fala para que o sistema seja capaz de se adaptar à fala do paciente. No trabalho, o fonoaudiólogo tem acesso à um conjunto de relatórios sobre o desempenho do paciente, e pode também fazer adaptações no roteiro de treinamento, caso seja necessário.

Os trabalhos apresentados nesta seção, apresentam sistemas de PE que necessitam de um SLP para realizar o uso das ferramentas. No entanto, a ideia deste trabalho é desenvolver um sistema autônomo, ou seja, que possa ser usado por pessoas que não necessariamente tenham conhecimento na área da fonoaudiologia. Além disso, deve-se permitir que o aplicativo desenvolvido por este trabalho seja difundido, permitindo seu uso em escolas, por professores, em hospitais públicos, com a finalidade realizar uma triagem fonológica. Finalmente, se detectado algum problema grave durante a avaliação, o paciente pode ser encaminhado para um SLP, para que seja feita uma análise mais aprofundada do problema. Isto é, realizar o diagnóstico e por fim, propor um tratamento adequando, com o objetivo de corrigir os problemas diagnosticados.

O próximo capítulo apresentará os passos necessários para atingir o objetivo final de construir um aplicativo de triagem fonológica. Portanto, apresenta a construção do sistema,

¹ Apraxia é uma condição neurológica em que a pessoa é incapaz de realizar voluntariamente movimentos motores mesmo que seus músculos estejam normais e ela saiba como fazer o movimento.

englobando a etapa de modelagem do sistema, bem como os resultados obtidos utilizando-se o sistema construído.

3 APLICAÇÃO *MOBILE* PARA TRIAGEM FONOLÓGICA INFANTIL

3.1 METODOLOGIA

Neste capítulo abordaremos questões relacionadas ao levantamento de requisitos, de forma a identificar os recursos que o aplicativo deve oferecer, o planejamento da mesma, visualizando-a de uma perspectiva de implementação. Dessa maneira, a seção 3.1 tem o objetivo de apresentar as ferramentas de modelagem utilizadas, a seção 3.2 apresenta as ferramentas de desenvolvimento utilizadas, a seção 3.3 apresenta os requisitos do aplicativo, bem como o modelo de dados utilizado, diagramas de classe, diagrama de entidade - relacionamento (DER), além de uma visão geral do sistema como um todo, a seção 3.4 tem por objetivo descrever como o sistema de sincronização foi desenvolvido, e finalmente, a seção 3.5 apresenta um cenário de avaliação utilizando o aplicativo. Finalmente, na última seção são apresentados os resultados obtidos utilizando o aplicativo.

3.1.1 Ferramentas de Modelagem

A construção de um *software* é um processo demorado e complexo, que envolve diversas etapas que devem ser obrigatoriamente executadas para se alcançar um produto final que consiga atender aos objetivos propostos de forma eficiente. Principalmente quando se trata de uma aplicação móvel, o qual o consumo de memória, conexão com *internet* e consumo de energia, são fatores que devem ser levados em consideração.

Existem diversas opções de ferramentas disponíveis no mercado, que podem ser utilizadas para a realização das tarefas de desenvolvimento, cada uma delas oferece diferentes tipos de funcionalidades. Além disso, cada uma possui suas vantagens e desvantagens na sua utilização, assim a escolha da ferramenta adequada para cada fase do desenvolvimento faz-se essencial, visando tornar a construção mais eficiente, rápida e precisa.

Durante o desenvolvimento de um *software* é possível encontrar uma grande quantidade de soluções válidas para um determinado problema, podendo sofrer constantes transformações durante seu ciclo de vida. Segundo Guedes (2011), os sistemas de informações são dinâmicos, ou seja, estão em constantes mudanças, que são ocasionadas por diversos fatores, como, por exemplo a realização de melhorias no sistema a pedido de um cliente, além de manter uma evolução constante ao longo do tempo.

A utilização de uma documentação detalhada, precisa e atualizada de todos os componentes do sistema junto com suas respectivas interações é de extrema importância, de maneira tal que a realização de modificações, correções de erros e melhorias possam ser realizadas de maneira rápida e descomplicada. Nesse sentido, a UML (*Unified Modeling Language*) tem como objetivo proporcionar uma representação visual do sistema, de modo a documentá-lo ao mesmo tempo que proporciona uma visão inicial do mesmo.

Dessa forma, a UML é uma linguagem de modelagem de propósito geral que pode ser aplicada a todos os domínios da aplicação. No entanto, a UML não é uma linguagem de programação, e sim uma notação, uma ferramenta disponível para auxiliar a definição das características do sistema.

A UML fornece uma série de diagramas com o objetivo de definir características do *software*, que vai desde sua estrutura lógica e comportamental e os requisitos até os meios físicos necessários para o desenvolvimento do sistema. Cada diagrama oferecido pela UML tem um objetivo específico no desenvolvimento de *software*. Por exemplo, o diagrama de atividades é um gráfico que mostra o fluxo de controle de uma atividade para outra mostrando-se de grande importância para a geração de uma "mapa" geral do sistema como um todo.

Porém, mesmo com as facilidades e a padronização proporcionada pela UML, a realização do processo de documentação do sistema de forma manual, torna o trabalho do engenheiro de software especialmente custoso, uma vez que existem diversos detalhes relacionados a cada diagrama, e que devem ser cuidadosamente levados em consideração durante a criação dos diagramas utilizados. O processo de desenvolvimento de tais diagramas torna-se muito mais simples, fácil e ágil quando são empregadas ferramentas no qual focam especificamente na criação e elaboração desses diagramas.

Nesse contexto, existe uma grande variedade de ferramentas que se propõem a auxiliar a elaboração dos diagramas UML, algumas com licença de uso privado e outras, com licença gratuita. Dentre as ferramentas mais utilizadas atualmente estão *StarUML* e *ArgoUML*, para o presente trabalho, optou-se por utilizar o *Astah* ².

O *Astah* é um *software* multiplataforma desenvolvido em java que tem por objetivo auxiliar a criação de diagramas UML. O *software* citado é oferecido em três versões: *AstahUML*, *Astah Professional* e *Astah Viewer*. Para o trabalho, foi escolhido a versão *Professional* do *Astah*, pelo fato de que tal versão oferece suporte a uma grande quantidade de diagramas, e

² Disponível para acesso em <http://astah.net/>

também, porque essa versão é gratuita para o uso acadêmico com algumas limitações.

Além dos diagramas fornecidos pela ferramenta *Astah*, existe o DER (*Diagrama Entidade Relacionamento*) o qual não é suportado na versão *Professional* do *Astah* na licença gratuita para uso acadêmico. Portanto, para o desenvolvimento de tal diagrama, foi utilizada o *MySQL Workbench*³. O *MySQL Workbench* é uma ferramenta de desenvolvimento de banco de dados *MySQL*, que permite realizar a modelagem do DER de um banco de dados, que permite exportar o modelo tanto em formato de imagem, como também permite exportar o modelo em um arquivo *SQL*. A ferramenta em questão foi escolhida pois é uma ferramenta gratuita, e simples de ser usada.

3.1.2 Ferramentas de Desenvolvimento

A análise e a modelagem do sistema é o processo inicial de construção de um *software*. Porém, da mesma forma que existem uma grande variedade de ferramentas focadas na criação de diagramas UML, existem também, inúmeras ferramentas que podem ser utilizadas para a real construção de um *software*.

Após a modelagem, o próximo passo no desenvolvimento do sistema consiste na implementação do modelo desenvolvido. Assim, a escolha da tecnologia que será utilizada no desenvolvimento está fortemente ligada tanto ao problema a ser resolvido quando à plataforma escolhida. Nesse contexto, aplicações móveis fazem o uso de diferentes tecnologias, trabalhando em conjunto para compor seu desenvolvimento.

A maioria das aplicações móveis funcionam sobre a arquitetura cliente-servidor, ou seja, a aplicação móvel (cliente) envia e recebe dados do servidor. Nesse contexto, a aplicação móvel desenvolvida por este trabalho funciona da mesma forma. A parte cliente, objetivo deste trabalho, faz contato diretamente com o usuário, portanto, precisa de uma interface gráfica amigável e de fácil usabilidade. Além disso, deve-se evitar o consumo elevado de memória e energia, já que os dispositivos móveis possuem essa restrição.

Para o desenvolvimento da aplicação, foi utilizado o ambiente de desenvolvimento *Android Studio*. Sua escolha se deve ao fato de que possui uma grande compatibilidade com os dispositivos móveis que funcionam sobre o sistema *Android*, pela fácil e simples usabilidade, por utilizar a linguagem *Java*, que permite utilizar orientação a objetos, e por ser uma ferramenta gratuita, além de possuir um editor de interfaces gráficas integrada.

³ Disponível para acesso em <https://www.mysql.com/>

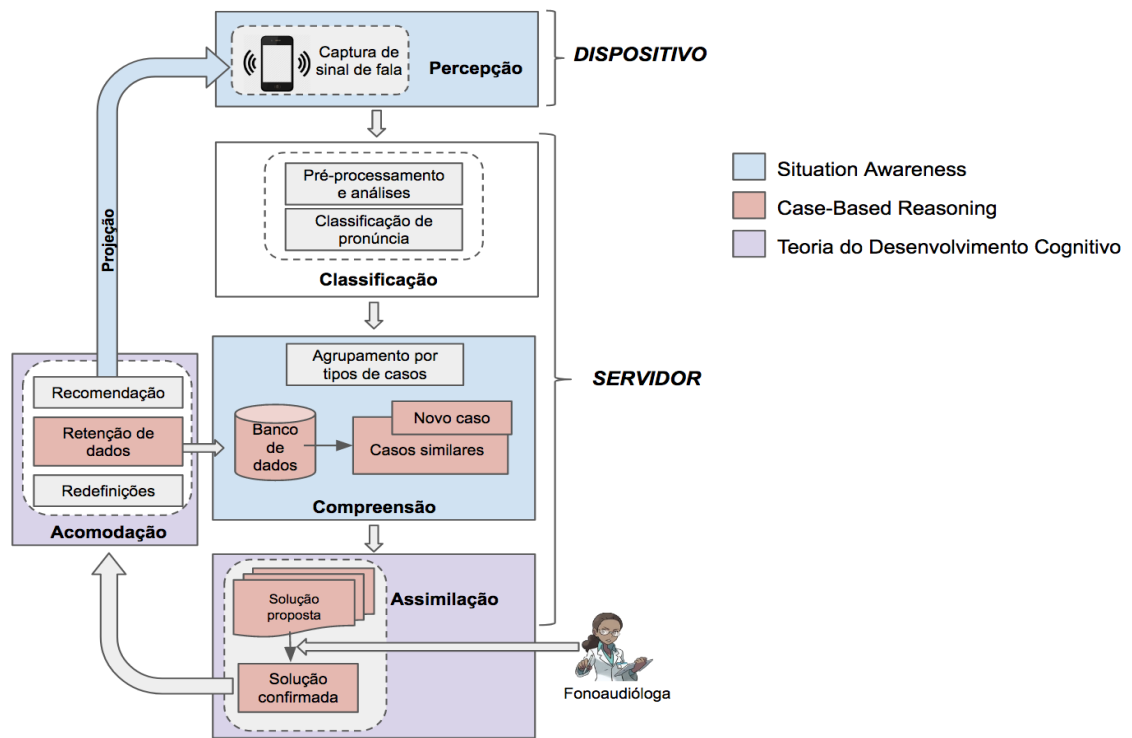
Inicialmente optou-se por desenvolver o aplicativo apenas para sistemas *android*, pois atualmente, representam a maioria dos dispositivos móveis em uso. Além disso, o *Android Studio* fornece algumas ferramentas que estão presentes apenas em dispositivos *android*, principalmente a ferramenta de reconhecimento de voz do *Android Studio*. No desenvolvimento do banco de dados utilizado na aplicação foi escolhido o banco *SQLite* porque esse sistema de banco de dados é nativo do *Android Studio*, além de ser voltado para dispositivos móveis.

A implementação da sincronização foi desenvolvida utilizando o *OkHttp*, que é uma biblioteca de código aberto, desenvolvida pela *Apache* que possui como principal objetivo fornecer um conjunto de ferramentas para o envio e recebimento de dados via requisições *http*. Além disso, para auxiliar no desenvolvimento e principalmente nos testes, foi utilizado o *software Postman*, que é uma ferramenta que tem como objetivo testar serviços *RESTful (Web APIs)* por meio do envio de requisições *HTTP* e da análise do seu retorno. Com ele é possível consumir facilmente serviços locais e na internet, enviando dados e recebendo o retorno das requisições enviadas. Portanto, permite a realização de testes das respostas recebidas e enviadas. Além das funcionalidades citadas o *Postman* também é um gerador de código das requisições em múltiplas linguagens, incluindo *java*, que é a linguagem padrão utilizada no *android studio*.

3.1.3 Modelagem e Desenvolvimento

O primeiro passo antes de iniciar o desenvolvimento propriamente dito, foi analisar a arquitetura do sistema proposta por (FRANCISCATTO, 2018), a qual é ilustrada na figura 3.1. Esta arquitetura propõe um sistema de duas camadas: cliente e servidor. A camada cliente, que é o objetivo deste trabalho, tem por objetivo realizar a coleta dos dados do usuário, e realizar a sincronização com o servidor para a obtenção dos resultados. Além disso, deve fornecer uma interface gráfica amigável para que sejam coletados os dados do usuário.

Figura 3.1: Arquitetura do sistema (FRANCISCATTO, 2018)

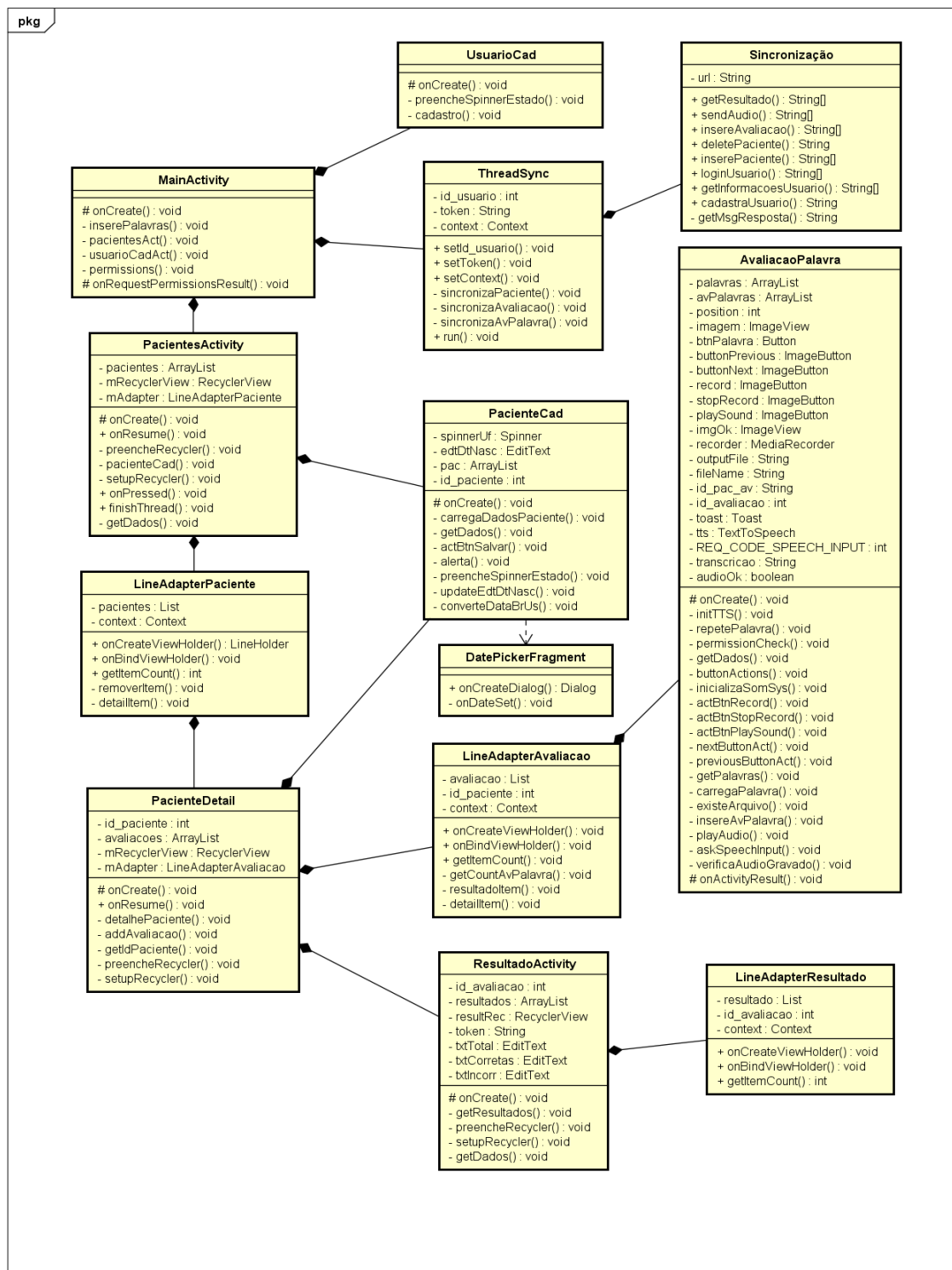


Fonte: (FRANCISCATTO, 2018)

O módulo da arquitetura que foi alvo de adaptação e implementação neste trabalho, foi o módulo de percepção, parte que realiza a coleta dos dados, e que entra em contato direto com o usuário do sistema, ou seja realiza uma ponte entre o usuário e o servidor.

O segundo passo foi realizar a modelagem do sistema, no qual foi desenvolvido um diagrama de classes preliminar, além de um DER do banco de dados. O diagrama de classes pode ser visto na figura 3.2 3.2.

Figura 3.2: Diagrama de Classes do Sistema



Fonte: Acervo Pessoal

Como se pode observar no diagrama de classes o aplicativo possui uma série de classes, cada uma com uma função específica, onde a maioria das classes representa uma tela específica

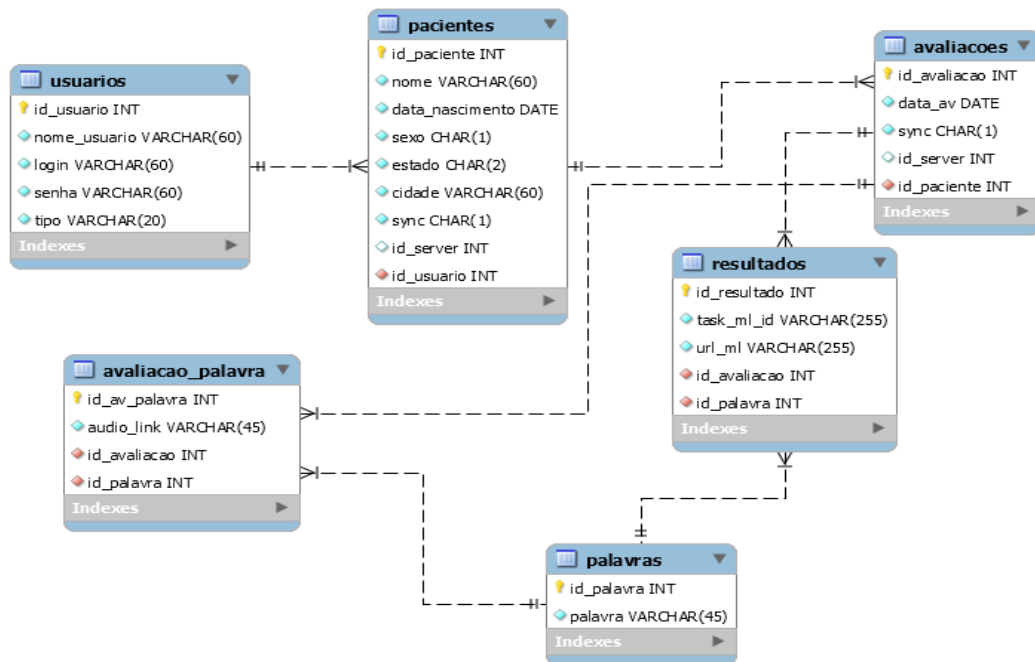
do aplicativo. Entre elas está a tela inicial que no diagrama está representado pela *MainActivity*, a qual é apresentada ao usuário campos de login, além de opções para realizar um novo cadastro e utilizar o aplicativo de forma *offline*. Existe também as classes *PacientesActivity*, *UsuarioCad*, *PacienteCad* e *PacienteDetail* que têm por objetivo, respectivamente, apresentar uma lista de pacientes cadastrados, cadastrar um novo usuário, cadastrar um novo paciente e apresentar as avaliações cadastradas para determinado usuário cadastrado. Como se pode observar, o diagrama apresenta uma visão gráfica do aplicativo como um todo. A partir da classe *MainActivity* é possível navegar para qualquer parte do aplicativo, desde que as restrições impostas para determinada etapa estejam cumpridas, como por exemplo, os resultados já estejam disponíveis para abrir a etapa de resultados.

Existem classes não gráficas, ou seja, classes que executam alguma ação em *background*, como por exemplo a classe *Sincronização*, que tem por objetivo realizar a sincronização dos dados com o servidor. Tal classe, possui métodos que são executados de forma oculta, ou seja, o usuário não percebe que determinados métodos são executados, para permitir uma maior transparência ao usuário. Além disso a classe *ThreadSync*, que é uma *Thread* que possui o objetivo de realizar chamadas aos métodos da classe *Sincronização* a fim de executar a sincronização dos dados com o servidor.

A coleta dos dados é feita pela classe *AvaliacaoPalavra*, a qual apresenta ao usuário uma sequência de imagens e palavras relacionadas, onde o aplicativo deve gravar sua pronúncia do usuário. Posteriormente o aplicativo deve realizar a sincronização dos dados com o servidor para que sejam feitas as análises. Um exemplo da tela de avaliação é mostrado na figura ??.

Uma vez criado o diagrama de classes, foi então elaborado o DER do sistema que é mostrado na figura 3.3.

Figura 3.3: DER do sistema

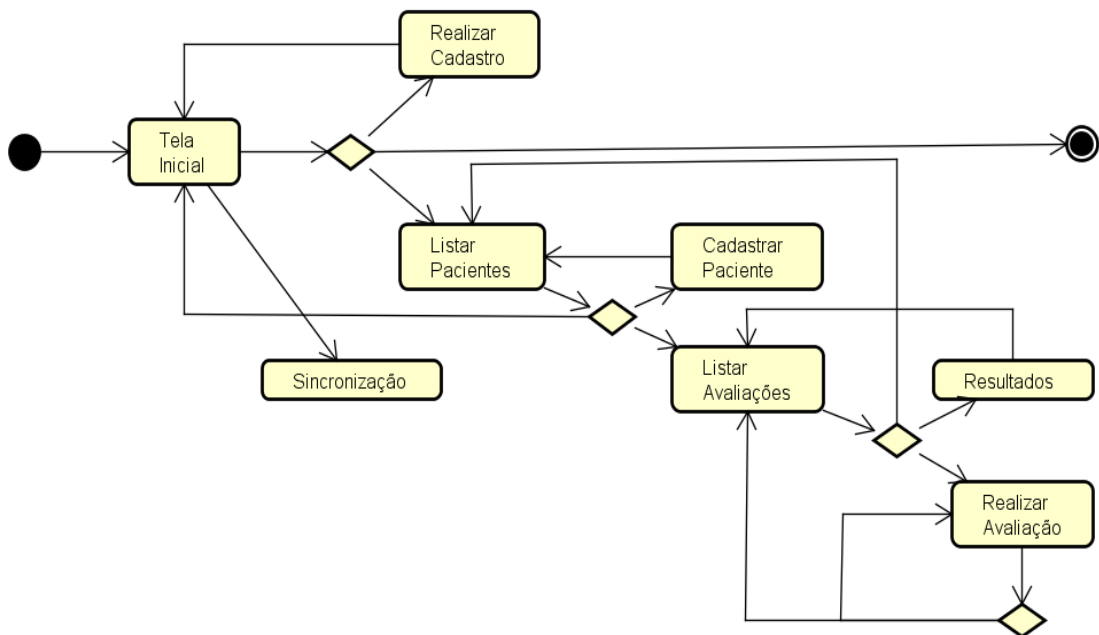


Fonte: Acervo Pessoal

O modelo apresentado é composto por 6 tabelas, onde a tabela *usuarios* guarda os dados de um usuário, onde um usuário pode ser de dois tipos: paciente e terapeuta, para fins de simplificação, este trabalho implementou apenas o modo terapeuta para o funcionamento. A tabela *Pacientes* guarda os dados pessoais do paciente, a tabela *Avaliacaoes* guarda os dados de uma avaliação. É importante salientar que um paciente pode realizar várias avaliações. A tabela *avaliacao_palavra* guarda a referência para o audio gravado de cada palavra, que deverá ser enviado ao servidor onde serão feitas as análises, e finalmente, a tabela *resultados*, que guarda os dados de uma avaliação, esta tabela, guarda a rota para acessar um resultado no servidor, além de um identificador, também gerado pelo servidor. O áudio gravado é salvo na memória interna do dispositivo apenas até o seu envio ao servidor, após isso, o áudio é apagado da memória com o objetivo de economizar espaço interno.

Foi elaborado também um diagrama de atividades do aplicativo que pode ser visto na figura 3.4.

Figura 3.4: Diagrama de Atividades do aplicativo



Fonte: Acervo Pessoal

Como se pode observar no diagrama de atividades, inicialmente o usuário é levado à uma tela inicial, onde é apresentado três opções: fazer login, realizar cadastro e utilizar *offline*. A opção de fazer login, exige conexão com rede para seu funcionamento, pois mantém a sincronização ativa durante todo o período em que o usuário estiver usando o aplicativo. A opção de realizar cadastro também exige conexão com rede, pois realiza o cadastro de um novo usuário no sistema. Nesta etapa, quando o cadastro for confirmado no servidor, o usuário também é inserido no banco de dados local, para tornar possível o uso do dispositivo sem conexão com rede. Finalmente, a opção de utilizar *offline* permite que um usuário previamente cadastrado no servidor faça avaliações sem a necessidade de conexão com rede. Os dados coletados serão enviados para o servidor quando o mesmo usuário que realizou a avaliação faça o *login*.

Quando concluída a etapa da tela inicial, o usuário é levado à opção de realizar cadastro ou à etapa de listar os pacientes cadastrados. A opção de realizar cadastro, como foi exposto anteriormente, exige que o usuário informe seus dados, como nome, login e senha e os envia ao servidor. A etapa de listar pacientes apresenta ao usuário uma lista de pacientes cadastrados no banco de dados. Ainda na etapa de listar pacientes, o aplicativo fornece ao usuário a opção de cadastrar um novo paciente, ou de listar as avaliações de um paciente em específico.

A etapa cadastrar paciente, permite que um usuário cadastre um novo paciente, esta

etapa pede que o usuário informe os dados do paciente tais como nome, data de nascimento, sexo, cidade e estado. Já na etapa de listar avaliações, o usuário consegue visualizar todas as avaliações realizadas por um paciente, identificadas pela data da avaliação. Além disso, para cada avaliação presente na lista, é possível visualizar os resultados da mesma, além de ser possível cadastrar uma nova avaliação.

Na etapa de resultados, é mostrado ao usuário os resultados de determinada avaliação selecionada, mostrando os resultados por palavra avaliada. Portanto, para cada palavra existe um indicador que diz se o paciente pronunciou corretamente. Além disso, é fornecido também as estatísticas dos resultados como total de palavras avaliadas, o número de acertos e o número de erros.

A etapa de realizar avaliação, é a etapa em que o usuário realiza efetivamente uma avaliação. Nesta etapa é apresentado ao paciente uma sequência de imagens e palavras, onde é feita a coleta de dados.

Finalmente, existe a etapa de sincronização, que é uma etapa não visual, ou seja, esta etapa funciona em *background*, de forma transparente ao usuário. A etapa de sincronização é iniciada quando o usuário realiza o login no sistema, ou seja, quando existe conexão com rede. O sistema de sincronização é apresentado na próxima seção.

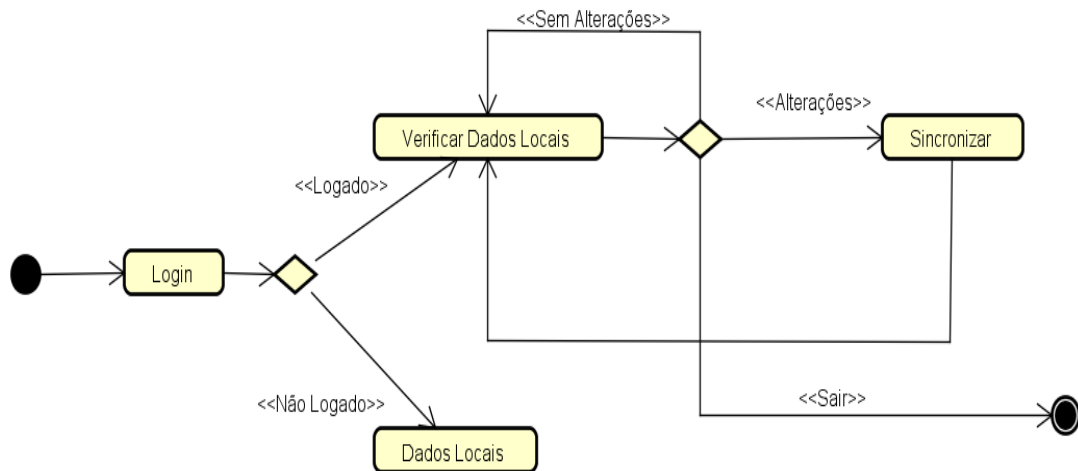
A tela de avaliação disponibiliza ao usuário basicamente seis botões, entre eles, o botão onde fica a palavra, a qual repete-a para o usuário utilizando o sintetizador de voz do *Google*, um botão para a gravação do áudio, um botão para parar a gravação, outro botão para ouvir o áudio que foi gravado, e finalmente, os botões que permitem navegar entre as imagens e palavras.

3.1.4 Sincronização

A sincronização dos dados coletados pelo aplicativo com o servidor é parte vital para o completo funcionamento do sistema, pois, apenas o servidor pode realizar a análise fonológica dos dados, fornecendo os resultados ao aplicativo para disponibilizá-los aos usuários.

Inicialmente, foi projetado um diagrama de atividades para o sistema de sincronização, que pode ser vista na figura 3.5.

Figura 3.5: Sistema de sincronização



Fonte: Acervo Pessoal

No banco de dados foi realizada a adição de colunas, com a finalidade de sinalizar a situação dos dados, ou seja, indicar se um determinado registro já foi enviado para o servidor. Resumindo, nas principais tabelas do sistema foram adicionadas duas colunas: *sync* e *id_server*, a primeira tem por objetivo indicar se o registro em questão já foi submetido o servidor, e a segunda tem a finalidade de guardar o identificador gerado durante a inserção do registro no servidor. A coluna *id_server* foi definida como uma coluna não obrigatória, pois como o aplicativo foi projetado para funcionar independentemente da conexão com rede, é necessário manter uma chave primária independente localmente no dispositivo.

A sincronização foi implementada utilizando um objeto com todos os métodos responsáveis por enviar e receber as mensagens do servidor. No entanto, foi necessário criar uma forma de realizar verificações frequentes dos dados localmente para detectar alterações, ou seja, todas as alterações realizadas no aplicativo, devem refletir no servidor quando existir conexão com *internet*. Para essa finalidade foi utilizada uma *Thread*, que é ativada quando o usuário está logado no sistema, e permanece executando, verificando as alterações realizadas localmente e submetendo-os ao servidor.

As avaliações, porém, quando enviadas ao servidor, são apagadas do dispositivo para evitar um desperdício de memória interna, já que dispositivos móveis possuem algumas limitações. Portanto, a *Thread* que verifica as alterações nas avaliações, realiza buscas no banco de dados local, procurando por registros existentes, que, quando encontrados, são enviados ao

servidor e ao final, quando confirmado o envio, são excluídos do dispositivo.

Quando o usuário efetua o *login*, o aplicativo imediatamente verifica se é a primeira vez que o *login* é efetuado. Caso seja positivo, o aplicativo imediatamente começa a buscar os dados referentes ao usuário no servidor e os transfere para o dispositivo. Caso o usuário esteja conectado com o servidor, é feita uma verificação nos dados locais para determinar quais as alterações pendentes. A determinação das alterações pendentes é realizada através da coluna *sync*, que foi adicionada para este fim. Cada registro que ainda não foi enviado para o servidor tem a coluna *sync* com o valor "N". Portanto, frequentemente é efetuada a busca dos registros que estão com o valor da coluna *sync* igual ao valor "N", tais dados são enviados ao servidor. Finalmente, após a confirmação da alteração dos dados no servidor, a coluna "sync" é alterada para "S", indicando que a mesma está sincronizada com o servidor.

Os resultados foram implementados sem o uso de sincronização, ou seja, apenas estão disponíveis quando existir conexão com internet. Graficamente os resultados são mostrados como uma lista de palavras, que, ao lado de cada uma, é mostrado um símbolo de correto para as palavras corretas e um símbolo de incorreto para as incorretas. Além disso, é fornecido uma estatística com os dados da avaliação, onde é feita uma contagem do total de palavras, o número de palavras corretas e o número de palavras incorretas.

3.2 CENÁRIO DE AVALIAÇÃO

Esta seção tem por objetivo apresentar e detalhar na prática, com imagens de cenários de testes como o aplicativo funciona.

Primeiramente, quando o usuário inicia o aplicativo, ele é levado à tela inicial, onde é exigido o login e senha. Porém, no primeiro uso, é necessário realizar um cadastro, além disso, é importante salientar que no primeiro uso é essencial que esteja disponível conexão com internet, para que o sistema possa realizar os devidos registros dos dados pessoais do usuário.

Um cenário da tela inicial é apresentado na figura 3.6

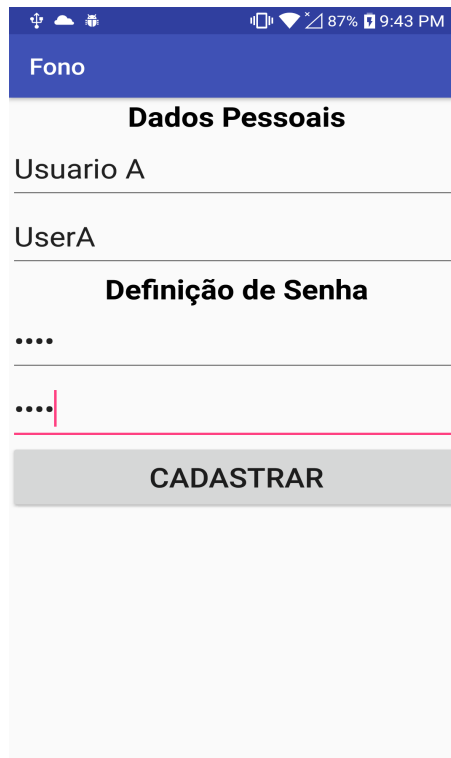
Figura 3.6: Tela Inicial do Aplicativo

A imagem mostra a tela inicial do aplicativo 'Fono'. No topo, há uma barra azul com o nome 'Fono' em branco. Abaixo, o título 'Login' está centralizado. Há dois campos de entrada: 'Email' e 'Senha', ambos com linhas de base vermelhas. Abaixo dos campos, há três botões cinza: 'ENTRAR', 'REALIZAR CADASTRO' e 'UTILIZAR OFFLINE'. O status bar no topo indica 86% de bateria e 9:31 PM.

Fonte: Acervo Pessoal

Como se pode observar, é mostrado ao usuário a opção de "Entrar", "Realizar Cadastro" ou "Utilizar Offline", como neste caso é um primeiro uso, deve ser realizado o cadastro pressionando o botão "Realizar Cadastro". Ao pressionar o botão, o usuário será levado à tela apresentada na figura 3.7.

Figura 3.7: Tela de Cadastro de Usuario



The image shows a mobile application registration screen. At the top, there is a blue header with the text "Fono". Below the header, the title "Dados Pessoais" is centered. There are two input fields: the first contains "Usuario A" and the second contains "UserA". Below these is the title "Definição de Senha" followed by two password input fields, both containing four dots. At the bottom, there is a grey button labeled "CADASTRAR".

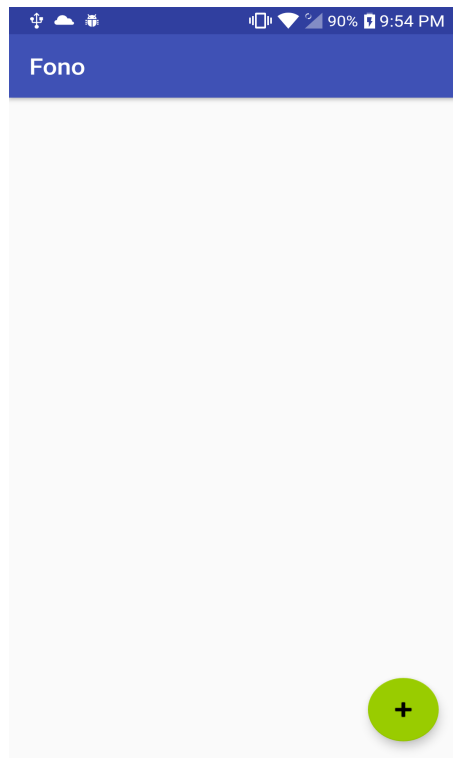
Fonte: Acervo Pessoal

A imagem mostra a tela de cadastro de usuário, onde deve ser preenchido o nome do usuário. Levando-se em consideração um usuário fictício chamado "Usuario A", cujo login "UserA" e finalmente uma senha, que deve ser repetida no último campo, que neste caso é "1234". Ao pressionar o botão "CADASTRAR", os dados preenchidos são enviados para o servidor para que seja realizado o cadastro do usuário. Quando o aplicativo receber a confirmação de que o usuário foi cadastrado no servidor, o usuário é inserido no banco de dados local para permitir que o aplicativo seja utilizado de forma *offline*.

Ao final, ao ser pressionado o botão "Cadastrar", o usuário é levado novamente à tela da figura 3.6, onde os dados informados no cadastro devem ser preenchidos.

Ao preencher os dados da figura 3.6, o usuário tem duas opções: "Entrar" ou "Utilizar Offline". Ambas as opções levam o usuário à uma mesma tela, com a diferença que, se o usuário pressionar o botão "Entrar", ele obrigatoriamente necessitará de conexão com internet, ao contrário de "Utilizar Offline", que não obriga o uso de *internet*. Ao selecionar uma das opções anteriormente explicadas, o usuário será levado à tela da figura 3.8.

Figura 3.8: Tela de listagem dos pacientes



Fonte: Acervo Pessoal

Na figura acima, são listados os pacientes cadastrados para determinado usuário, como neste caso é um primeiro uso, não haverá nenhum paciente cadastrado, portando deve-se cadastrar um novo paciente. Para isso deve-se pressionar o botão verde com o símbolo "+", que levará o usuário à tela de cadastro de paciente, que é representada pela figura 3.9.

Figura 3.9: Tela de cadastro de paciente

Fono

Identificação

Paciente C

28/11/2013

Sexo

Masculino

Feminino

Região

RS

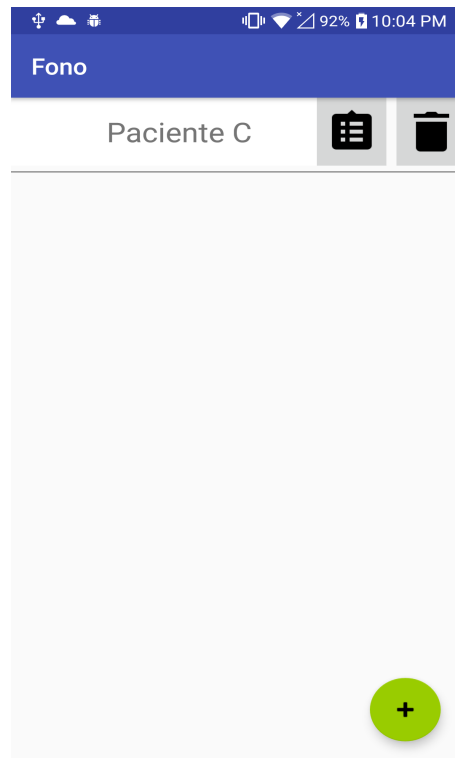
Santa Maria

SALVAR

Fonte: Acervo Pessoal

A figura acima apresenta a tela de cadastro de um novo paciente, nela é exigido que o usuário forneça o nome do paciente, data de nascimento, sexo, estado e cidade, todos os campos são obrigatórios. Considerando um paciente fictício chamado "Paciente C", nascido em "28/11/2013" do sexo "Masculino", do estado do "RS - Rio Grande do Sul", da cidade de Santa Maria, é inserido no aplicativo. Ao final, o usuário deve pressionar o botão salvar e será levado de volta à tela da figura 3.8, onde o paciente cadastrado será listado. Um exemplo é mostrado na figura 3.10

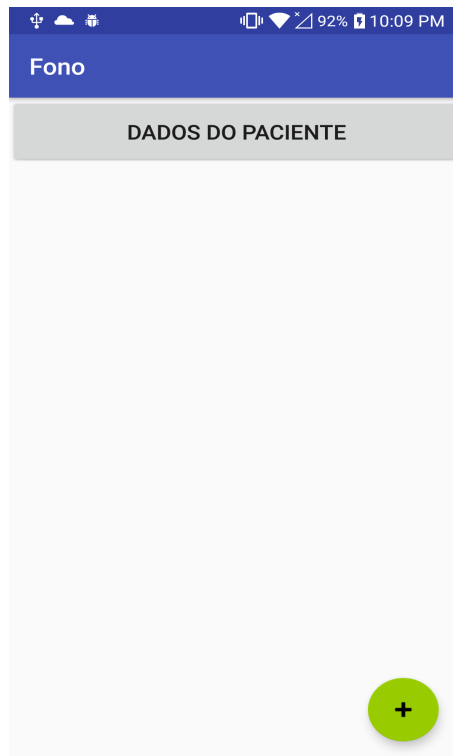
Figura 3.10: Tela de listagem de pacientes



Fonte: Acervo Pessoal

A figura acima mostra o paciente anteriormente cadastrado, com dois botões no item, um para detalhar o paciente e outro para excluí-lo. O botão de detalhar o paciente, leva o usuário à tela da figura 3.11.

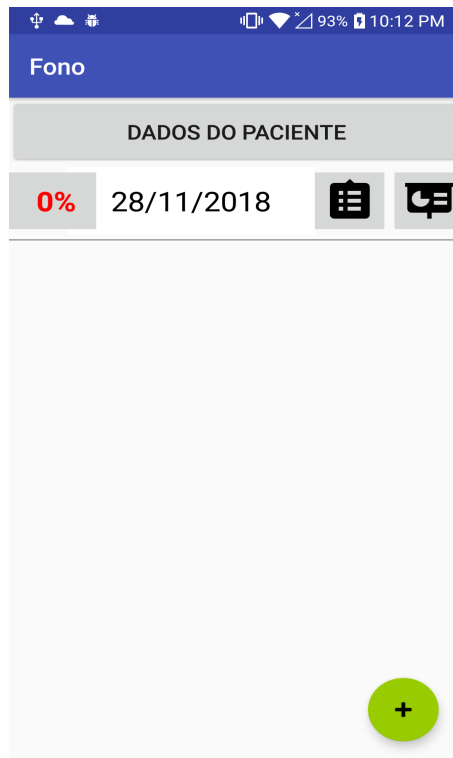
Figura 3.11: Tela de listagem de avaliações



Fonte: Acervo Pessoal

Como se pode observar, não há nenhuma avaliação cadastrada para o paciente selecionado, portanto, deve-se pressionar o botão verde com o símbolo "+" para adicionar uma nova avaliação. Feito isso, será listada a avaliação criada, identificada pela data de criação, que é a data atual. Um exemplo pode ser visto na figura 3.12.

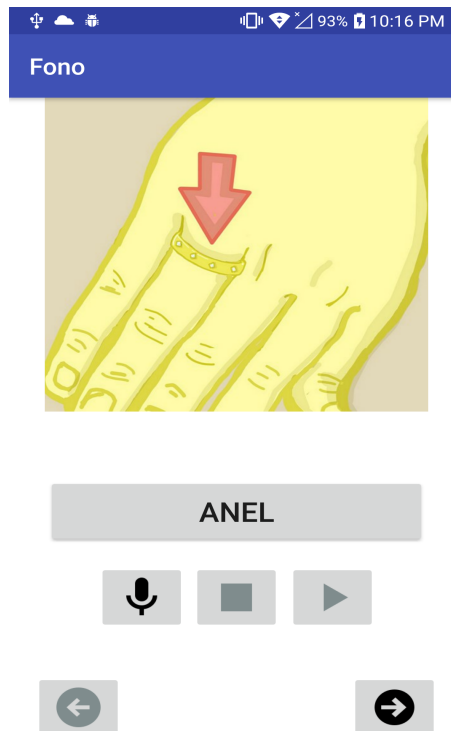
Figura 3.12: Tela de listagem de avaliações



Fonte: Acervo Pessoal

A figura mostra uma avaliação cadastrada, à esquerda em vermelho, mostra o percentual concluído da avaliação, o primeiro botão à direita leva o usuário à tela de realizar a avaliação e o botão mais à direita leva o usuário aos resultados. A tela de avaliação pode ser vista na figura 3.13

Figura 3.13: Tela de Avaliação



Fonte: Acervo Pessoal

Como pode ser visto na figura acima, é apresentado ao usuário uma sequência de imagens e palavras, onde cada palavra é representada por uma imagem, o usuário deve pressionar o botão com o símbolo de gravar para capturar o áudio e finalmente pressionar o botão da seta direita para avançar, ou a seta esquerda para voltar.

Finalmente a tela de resultados, que é mostrada quando o usuário pressionar o botão mais à direita da figura 3.13. A tela de resultados das avaliações é apresentado na figura 3.14.

Figura 3.14: Tela de resultado do aplicativo

Fono	
Anel	✓
Barriga	✓
Batom	✓
Bebê	✓
Beijo	✗
Biblioteca	✓
Bicicleta	✗
Bolso	✗
Total: 10 -- 100.00% Corretas: 5 -- 50,00% Incorretas: 5 -- 50,00%	

Fonte: Acervo Pessoal

Como mostrado acima, a tela de resultados apresenta a lista de palavras avaliadas, com seus respectivos resultados ao lado, o símbolo verde significa que a análise considerou a avaliação como correta, e o símbolo vermelho significa que a análise considerou a avaliação incorreta. Além disso, abaixo da lista de palavras é mostrada algumas estatísticas da avaliação, tais como o total de palavras avaliadas, o total de corretas e o total de incorretas.

Com o objetivo de deixar mais claro o funcionamento do aplicativo, foi desenvolvido um fluxograma gráfico baseado nas telas do aplicativo, apresentando as telas as quais o usuário é levado quando pressiona determinado botão ou executa determinada ação. O fluxograma é apresentado em várias imagens afim de facilitar sua visualização. A etapa inicial do fluxograma é mostrado na figura 3.15.

Figura 3.15: Fluxograma gráfico



Fonte: Acervo Pessoal

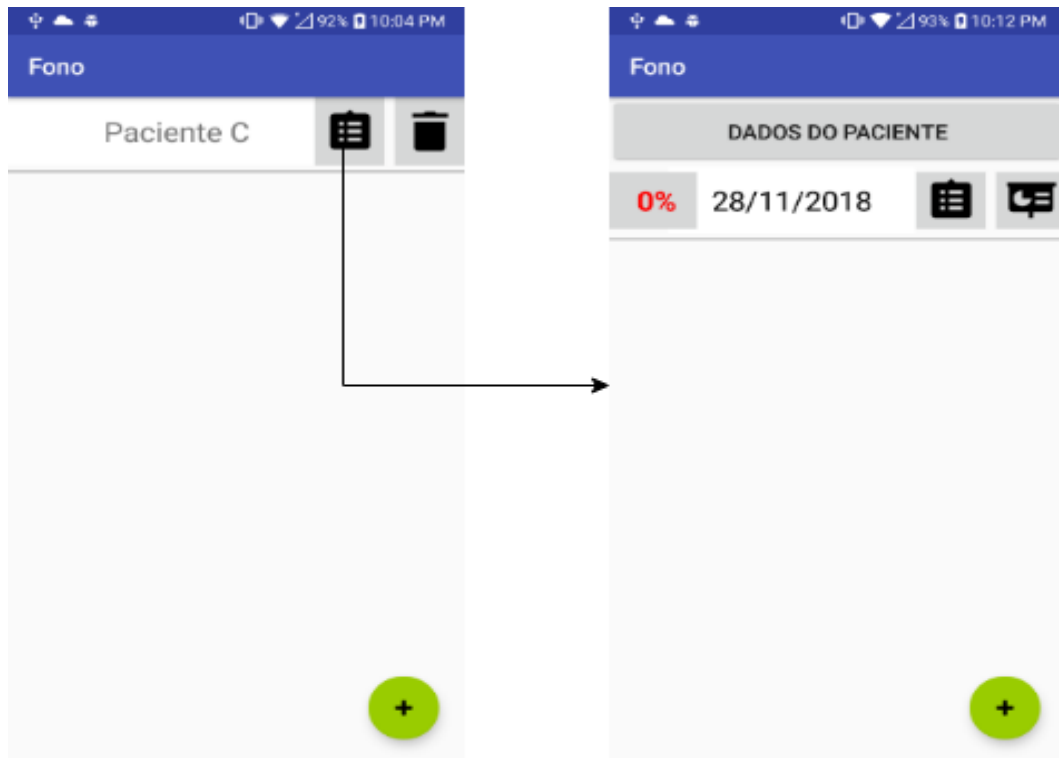
Como se pode observar pela figura acima, a tela inicial do aplicativo possui 3 botões, onde para cada botão é apresentada uma linha a qual liga à tela em que determinado botão está relacionado, ou seja, representa a tela apresentada ao usuário quando o botão é pressionado. Portanto, ambos os botões "ENTRAR" e "UTILIZAR OFFLINE" levam o usuário à mesma tela, que na imagem é a tela inferior direita, e o botão "REALIZAR CADASTRO" leva o usuário à tela de cadastro de usuários que é representada pela tela superior direita da imagem. No cadastro de um novo paciente, quando pressionado o botão "CADASTRAR" da tela de cadastro, o usuário é levado novamente à tela inicial do aplicativo, para que possa realizar o login para finalmente utilizar o aplicativo.

A partir da tela de listagem de pacientes, o usuário pode excluir o paciente ou selecioná-

lo para listar suas avaliações. Tal esquema é mostrado na figura 3.16.

Levando-se em consideração a tela de avaliações do paciente selecionado, o usuário possui duas opções: iniciar uma nova avaliação ou listar os resultados para a avaliação selecionada. Um fluxograma mostrando isso pode ser visto na 3.16.

Figura 3.16: Fluxograma gráfico

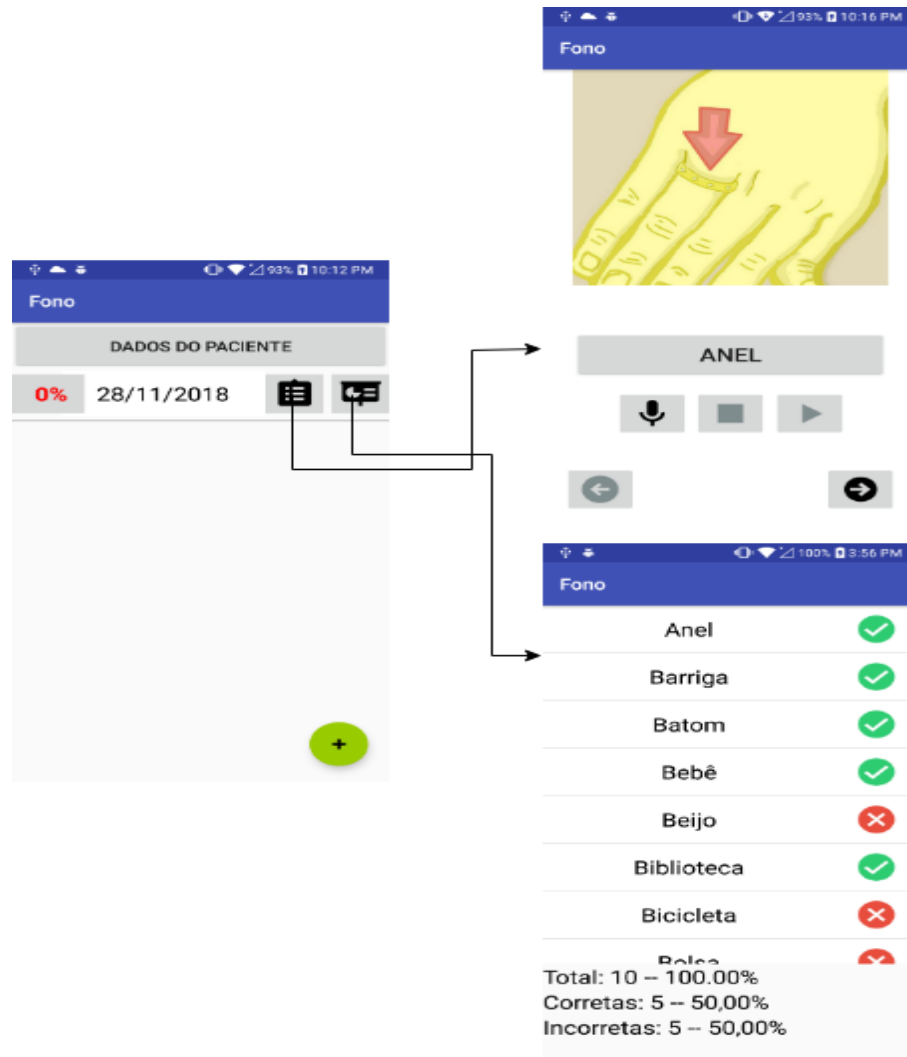


Fonte: Acervo Pessoal

Como mostrado na figura acima, quando os pacientes são listados, o usuário pode selecionar pressionando no botão de saída de seta do fluxo. Portanto, quando pressionado, o botão leva o usuário à lista de avaliações do paciente selecionado, que é mostrada na tela da direita da imagem.

Quando o usuário está na tela de listagem de avaliações de determinado paciente, ele tem a opção de realizar uma nova avaliação, ou de visualizar os resultados de determinada avaliação, o fluxo neste caso é apresentado pela figura 3.17.

Figura 3.17: Fluxograma gráfico



Fonte: Acervo Pessoal

3.3 RESULTADOS

Os resultados foram obtidos utilizando registros já validados pela fonoaudiologia retirados do software *Infono*. Foram escolhidas as palavras "Anel" e "Batom" para realizar os testes. Tal escolha se deve ao fato de que as palavras citadas foram mais treinadas que as demais palavras do sistema. Ao todo, foram escolhidas aleatoriamente 100 palavras de cada, portanto, 200 palavras no total. Além disso, devido às restrições do aplicativo, foram inseridos dados manualmente para possibilitar a realização dos testes. Foram utilizados *scripts SQL* para realizar a inserção manual de dados no aplicativo. Um exemplo de inserção utilizado no *script* pode ser visto na figura 3.18.

Figura 3.18: Script de Inserção

```
INSERT INTO avaliacao_palavra (  
    audio_ok,  
    id_palavra,  
    audio_link,  
    id_avaliacao,  
    id_av_palavra  
)  
VALUES (  
    'N',  
    2131558402,  
    '/storage/emulated/0/3.wav',  
    1,  
    518  
);
```

Fonte: Acervo Pessoal

Na imagem acima, é inserido uma avaliação de um caso extraído do *Infono*. É inserido um caminho com o áudio que estará no dispositivo para ser analisado, é através do *audio_link* que o áudio é acessado para que sejam feitas as análises.

Foi desenvolvido também um pequeno programa em java para buscar os registros no banco de dados do *Infono* bem como seus respectivos áudios. Um pequeno trecho de código que mostra o funcionamento desse pequeno programa pode ser visto na figura 3.19.

Figura 3.19: Programa Java

```

33 public static void main(String[] args) throws IOException {
34     Connection c = new ConnectionFactory().getConnection();
35     Consultas cons = new Consultas(c);
36     Connection c1 = new Conexao().getConexao();
37     ArrayList<AvaliacaoPalavra> avp = cons.getAvaliacoes("Batom");
38     Insert ins = new Insert(c1);
39     for (int i = 0; i < avp.size(); i++) {
40         ins.inserereAvPalavra(avp.get(i));
41         String pathSource = caminho1 + avp.get(i).getAudio_link();
42         String pathDest = caminho2 + avp.get(i).getAudio_link();
43         pathSource = pathSource.replace("/storage/emulated/0/", "");
44         pathDest = pathDest.replace("/storage/emulated/0/", "");
45         File source = new File(pathSource);
46         // diretorio de destino
47         File dest = new File(pathDest);
48         copyFile(source, dest);
49     }
50 }

```

Fonte: Acervo Pessoal

A figura acima mostra as etapas que são realizadas para buscar os dados do *infono* e inserir no aplicativo. A linha 34 realiza a conexão com o banco de dados do *Infono*, a linha 36 realiza a conexão com o banco de dados local do aplicativo, a linha 37 realiza a busca dos dados no banco de dados do *Infono* e os insere em uma *array*. Os dados são inseridos e processados dentro do *loop for* a partir da linha 39. A linha 40 realiza a inserção dos dados presentes no *array* que foi preenchido com os dados do *Infono* no banco de dados do aplicativo. A linha 41 realiza a cópia dos áudios da pasta do *Infono* e a linha 42 insere os áudios copiados para uma pasta de destino. Finalmente, a última etapa antes de realizar os testes, foi transferir os áudios para o dispositivo móvel.

O teste foi realizado no sistema como um todo, porém, efetuando o envio dos dados a partir do dispositivo móvel, para realizar o processamento dos mesmos no servidor. No entanto, foi necessário realizar a inserção manual dos dados no dispositivo.

Após a realização dos testes, foram realizadas consultas SQL no servidor para a extração dos resultados. As consultas utilizadas podem ser vistas na figura 3.20.

Figura 3.20: Consultas

```
SELECT COUNT(*)  
FROM tasks  
WHERE result = 1  
  
SELECT COUNT(*)  
FROM tasks  
WHERE result = 0
```

Fonte: Acervo Pessoal

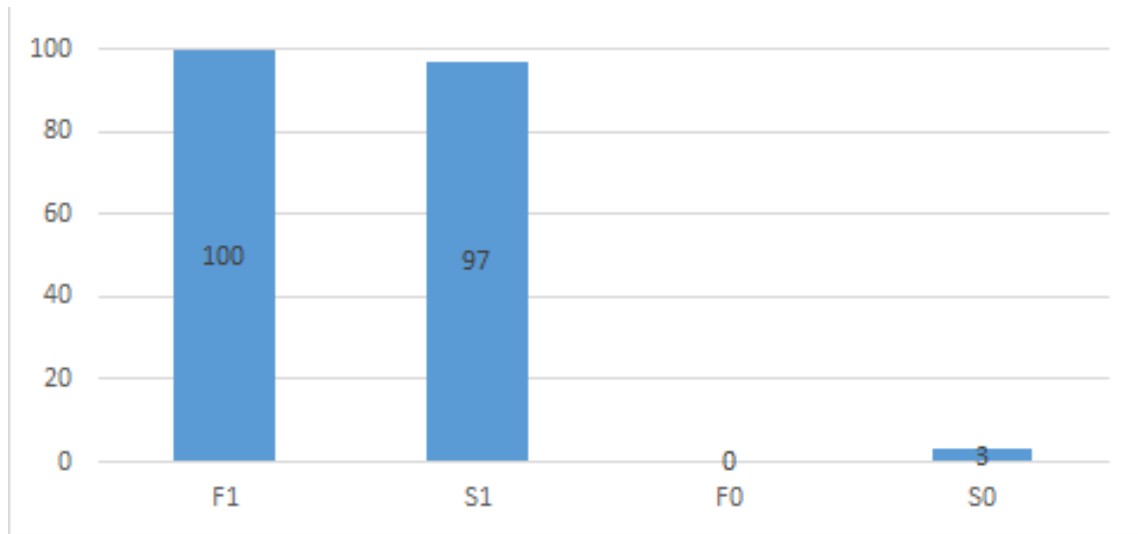
A primeira consulta busca as avaliações nas quais o resultado foi igual ao valor 1, ou seja, as avaliações nas quais o sistema considerou correto. A segunda consulta busca as avaliações nas quais o resultado foi igual a 0, portanto, as avaliações nas quais o sistema considerou como erradas.

O figura 3.21 e 3.22 apresentam os gráficos com os resultados da palavra "Anel" e "Batom" respectivamente.

Os resultados foram classificados da seguinte forma:

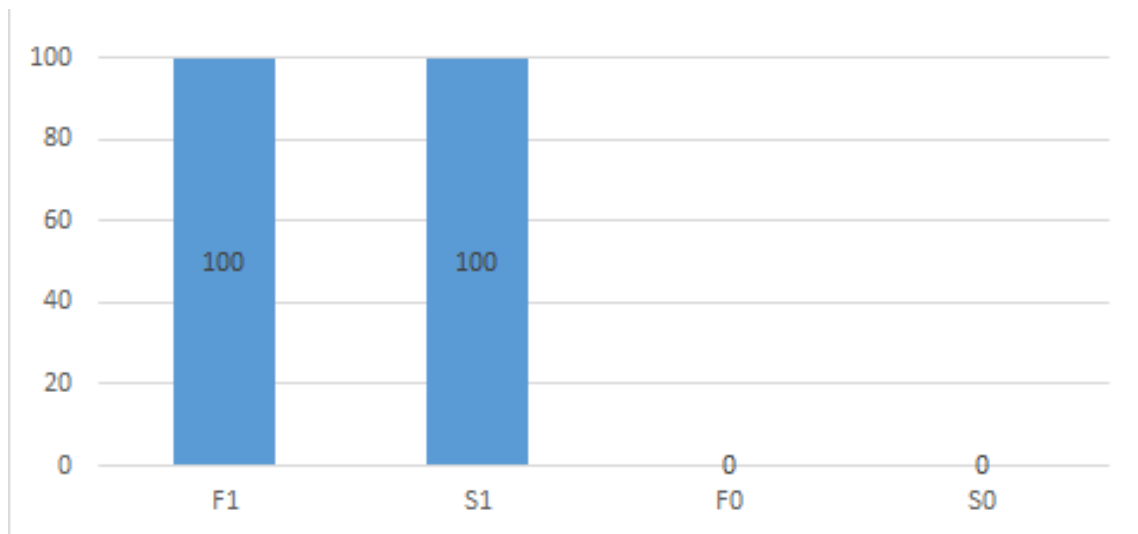
- **F1**: Representa as palavras nas quais a fonoaudiologia considerou como corretas;
- **S1**: Representa as palavras nas quais o aplicativo considerou como corretas;
- **F0**: Representa as palavras nas quais a fonoaudiologia considerou como incorretas;
- **S0**: Representa as palavras nas quais o aplicativo considerou como incorretas;

Figura 3.21: Resultados da palavra Anel



Fonte: Acervo Pessoal

Figura 3.22: Resultados da palavra Batom



Fonte: Acervo Pessoal

A palavra "Anel" a fonoaudiologia considerou que todas as 100 avaliações testadas estavam corretas, porém, o sistema considerou 97 delas como corretas, e 3 como incorretas, apresentando neste caso uma confiabilidade de 97%. É possível observar que tanto a palavra "Anel" quando a palavra "Batom" a fonoaudiologia considerou 100% das avaliações escolhidas como corretas, isto se deve ao fato de que "Anel" e "Batom", são as palavras as quais apresentam menor frequência de erros entre os pacientes, ou seja, são palavras mais simples e menos

passíveis a erros.

A palavra "Batom" a fonoaudiologia novamente considerou que todas as 100 avaliações testadas estavam corretas, o sistema, apresentou o mesmo resultado, ou seja, houve uma concordância de 100% entre a fonoaudiologia e o sistema. Como se pode observar nos resultados como um todo, o aplicativo apresentou um grau de confiabilidade superior a 95 %, atingindo o objetivo proposto, de fornecer ao usuário um resultado com uma confiabilidade aceitável para possibilitar a recomendação de apoio fonológico.

4 CONCLUSÃO E TRABALHOS FUTUROS

PETs, tem como objetivo facilitar a vida dos SLPs. Além disso, com a avanço da tecnologia, especialmente na área da computação móvel, tornou possível fornecer suporte a pacientes mesmo que não sejam submetidos a uma PE especializada. Permitiu também, a possibilidade de disponibilizar ferramentas que possam realizar triagem fonológica no público infantil que pode reduzir o risco de que uma PD se desenvolva em um paciente, o que pode ocasionar outros problemas em longo prazo. Portanto, a detecção precoce de tais problemas são de suma importância.

O desenvolvimento da ferramenta proposta por este trabalho apresentou alguns desafios durante seu desenvolvimento. No entanto, o resultado obtido apresentou uma confiabilidade bastante alta. Além disso, foi possível validar os resultados utilizando os dados do software *Infono*, o qual possui dados que já foram previamente avaliados por fonoaudiólogos. Além disso, o desenvolvimento de uma arquitetura que torne possível o uso do aplicativo independentemente de conexão com rede é essencial para que seja possível a coleta de dados em lugares onde não existe conexão, pois dessa forma, permite que mesmo pessoas que residem em um local distante tenham acesso à triagem fonológica.

Assim, esta ferramenta permite que sejam coletados áudios de pacientes, armazenados no dispositivo, e, quando existir conexão com internet, esses áudios são enviados para um servidor para que seja realizadas análises, com o objetivo de detectar distúrbios fonológicos. Além disso, o aplicativo fornece um resultado ao usuário, informando quais foram as avaliações problemáticas e quais as corretas. A ferramenta através dos resultados obtidos mostrou-se confiável e eficaz na função de realizar uma triagem fonológica, permitindo que pacientes portadores de distúrbios fonológicos realizem uma avaliação preliminar com o objetivo de verificar a necessidade de busca por apoio fonológico, com isso, é possível à logo prazo, com o aprimoramento da ferramenta, seja possível diminuir a incidência de distúrbios fonológicos, e além disso, que faça com que os pacientes busquem por apoio fonológico para o diagnóstico e tratamento adequado dos problemas que porventura sejam diagnosticados.

Futuramente, a proposta é aprimorar o aplicativo, ou seja, corrigir eventuais problemas que sejam detectados durante seu uso. Além disso, deve permitir que um usuário seja tanto um paciente quanto um fonoaudiólogo, ou seja, permitir que um fonoaudiólogo possua uma lista de seus pacientes, os quais o próprio SLP realiza a PE. Também deve permitir que um

paciente realize uma autoavaliação, permitindo que o paciente possa realizar uma triagem fonológica fornecendo à ele os resultados, e também, realizar um estudo em conjunto com a área da fonoaudiologia, para se determinar quando um resultado deve recomendar a busca por apoio fonológico.

Deve ser desenvolvido também um sistema para o reconhecimento da voz do usuário quando ele for o paciente. Isto é, deve coletar a assinatura de voz do usuário, para que no futuro, o aplicativo tenha a capacidade de reconhecer o usuário que está sendo avaliado e até mesmo, permitir o *login* do usuário através de sua voz. Além disso, é preciso buscar formas de realizar a filtragem do áudio coletado e também determinar sua qualidade de forma que seja possível evitar que um áudio de má qualidade seja enviado para análise, evitando assim o produção de um resultado incorreto em decorrência de um dado de baixa qualidade.

REFERÊNCIAS

- BETTINI, C. et al. A survey of context modelling and reasoning techniques. **Pervasive and Mobile Computing**, [S.l.], v.6, n.2, p.161–180, 2010.
- CERON, M. I. **INSTRUMENTO DE AVALIAÇÃO FONOLÓGICA (INFONO)**: desenvolvimento e estudos psicométricos. 2015. 148p. Doutorado em Fonoaudiologia — Universidade Federal de Santa Maria, Santa Maria.
- CHUCHUCA-MÉNDEZ, F. et al. An educative environment based on ontologies and e-learning for training on design of speech-language therapy plans for children with disabilities and communication disorders. In: CIENCIAS DE LA INFORMÁTICA Y DESARROLLOS DE INVESTIGACIÓN (CACIDI), IEEE CONGRESO ARGENTINO DE. **Anais...** [S.l.: s.n.], 2016. p.1–6.
- ENDSLEY, M. R. Toward a theory of situation awareness in dynamic systems. **Human factors**, [S.l.], v.37, n.1, p.32–64, 1995.
- FERRANTE C. BORSEL J. V., P. M. M. D. B. Análise dos processos fonológicos em crianças com desenvolvimento fonológico normal. **Revista da Sociedade Brasileira de Fonoaudiologia**, [S.l.], v.14, n.1, p.36–40, 2009.
- FRANCISCATTO, M. H. **SISTEMA DE RECOMENDAÇÃO PARA TRIAGEM DE DESSORDENS DOS SONS DA FALA INFANTIL BASEADO EM UM MODELO DE CONSCIÊNCIA DE SITUAÇÃO**. 2018. 91p. Mestrado em Computação — Universidade Federal de Santa Maria, Santa Maria.
- GALCERAN, L. B. **Evaluación fonológica del habla infantil**. Barcelona: Masson, 2005.
- GRUNWELL, P. The nature of phonological disability in children. **London: Academic Press**, [S.l.], v.10, n.2, p.479–484, 1981.
- GUEDES, G. T. **UML 2-Uma Abordagem Prática-1ª Edição**. [S.l.]: Novatec Editora, 2011.
- LOWE, R. J. **Fonologia - avaliação e intervenção**: aplicações na patologia da fala. Porto Alegre: Artes Médicas, 1996.

LYYTINEN, K.; YOO, Y. Ubiquitous computing. **Communications of the ACM**, [S.l.], v.45, n.12, p.63–96, 2002.

O. A. SCHIPOR M. DANUBIANU, I. U. S. P.; NESTOR, M. Distributed Intelligent System for Personalized Therapy of Speech Disorders. **Third International Multi-Conference on Computing in the Global Information Technology**, [S.l.], p.166–170, 2008.

PARNANDI, A. et al. Development of a Remote Therapy Tool for Childhood Apraxia of Speech. **ACM Transactions on Accessible Computing (TACCESS)**, [S.l.], v.7, n.3, p.10, 2015.

ROBLES-BYKBAEV, V. E. et al. An ontology-based expert system to generate therapy plans for children with disabilities and communication disorders. In: ECUADOR TECHNICAL CHAPTERS MEETING (ETCM), IEEE. **Anais...** [S.l.: s.n.], 2016. v.1, p.6.

WERTZNER H. F., A. L. . T. S. S. Gravidade do distúrbio fonológico: julgamento perceptivo e porcentagem de consoantes corretas. **Pró-Fono: revista de Atualização Científica**, [S.l.], v.17, n.11, p.185–194, 2005.

WERTZNER H. F. PAGAN L. O., G. D. E. S. . P. A. C. C. S. Características fonológicas de crianças com transtorno fonológico com e sem histórico de otite média. **Revista da Sociedade Brasileira de Fonoaudiologia**, [S.l.], v.12, n.1, p.41–47, 2007.

YE, J.; DOBSON, S.; MCKEEVER, S. Situation identification techniques in pervasive computing: a review. **Pervasive and mobile computing**, [S.l.], v.8, n.1, p.36–66, 2012.