

PAULA BRANDÃO FURLAN

**Desenvolvimento de Aplicativo controlado por  
Sinais Eletroencefalográficos para auxílio na  
Comunicação de pessoas com Deficiências na  
Fala**

Manaus - Amazonas

Junho - 2016

PAULA BRANDÃO FURLAN

**Desenvolvimento de Aplicativo controlado por Sinais Eletroencefalográficos para auxílio na Comunicação de pessoas com Deficiências na Fala**

Trabalho de Conclusão de Curso submetido à Coordenação do curso de Engenharia de Controle e Automação da Universidade do Estado do Amazonas como parte dos requisitos necessários para a obtenção do grau de Engenheiro de Controle e Automação em conformidade com as normas ABNT.

Universidade do Estado do Amazonas – UEA  
Escola Superior de Tecnologia - EST  
Engenharia de Controle e Automação

Orientador Prof. Me. Almir Kimura Junior

Manaus - Amazonas  
Junho - 2016

PAULA BRANDÃO FURLAN

**Desenvolvimento de Aplicativo controlado por Sinais Eletroencefalográficos para auxílio na Comunicação de pessoas com Deficiências na Fala**

Trabalho de Conclusão de Curso submetido à Coordenação do curso de Engenharia de Controle e Automação da Universidade do Estado do Amazonas como parte dos requisitos necessários para a obtenção do grau de Engenheiro de Controle e Automação em conformidade com as normas ABNT.

Aprovado em 16 de Junho de 2016:

BANCA EXAMINADORA

---

**Prof. Me. Almir Kimura Junior**  
Orientador

---

**Prof. Me. Charles Luiz Silva de Melo**  
Presidente

---

**Prof. Dr. João Evangelista Neto**  
Membro

---

**Prof. Ma. Ingrid Sammyne Gadelha Figueiredo**  
Membro

Manaus - Amazonas  
Junho - 2016

PAULA BRANDÃO FURLAN

Desenvolvimento de Aplicativo controlado por Sinais Eletroencefalográficos para auxílio na Comunicação de pessoas com Deficiências na Fala/ PAULA BRANDÃO FURLAN. – Manaus - Amazonas, Junho - 2016

80 p. : il. (algumas color.).

Orientador Prof. Me. Almir Kimura Junior

Trabalho Acadêmico – Universidade do Estado do Amazonas – UEA

Escola Superior de Tecnologia - EST

Engenharia de Controle e Automação, Junho - 2016.

1. Eletroencefalografia. 2. Aplicativo. 3. Esclerose Lateral Amiotrófica 4. Afonia. I. Prof. Me. Almir Kimura Junior. II. Universidade do Estado do Amazonas. III. Escola Superior de Tecnologia. IV. Desenvolvimento de Aplicativo controlado por Sinais Eletroencefalográficos para auxílio na Comunicação de pessoas com Deficiências na Fala

À minha família, meus amigos  
e professores pelo incentivo  
para realização deste  
trabalho.

# Agradecimentos

Agradeço primeiramente aos meus pais por fazerem de mim o que sou hoje e pelo apoio durante toda a minha trajetória nesta universidade. À minha irmã, pelo companheirismo e pelos conselhos dados. Aos meus amigos, por não me deixarem desanimar ao longo do caminho. Ao professor Me. Almir Kimura Junior pela orientação e suporte dados não somente a esse trabalho, mas em tantas outras atividades acadêmicas realizadas. Ao Grupo de Robótica e Automação da Escola Superior de Tecnologia e as pessoas que dele fazem parte, pela oportunidade de adquirir muitos dos conhecimentos aplicados neste trabalho. À coordenação do curso de Engenharia de Controle e Automação e aos professores do curso, pelos conhecimentos e experiências passados e, em especial ao professor Me. Charles Luiz Silva de Melo por estar sempre pronto a me auxiliar.

*“Pense antes de falar. Leia antes de pensar.”*  
*(Fran Lebowitz)*

# Resumo

A Esclerose Lateral Amiotrófica (ELA) é uma doença neurodegenerativa que atinge entre 0,6 e 2,6 pessoas por 100.000 habitantes no mundo. Ela envolve a degeneração do sistema motor, sem afetar as funções cerebrais na maioria dos casos e chega a afetar as funções vocais, o que demanda a adaptação de sistemas de comunicações para evitar que o paciente venha a desenvolver a Síndrome do Encarceramento, a condição de prisioneiro do próprio corpo. A afonia é a condição do indivíduo incapaz de falar e é uma deficiência que afeta cerca de 3,59% de pessoas no Brasil. Um grande problema enfrentado pelas pessoas que sofrem com ela é a pouca difusão do ensino da Linguagem Brasileira de Sinais (LIBRAS), o qual muito se concentra nas pessoas acometidas pela deficiência e seus familiares. Este trabalho apresenta o projeto e desenvolvimento de um sistema para aquisição de sinais de Eletroencefalograma, utilizando um eletrodo seco da *Neurosky, Think Gear ASIC Module* conectado a um dispositivo móvel por meio de um módulo de comunicação serial via *Bluetooth*, para análise por meio de um algoritmo inteligente escrito em Linguagem Java a fim de emitir mensagens sonoras e envio de *e-mails* por meio de um aplicativo para *Android*. O estudo foi conduzido no laboratório da Universidade do Estado do Amazonas. Os resultados obtidos com os testes mostraram que, após o sensor estar conectado ao aplicativo, o tempo necessário para aprendizado não foi alto e o tempo para a escrita de frases inteiras e envio de *e-mails* diminuiu com o uso. Com base nos resultados, constatou-se que o sistema pode ser usado como um meio de comunicação eficaz.

**Palavras-chaves:** Eletroencefalografia. Aplicativo. Esclerose Lateral Amiotrófica. Afonia.

# Abstract

The Amyotrophic Lateral Sclerosis is a neurodegenerative disease that reaches between 0.6 and 2.6 people per 100,000 inhabitants in the world. It involves the degeneration of the motor system, without affecting the brain functions in most cases and comes to affect the vocal functions, what demand the adaptation of communication systems to avoid that the patient comes to develop the Locked-in Syndrome, the condition of prisoner in his own body. The Aphony is the condition of the individual unable to speak and is a disability that reaches about 3.59% of people in the Brazil. A big problem faced by people who have it, is the little spread of sign?s language teaching, which is concentrated in the people with the disability and their relatives. This project presents the design and development of a system to acquire signals of Electroencephalogram, using a dry electrode from Neurosky, Think Gear ASIC Module, connected to a mobile device by means of a serial communication module via bluetooth, to be analyzed by an intelligent algorithm wrote in Java Language in order to vocalize words and send e-mails by means of an app for Android. The study was conducted and tested in the Lab of the Universidade do Estado do Amazonas. The results obtained with tests showed that after having the sensor connected to the app, the time needed for learning was not high and the time to write full sentences and send e-mails decreased with the usage. Based on the results, it was found that the system can be used as an effective communication system.

**Key-words:** Electroencephalography. App. Amyotrophic Lateral Sclerosis. Aphony.

# Lista de ilustrações

Figura 1 – Bandas de Frequência de um sinal de EEG . . . . .	19
Figura 2 – Sistema Internacional 10/20 . . . . .	19
Figura 3 – Extração do potencial P300 . . . . .	21
Figura 4 – EEG durante a performance de quatro diferentes tarefas com piscar dos olhos ao final de cada uma (linha vertical), vista com o <i>software TruScanExplorer</i> . . . . .	22
Figura 5 – Eletrodos de Ag e de AgCl para aquisição de sinais Eletroencefalográficos	23
Figura 6 – Eletrodos secos para aquisição de sinais Eletroencefalográficos . . . . .	24
Figura 7 – Exemplo de sinal contínuo no Tempo . . . . .	25
Figura 8 – Exemplo de sinal de Tempo Discreto . . . . .	25
Figura 9 – Exemplo de módulo de comunicação serial via <i>Bluetooth</i> . . . . .	27
Figura 10 – Kit para iniciantes TGAM . . . . .	31
Figura 11 – Esquemático do circuito para aquisição de sinais de EEG . . . . .	32
Figura 12 – Bandagem para a cabeça adaptada com o circuito de aquisição de sinais de EEG . . . . .	33
Figura 13 – Resposta obtida após a modificação do TGAM via <i>software</i> . . . . .	34
Figura 14 – <i>Tablet</i> escolhido para o desenvolvimento e testes do aplicativo . . . . .	35
Figura 15 – Tela de Login do aplicativo "Mind Talks" . . . . .	36
Figura 16 – Tela de Cadastro de Novos Usuários do aplicativo "Mind Talks" . . . . .	37
Figura 17 – Tela Principal do aplicativo "Mind Talks" . . . . .	38
Figura 18 – Diagrama de Classes do aplicativo "Mind Talks" . . . . .	39
Figura 19 – DER do aplicativo "Mind Talks" . . . . .	40
Figura 20 – Lógica para o botão "Confirmar" da tela de Login . . . . .	41
Figura 21 – Lógica para o botão "Confirmar" da tela de Cadastro de Novos Usuários	42
Figura 22 – Lógica para quando o usuário pressiona qualquer caractere da tela principal . . . . .	43
Figura 23 – Lógica para o método Criar Sugestões . . . . .	44
Figura 24 – Lógica para quando o usuário apertar o botão "Ok" da tela principal .	45
Figura 25 – Lógica para quando o usuário pressiona o botão "Enviar e-mail" da tela principal . . . . .	46
Figura 26 – Lógica para quando o usuário pressiona o botão "Enter" da tela principal	47
Figura 27 – Mensagem de Alerta para o usuário escolher se deseja ou não conectar o aplicativo ao TGAM . . . . .	50
Figura 28 – Resultado do teste de caracteres combinado ao teste de Sugestões . . .	52
Figura 29 – Tela principal no estado de Escrever destinatário do <i>e-mail</i> . . . . .	53
Figura 30 – Resultado do teste de envio de <i>e-mail</i> . . . . .	53

Figura 31 – Tela do aplicativo de teste para aquisição dos sinais do TGAM . . . . .	54
Figura 32 – Tela principal no estado Cores da Interface Cerebral . . . . .	55
Figura 33 – Tela principal no estado Letras da Interface Cerebral . . . . .	55
Figura 34 – Tela principal no estado Botões da Interface Cerebral . . . . .	56

# Lista de abreviaturas e siglas

ELA	Esclerose Lateral Amiotrófica
LIBRAS	Linguagem Brasileira de Sinais
ICM	Interface Cérebro-Máquina
IDE	Ambiente Integrado de Desenvolvimento
EEG	Eletroencefalografia
ERP	Potencial Relacionado a Evento
SSVEP	Potencial Visualmente Evocado de Estado Estacionário
ADC	Conversor Analógico-Digital
API	Interface de Programação de Aplicativos
OO	Orientada a Objetos
OS	Sistema Operacional
XML	Linguagem de Marcação Estendida
TGAM	<i>Think Gear Asic Module</i>
EST	Escola Superior de Tecnologia
UEA	Universidade do Estado do Amazonas
AAC	Comunicação Suplementar e Alternativa
UML	Linguagem de Modelagem Unificada
DER	Diagrama Entidade Relacionamento
DAO	Objeto de Acesso aos Dados
SQL	Linguagem de Consulta Estruturada
BD	Banco de Dados

# Sumário

	<b>Lista de ilustrações</b> . . . . .	<b>9</b>
	<b>Sumário</b> . . . . .	<b>12</b>
<b>1</b>	<b>INTRODUÇÃO</b> . . . . .	<b>14</b>
1.1	<b>Formulação do Problema</b> . . . . .	<b>14</b>
1.2	<b>Hipótese</b> . . . . .	<b>15</b>
1.3	<b>Justificativa</b> . . . . .	<b>15</b>
1.4	<b>Objetivos</b> . . . . .	<b>16</b>
1.4.1	Objetivo Geral . . . . .	16
1.5	<b>Organização do Trabalho</b> . . . . .	<b>16</b>
<b>2</b>	<b>REFERENCIAL TEÓRICO</b> . . . . .	<b>18</b>
2.1	<b>Eletroencefalografia</b> . . . . .	<b>18</b>
2.1.1	Padrão para disposição dos eletrodos no escalpo cerebral . . . . .	19
2.1.2	Extração de características de sinais de EEG . . . . .	20
2.1.3	Padrões de tradução de sinais de EEG . . . . .	20
2.1.4	Eletrodos secos versus eletrodos convencionais . . . . .	22
2.2	<b>Conversores Analógico-Digitais</b> . . . . .	<b>23</b>
2.3	<b>Sinais Discretos</b> . . . . .	<b>24</b>
2.3.1	Resposta em frequência de sistemas . . . . .	25
2.4	<b>Comunicação serial via <i>Bluetooth</i></b> . . . . .	<b>26</b>
2.5	<b>Linguagem de Programação Java</b> . . . . .	<b>27</b>
2.6	<b>Sistema Operacional <i>Android</i></b> . . . . .	<b>28</b>
<b>3</b>	<b>MATERIAIS E MÉTODOS</b> . . . . .	<b>30</b>
3.1	<b>Dispositivo para aquisição dos sinais de EEG</b> . . . . .	<b>30</b>
3.1.1	Montagem do kit . . . . .	30
3.1.2	Aquisição dos primeiros sinais de EEG . . . . .	33
3.2	<b>Desenvolvimento do Aplicativo para <i>Android</i></b> . . . . .	<b>34</b>
3.2.1	Desenvolvimento dos <i>Layouts</i> das telas . . . . .	35
3.2.2	Modelagem do aplicativo e conexão com o Banco de Dados . . . . .	38
3.2.3	Criação das <i>Activities</i> . . . . .	40
3.3	<b>Desenvolvimento do algoritmo para interpretar os sinais adquiridos</b> .	<b>48</b>
3.4	<b>Integração do aplicativo com o algoritmo para interpretação dos sinais</b> . . . . .	<b>49</b>

---

<b>4</b>	<b>RESULTADOS E DISCUSSÃO</b>	<b>52</b>
<b>4.1</b>	<b>Testes do aplicativo Mind Talks</b>	<b>52</b>
<b>4.2</b>	<b>Testes do algoritmo para tradução dos sinais do TGAM</b>	<b>54</b>
<b>4.3</b>	<b>Testes da Interface Cerebral sem o TGAM conectado</b>	<b>54</b>
<b>4.4</b>	<b>Testes da Interface Cerebral com o TGAM conectado</b>	<b>56</b>
<b>4.5</b>	<b>Testes finais com o sistema integrado</b>	<b>56</b>
<b>4.6</b>	<b>Discussão acerca dos testes realizados</b>	<b>57</b>
<b>5</b>	<b>CONSIDERAÇÕES FINAIS</b>	<b>58</b>
<b>5.1</b>	<b>Trabalhos Futuros</b>	<b>58</b>
	<b>REFERÊNCIAS</b>	<b>60</b>

# 1 Introdução

A Esclerose Lateral Amiotrófica (ELA) é uma das principais doenças neurodegenerativas, ao lado das doenças de Alzheimer e Parkinson. Ela possui uma incidência de 0,6 a 2,6 por 100.000 habitantes de acordo com (CRONIN; HARDIMAN; TRAYNOR, 2007, traduzido pelo autor). Devido a se tratar de "um distúrbio progressivo que envolve a degeneração do sistema motor em vários níveis" (MITCHELL; BORASIO, 2007, traduzido pelo autor) e em certo ponto afetar as funções vocais e respiratórias, faz-se necessário a criação de diversos mecanismos para adaptação direcionados aos pacientes com a ELA.

Um grande problema enfrentado por pessoas que sofrem com a ELA é o desenvolvimento da Síndrome do Encarceramento, a condição de "prisioneiro no próprio corpo". Diversas soluções são criadas, porém não se mostram viáveis à toda a população acometida pela ELA. Uma delas é a criada para o célebre paciente Stephen Hawking, o *Equalizer*, que permite a escrita de frases com a seleção de palavras com o toque da mão e a reprodução delas por um sintetizador de voz em um *tablet*, de acordo com (CALADO, 2010).

Mudez ou afonia é a condição do indivíduo incapaz de falar. Entre a condição normal e a mudez existe uma gama de distúrbios da fala (afasias), geralmente decorrentes de transtornos do sistema nervoso central. De acordo com o Censo de 2000, cerca de 3,59% da população brasileira sofre de mudez causada pela surdez. Um grande problema enfrentado pelas pessoas com essa deficiência é a pouca difusão do ensino da Linguagem Brasileira de Sinais (LIBRAS), o qual muito se concentra nas pessoas acometidas pela deficiência e seus familiares. Dessa forma, a comunicação pode se tornar de certa forma restrita para essas pessoas.

## 1.1 Formulação do Problema

O maior desafio encontrado pelos pesquisadores para lidar com a comunicação de pessoas com a ELA é que os sistemas criados devem ser sempre readaptados levando em consideração a crescente degradação na condição motora dos pacientes. No caso do célebre Stephen Hawking, o nervo motor da mão antes utilizado, com o tempo sofreu degradação e teve que ser substituído pelo músculo da sua bochecha, cujo movimento é detectado por um *switch* acoplado em seus óculos, como visto em (HAWKING, , traduzido pelo autor). Outro grande desafio para as pessoas com ELA, é o alto custo e a singularidade dada ao desenvolvimento desses sistemas, que muitas vezes são focados em pacientes específicos e não se aplicam a todos.

O *hardware* utilizado nos sistemas criados para as pessoas com paralisia e deficiência na fala ainda não se encontram amplamente disponibilizados a uma grande parcela da população. Além disso, os sensores para obter sinais de Eletroencefalografia (EEG) utili-

zados em Interfaces Cérebro-Máquina (ICM), como em (BARBOSA, 2012), apresentam alto custo e não são adequados para uso contínuo.

Outro ponto a ser levantado é como tornar a tecnologia disponível para ser utilizada também por pessoas que sofrem de mudez ou afonia, levando em consideração que o *hardware* utilizado pode não ser adequado a essas pessoas, como é para as que sofrem de paralisia.

## 1.2 Hipótese

Devido à popularização dos aplicativos para dispositivos móveis e a grande variedade de ferramentas disponibilizadas para o seu desenvolvimento, como o poderoso Ambiente Integrado de Desenvolvimento (IDE) *Android Studio*, eles se tornam uma poderosa plataforma para a criação de alternativas de baixo custo e maior acessibilidade para a concepção de sistemas de comunicação.

Aliando essa tecnologia com os recentes sensores de *biofeedback*, que utilizam eletrodos secos, desenvolvidos inicialmente para uso em jogos de computador, mas que tiveram sua eficácia como medidores de ondas cerebrais confiáveis comprovada por pesquisas na área de ICMS, como visto em (APEKSHA; PRATHIBHA, 2015, traduzido pelo autor), pode-se chegar a um sistema para assistir na comunicação de pessoas que sofrem de ELA e das que possuem mudez ou afonia de forma acessível para ambas e com baixo custo.

## 1.3 Justificativa

Os métodos de aquisição de sinais de (EEG) usados atualmente são eficientes para a concepção de ICMS, porém muitas vezes se mostram incômodos pelo uso de eletrodos que demandam uma limpeza inicial com pasta abrasiva do local de aplicação e de gel eletrolítico para fixação, de acordo com (LOPEZ-GORDO; SANCHEZ-MORILLO; VALLE, 2014, traduzido pelo autor), o qual recomenda o uso de eletrodos secos.

Outro problema encontrado no método tradicional de aquisição de sinais é a necessidade dos eletrodos estarem conectados a um computador por meio de cabos, o que limita bastante a mobilidade do usuário, como visto em (LAZZARI, 2013). Para resolver isso, uma alternativa é utilizar um módulo de comunicação serial via *Bluetooth* para a transmissão dos sinais.

Outro ponto importante é o fato de o dispositivo móvel ser portátil e fazer parte da vida de cerca de 86% da população brasileira, de acordo com (ANATEL, ). Dessa forma, o desenvolvimento de um sistema em formato de aplicativo para esses dispositivos será facilmente difundido para a maioria da população que sofre com a ELA ou com a mudez ou afonia.

## 1.4 Objetivos

### 1.4.1 Objetivo Geral

Desenvolver um sistema para aquisição de sinais de EEG, utilizando um eletrodo seco fixado em uma bandagem para a cabeça, conectado a um dispositivo móvel via módulo de comunicação serial por *Bluetooth*, para filtragem, extração de características e análise e interpretação do sinal, por meio de um algoritmo de controle inteligente, para fins de emissão de mensagens por meio de um aplicativo para dispositivos móveis. De forma mais específica, deve-se:

- Adaptar um sensor medidor de ondas cerebrais a uma bandagem para a cabeça, obter sinais e transmiti-los com integridade, via *Bluetooth*;
- Desenvolver o aplicativo utilizando a IDE *Android Studio*, de distribuição gratuita;
- Fazer a comunicação do sensor com o dispositivo móvel e interpretar os sinais por meio de algoritmo desenvolvido em Linguagem Java;
- Fazer a integração do aplicativo com o algoritmo para interpretação dos sinais e realizar testes de usabilidade e estresse do sistema final.

## 1.5 Organização do Trabalho

Este trabalho está dividido em cinco capítulos para seu melhor entendimento. A subdivisão em capítulos é a seguinte:

- Introdução (Capítulo 1)
- Referencial Teórico (Capítulo 2)
- Materiais e Métodos (Capítulo 3);
- Resultados e Discussão (Capítulo 4);
- Considerações Finais (Capítulo 5);

No Capítulo 1 se dá uma breve introdução à temática abordada por esse trabalho e é apresentada a problemática, a hipótese a ser provada, os objetivos gerais e específicos e a justificativa para o mesmo.

O Capítulo 2 traz uma revisão da bibliografia necessária para o desenvolvimento deste trabalho.

O Capítulo 3 descreve o processo de desenvolvimento do sistema de comunicação e uma visão geral sobre o mesmo é dada, com uma explicação sobre o desenvolvimento de

cada uma de suas componentes, como a adaptação do sensor para obter os sinais de EEG, além do desenvolvimento do aplicativo.

O Capítulo 4 apresenta os resultados obtidos, dos quais se faz uma análise e avaliação da performance dos métodos propostos.

Por último, o Capítulo 5 apresenta as conclusões feitas deste trabalho, assim como propostas para trabalhos futuros.

## 2 Referencial Teórico

COM o objetivo de desenvolver um sistema para aquisição de sinais de EEG a fim de fornecer comunicação por meio de um dispositivo móvel, faz-se necessário conhecer os principais conceitos utilizados para a sua concepção.

Devido ao sistema ser baseado na obtenção de sinais eletroencefalográficos, é preciso conhecer o conceito de EEG e como se dá a obtenção dos mesmos.

É preciso também expor qual a melhor técnica para transmissão dos sinais para dispositivos móveis e como será feito o tratamento deles para a extração de características.

Por fim, também é necessário falar sobre a plataforma que será utilizada para o desenvolvimento do aplicativo e a Linguagem de Programação que será utilizada para isso.

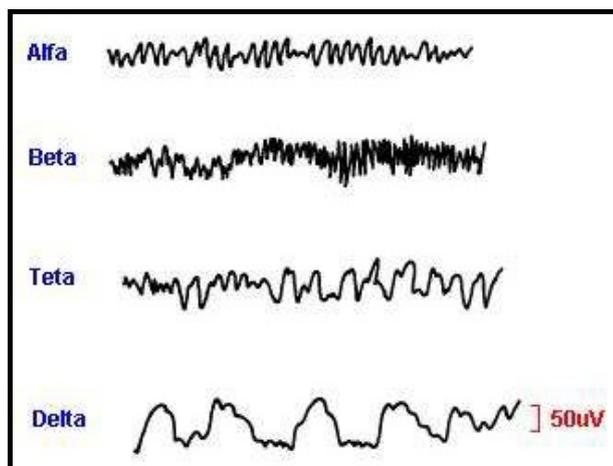
É importante ressaltar que o conceito de EEG é bastante utilizado no campo da Medicina, mas o estudo de seu uso em aplicações no campo da Engenharia só teve início significativo há pouco mais de uma década e já se tornou bastante explorado. Dessa forma, deve-se observar que o *hardware* utilizado para aquisição desses sinais possui rápida evolução, com novas tecnologias surgindo todos os anos.

### 2.1 Eletroencefalografia

De acordo com (BARBOSA, 2012), "O eletroencefalograma é o registro eletrofisiológico da atividade elétrica pós-sináptica de uma grande quantidade de neurônios piramidais do córtex cerebral". Esse registro é usado para medir atividade cerebral de forma não invasiva, sendo o mais predominante para aquisição de sinais utilizados em ICMs, de acordo com (HE, 2005, traduzido pelo autor). Ele é bastante utilizado no campo da Medicina, mas o estudo de seu uso em aplicações no campo da Engenharia só teve início significativo há pouco mais de uma década e já se tornou bastante explorado.

Para (KANDA, 2012), o sinal de EEG é a resultante do potencial de campo projetado para o escalpo cerebral, pode ter aparência sinusoidal e sua frequência tem banda entre 1 e 70 Hz, que se subdivide em outras, a teta (0,5 a 3,9 Hz), a delta (4,0 a 7,9 Hz), a alfa (8,0 a 12,5 Hz) e a beta (> 13 Hz), como visto na Figura 1.

Figura 1 – Bandas de Frequência de um sinal de EEG

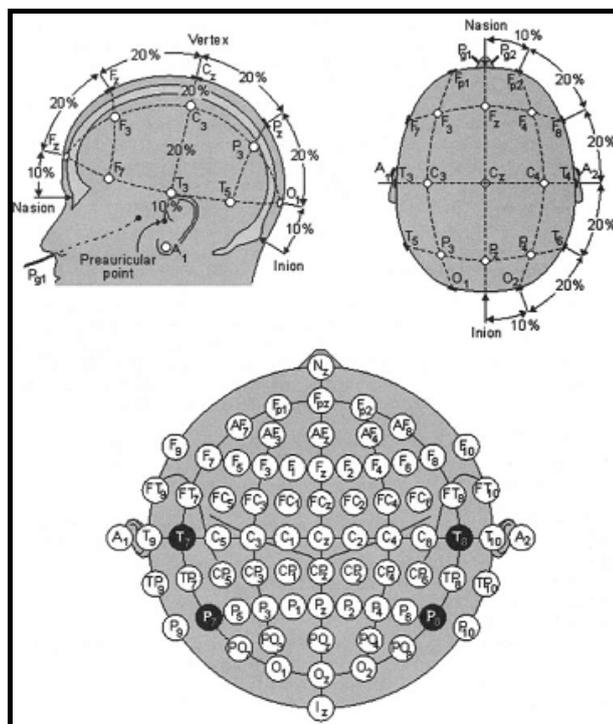


Fonte: (ADUR, 2008)

### 2.1.1 Padrão para disposição dos eletrodos no escalpo cerebral

De acordo com (HE, 2005, traduzido pelo autor), muitas ICMS baseadas em EEG utilizam a estratégia de fixação dos eletrodos sugerida pelo Sistema Internacional 10/20, o qual está ilustrado na Figura 2. Conforme (HE, 2005, traduzido pelo autor), para obter uma melhor resolução espacial, é muito comum usar uma variação do sistema 10/20, a qual preenche os espaços entre os eletrodos do mesmo com outros.

Figura 2 – Sistema Internacional 10/20



Fonte: (HE, 2005)

Ainda de acordo com (HE, 2005, traduzido pelo autor), o Sistema Internacional 10/20

garante uma cobertura ampla de todas as partes da cabeça. A letra em cada eletrodo identifica o lobo subcranial em particular, sendo FP para lobo pré-frontal, F para lobo frontal, C para lobo central, P para lobo parietal e O para lobo occipital, e o número da segunda letra identifica a localização hemisférica, com Z denotando a linha zero que se refere a um eletrodo fixado ao longo da linha média do cérebro, números pares representando o hemisfério direito e números ímpares representando o hemisfério esquerdo, todos eles em ordem crescente a partir da linha média.

De acordo com (LAZZARI, 2013):

Uma das correspondências mais conhecidas é a contralateral esquerda-direita. Ou seja, o hemisfério esquerdo do cérebro é responsável por controlar o lado direito, enquanto o hemisfério direito controla o lado esquerdo do corpo, respectivamente. Isto é válido especialmente para o controle dos membros superiores.

De acordo com (LAZZARI, 2013), essa correspondência não é tão observada nos membros inferiores, devido ao fato dos mesmos terem sua região de controle localizada na área central do córtex, tipicamente no local em torno de Cz.

### 2.1.2 Extração de características de sinais de EEG

A extração de características é um dos módulos fundamentais para reconhecimento de padrões e uma tarefa importante na concepção de ICMS, de acordo com (SORIANO, 2014).

O objetivo da extração de características é obter aspectos fisiológicos específicos do sistema nervoso ao longo do tempo, segundo (HE, 2005, traduzido pelo autor). De acordo com (ROSA, 2009), o sinal bruto de EEG apresenta ruídos de alta frequência, os quais distorcem completamente o mesmo, e para resolver isso é importante a aplicação de um filtro para eliminar altas frequências, além de também sofrer interferência das frequências de rede, 50 Hz para padrão europeu e 60 Hz para padrão americano, então se deve aplicar um filtro *Notch* que elimine as mesmas.

Após isso, faz-se a seleção de características relevantes, para isso utilizando métodos com foco em maximizar a relação sinal-para-ruído, e o último passo é o uso de técnicas de tradução para transformar as características relevantes em um estado para a tomada de uma decisão de acionamento, segundo (HE, 2005, traduzido pelo autor).

### 2.1.3 Padrões de tradução de sinais de EEG

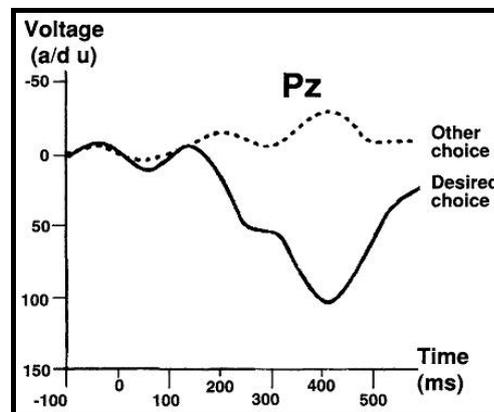
Os padrões para tradução das características extraídas, de acordo com (BARBOSA, 2012), podem ser agrupados em categorias, de acordo com o processo fisiológico subjacente utilizado, e são elas: Potenciais Relacionados a Eventos (ERPs), ritmos Mu e beta, além da aquisição de sinais relacionados ao fechamento dos olhos, entre outros.

Os ERPs de acordo com (HE, 2005) são respostas cerebrais que ocorrem após um tempo fixado depois de eventos particulares externos ou internos. Os ERPs podem ser classificados em Exógenos ou Endógenos, dependendo do tipo de evento a que são relacionados.

Um exemplo de ERP exógeno é o Potencial Visualmente Evocado de Estado Estacionário (SSVEP), o qual segundo (HE, 2005) ocorre quando o usuário foca visualmente em um de dois objetos em uma tela que piscam com diferentes frequências. O componente SSVEP é amplificado quando o usuário muda seu foco para o outro objeto e então retorna a um valor de base e continua mudando de acordo com a mudança de foco do usuário.

O P300 é um ERP do tipo endógeno que ocorre de acordo com o "paradigma *oddball*" de (DONCHIN; COLES, 1988) e (DONCHIN; SPENCER; WIJESINGHE, 2000). De acordo com (HE, 2005), os usuários são submetidos a dois eventos distintos, sendo que um deles será raramente mostrado, o mesmo então recebe uma tarefa que não pode ser completada sem que ele tenha a categorizado nos dois eventos. Quando o evento raro é mostrado, isso faz com que o P300 seja extraído, o qual é uma larga onda positiva que ocorre aproximadamente 300 ms depois que ocorreu o evento raro e a amplitude do potencial é maior tanto quanto mais raro for o acontecimento de tal evento. A aquisição dele pode ser vista na Figura 3.

Figura 3 – Extração do potencial P300



Fonte: (HE, 2005)

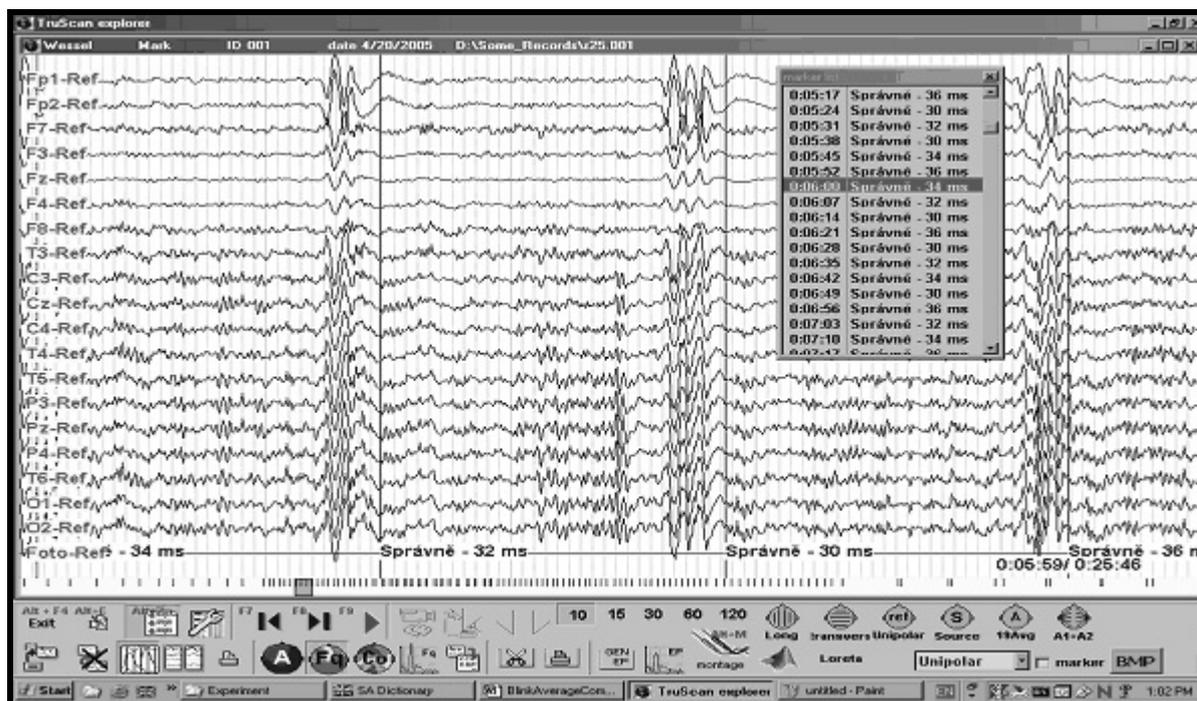
De acordo com (HE, 2005, traduzido pelo autor), a atividade oscilatória eletroencefalográfica se deve a uma complexa rede de neurônios que criam laços de realimentação e duas oscilações de interesse, que ocorrem nos períodos de marcha lenta e de repouso, são a do ritmo Mu, que ocorre na frequência de 10 a 12 Hz, e a do ritmo central beta, a qual ocorre na frequência de 14 a 18 Hz. A amplitude das oscilações diminui com o aumento da frequência, de acordo com (HE, 2005, traduzido pelo autor). Segundo (BARBOSA, 2012), o ritmo Mu está relacionado principalmente com o córtex motor e é melhor captado sobre o giro pré-central, C3 e C4 no sistema internacional 10/20.

De acordo com (SANTOS; SILVA; STEDILE, 2014), o sinal de EEG está relacionado

ao nível de consciência da pessoa, quando a atividade aumenta o sinal possui maior frequência e menor amplitude, quando o usuário fecha os olhos as ondas da banda alfa aparecem e quando a pessoa adormece a frequência diminui ainda mais, atingindo a banda delta e teta. De acordo com (BARBOSA, 2012), as ondas alfa são captadas no lobo occipital quando a pessoa fecha os olhos e não se concentra em nenhum pensamento profundo.

O piscar dos olhos é geralmente uma característica indesejada encontrada nas medições de EEG, que ocorrem por causa da proximidade dos músculos das pálpebras com os sensores nas posições Fp1 e Fp2, de acordo com (HAAK et al., 2009, traduzido pelo autor). Porém essa característica não oferece influência apenas nas proximidades do lobo frontal e como pode ser visto em (MANOILOV, 2006, traduzido pelo autor), foi escolhido o eletrodo na posição C3 e observou-se que quando ocorria uma piscada, havia um aumento no Espectro de Poder do sinal na banda de frequência Teta, a qual varia de 0,5 – 3Hz. Na Figura 4 pode ser vista a influência do piscar dos olhos em medições de EEG de eletrodos de várias posições no sistema internacional 10/20.

Figura 4 – EEG durante a performance de quatro diferentes tarefas com piscar dos olhos ao final de cada uma (linha vertical), vista com o *software TruScanExplorer*



Fonte: (MANOILOV, 2006)

#### 2.1.4 Eletrodos secos versus eletrodos convencionais

De acordo com (LOPEZ-GORDO; SANCHEZ-MORILLO; VALLE, 2014, traduzido pelo autor), para aquisição de sinais de EEG, a preparação é um processo trabalhoso, o qual começa com a localização dos sítios para montagem elétrica, depois essas áreas

são esfregadas com uma pasta abrasiva para diminuir a impedância da pele para entre  $5\text{k}\Omega$  e  $20\text{k}\Omega$ , então eletrodos de Ag ou AgCl, os quais podem ser vistos na Figura 5, são impregnados com um gel eletrolítico que facilita a transdução da corrente iônica para corrente elétrica. Ainda de acordo com (LOPEZ-GORDO; SANCHEZ-MORILLO; VALLE, 2014, traduzido pelo autor), esse trabalho requer pessoal com perícia no assunto, resulta em incômodo para o usuário, além de que o tempo para se chegar ao valor adequado de impedância, pode ser elevado e também o tempo para o gel eletrolítico deteriorar é de apenas cinco horas, o que faz com que esses eletrodos não sejam os ideais para medições de longo prazo.

Figura 5 – Eletrodos de Ag e de AgCl para aquisição de sinais Eletroencefalográficos



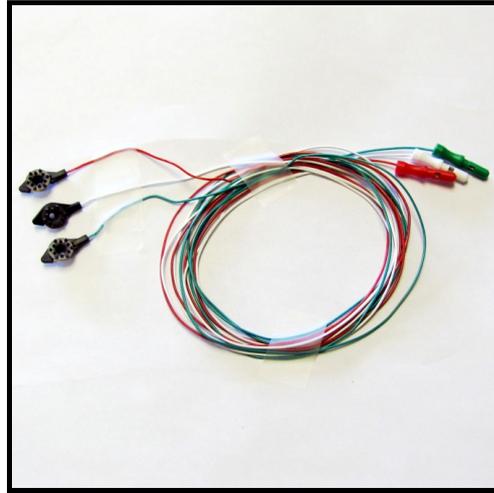
Fonte: Página da loja SPMédica

Graças a novos avanços em materiais e sistemas eletrônicos integrados, uma geração de eletrodos secos está sendo criada, de acordo com (LOPEZ-GORDO; SANCHEZ-MORILLO; VALLE, 2014, traduzido pelo autor), os quais podem ser vistos na Figura 6. Porém um grande problema para esses eletrodos é o cabelo, o qual pode causar perda de contato com o escalpo, segundo (LOPEZ-GORDO; SANCHEZ-MORILLO; VALLE, 2014, traduzido pelo autor), em vista disso alguns pesquisadores, ao dar atenção a esse problema resolveram evitar o contato com o escalpo, mas ao custo de um grande aumento na impedância. Em (HARLAND; CLARK; PRANCE, 2002), os autores conseguiram gravar sinais de EEG com eletrodos a uma distância de três milímetros do escalpo, nas posições P3 – O1 do sistema internacional 10/20, e solucionaram o problema da impedância com um amplificador de ultra-alta impedância.

## 2.2 Conversores Analógico-Digitais

O sinal eletroencefalográfico é uma grandeza analógica com amplitude que varia de 25 a  $100\ \mu\text{V}$ , de acordo com (ADUR, 2008). Para o processamento dessa grandeza por sistemas digitais, faz-se necessário a conversão da mesma para um conjunto de *bits*, o qual representa de forma adequada a informação contida no sinal analógico, processo esse chamado de Conversão Analógica-Digital, segundo (PEREIRA; PIZZATO; CARATI, 2010).

Figura 6 – Eletrodos secos para aquisição de sinais Eletroencefalográficos



Fonte: Página da empresa *Florida Research Instruments*

De acordo com (PEREIRA; PIZZATO; CARATI, 2010), quando se fala em Conversor Analógico-Digital (ADC), há várias características importantes a serem levadas em conta, entre elas:

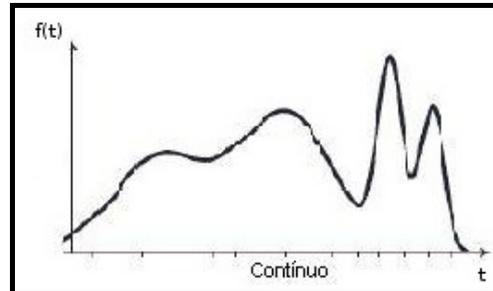
- Resolução, a qual é proporcional ao número de *bits* do ADC e representa o número de níveis de conversão do mesmo. Por exemplo, um ADC de 12 *bits* apresenta 212 ou 4096 níveis de conversão.
- Faixa de entrada, que é o intervalo de variação da amplitude do sinal a ser convertido. Por exemplo, uma entrada em tensão que pode variar de 0 a 5 V.
- Resolução de entrada, que é o menor valor na unidade da faixa de entrada que pode ser identificado pelo ADC. Por exemplo, um ADC que tem faixa de entrada de 0 a 5 V e resolução de 12 *bits* tem como resolução de entrada  $5/4096$  ou 0,0012207 V.
- Valores máximos admitidos de entrada, os quais são os máximos valores que não causarão danos elétricos ao aparelho. Por exemplo, um ADC que suporta entradas de  $-25$  V a  $+25$  V.
- Taxa de conversão do ADC, que é a frequência de conversão do sistema. Por exemplo, uma frequência de 40 kHz a 20 kHz.

## 2.3 Sinais Discretos

”Um sinal define a variação de uma grandeza física como função de uma ou mais variáveis e esta variação contém a informação que é importante”, (ROSA, 2009). Um sinal contínuo no tempo contém no mínimo uma variável independente que é contínua, pode ser representado por uma função no tempo que varia continuamente de  $-\infty$  a  $+\infty$

e é representado tradicionalmente por  $f(t)$ , como visto em (ROSA, 2009). Um exemplo de sinal contínuo pode ser visto na Figura 7.

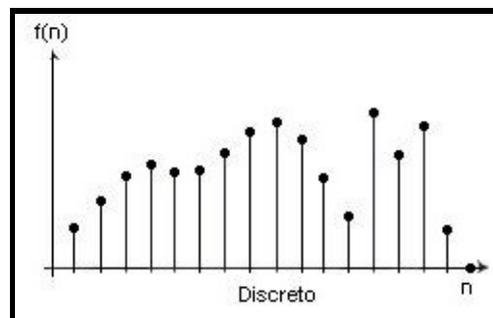
Figura 7 – Exemplo de sinal contínuo no Tempo



Fonte: (ROSA, 2009)

Um sinal discreto é uma função definida apenas em instantes discretos do tempo indefinida em qualquer outro instante, segundo (ROSA, 2009), e enquanto para o sinal contínuo tem-se o tempo  $t$ , para o discreto tem-se o  $nT$ , onde  $T$  é um intervalo de tempo conhecido e constante em segundos que se chama taxa de amostragem e  $n$  um valor inteiro que pode variar de  $-\infty$  a  $+\infty$ . O sinal discreto é geralmente representado por  $f(n)$ . Um exemplo de sinal discreto pode ser visto na Figura 8.

Figura 8 – Exemplo de sinal de Tempo Discreto



Fonte: (ROSA, 2009)

### 2.3.1 Resposta em frequência de sistemas

”Sistemas discretos são processos que atuam sobre os sinais discretos, recebendo um ou mais sinais discretos de entrada e fornecendo como saída um ou mais sinais discretos.” (ROSA, 2009). Para manter a integridade do sinal, deve-se obedecer ao teorema de *Nyquist*, que diz que ”a condição para um sinal íntegro é que a maior frequência no sinal de entrada de um amostrador analógico/digital de um sistema deve ser igual ou menor que a metade da taxa de amostragem” (ROSA, 2009).

De acordo com (ROSA, 2009), pode-se obter a resposta em frequência de um sistema discreto pelo método da aproximação Geométrica, para o qual deve-se conhecer todos os polos e zeros do sistema, utilizando a expressão geral na Equação 2.1, onde  $K$  é o ganho

proporcional do sistema,  $\omega$  é o ângulo a ser variado e  $z$  e  $p$  são os zeros e polos do sistema, respectivamente.

$$H(e^{j\omega T}) = \frac{\prod_{i=1}^N K(e^{j\omega T} - z_i)}{\prod_{k=1}^M (e^{j\omega T} - p_k)} \quad (2.1)$$

Simplificando a expressão geral e utilizando um sistema com dois polos e dois zeros, chega-se a uma fórmula simplificada, vista na Equação 2.2, onde  $U$  é a distância entre o zero e o ponto onde  $\omega T$  se encontra e  $V$  a distância entre o polo e esse ponto,  $\theta$  é o ângulo formado entre  $U$  e a referência  $X$  e  $\phi$  é o ângulo formado entre o  $V$  e essa referência  $X$ , como visto em (ROSA, 2009).

$$H(e^{j\omega T}) = \frac{KU_1\theta_1U_2\theta_2}{V_1\phi_1V_2\phi_2} \quad (2.2)$$

Ao simplificar a fórmula obtida para o cálculo da resposta em frequência para magnitude chega-se à Equação 2.3, e para o cálculo da resposta em frequência para fase chega-se à Equação 2.4. Dessa forma, torna-se simples a implementação deste cálculo em *software*, onde o mesmo se torna iterativo e o tempo de processamento definirá a resolução e a qualidade da resposta, segundo (ROSA, 2009).

$$H(e^{j\omega T}) = \frac{U_1U_2}{V_1V_2} \quad (2.3)$$

$$H(e^{j\omega T}) = \theta_1 + \theta_2 - (\phi_1 + \phi_2) \quad (2.4)$$

## 2.4 Comunicação serial via *Bluetooth*

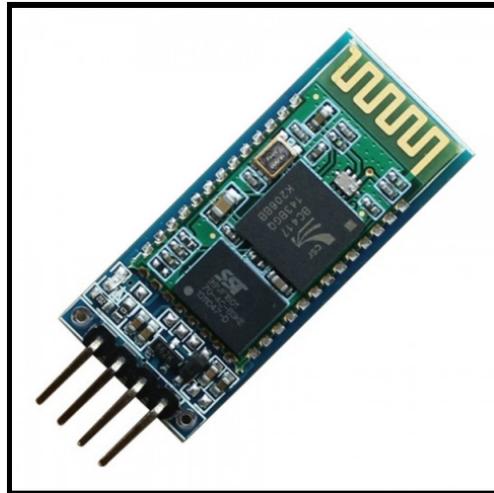
De acordo com (CRUZ; JUNIOR, 2013), o nome comunicação serial se deriva do fato de a transmissão dos *bits* ser feita em séries. Essa tecnologia utiliza um fio para transmitir os dados e um para receber. Inicialmente a velocidade de comunicação era 9600 *bits* por segundo, mas em 2009 essa velocidade já passava de 115 *kBytes* ou 920000 *bits* por segundo (CRUZ; JUNIOR, 2013).

De acordo com (SILVA, 2009), "A tecnologia *Bluetooth* nasceu no final da década de 90, sendo que, os primeiros produtos só apareceram comercialmente após o ano de 2000". Ainda de acordo com (SILVA, 2009), ela é uma comunicação sem fio de curto alcance, que visa fazer a comunicação entre dispositivos eletrônicos mantendo níveis elevados de segurança, tendo baixo consumo de energia e baixo custo.

Essa tecnologia opera sobre uma banda de radiofrequência única e utiliza uma técnica de transmissão de espalhamento espectral por saltos em frequência, de acordo com (SILVA, 2009). Com ela é possível a criação de uma rede de área pessoal e transmissões assíncronas, como transmissão de voz, e síncronas, como a transferência de arquivos.

Um exemplo de módulo de comunicação serial via *bluetooth* pode ser visto na Figura 9.

Figura 9 – Exemplo de módulo de comunicação serial via *Bluetooth*



Fonte: Página da empresa *Build Bot*

## 2.5 Linguagem de Programação Java

De acordo com (CLARO; SOBRAL, 2008), a linguagem Java é uma linguagem Orientada a Objetos (OO) que foi desenvolvida pela *Sun Microsystems* e é capaz de criar diversas aplicações como para *desktop*, *softwares* robustos, aplicativos para *Web*, entre outras. A primeira versão da linguagem Java foi lançada em 1996.

Essa linguagem, segundo (SILVA, 2002), possui várias vantagens por utilizar o paradigma de orientação a objetos, entre eles simplicidade em relação a outras linguagens, segurança em relação a normas específicas de cada aplicativo, robustez, interatividade, neutralidade em relação à sua arquitetura possibilitando a portabilidade de código e alto desempenho com o uso de uma interpretação intermediária chamada *bytecode*.

Segundo (CLARO; SOBRAL, 2008), a linguagem Java é composta de:

- Atributos (Variáveis), os quais são os locais onde os dados são mantidos e são de tipos específicos de acordo com o tipo de informação a ser armazenada neles.
- Métodos, que nada mais são que sub-rotinas, os quais possuem um tipo de retorno e podem receber parâmetros. Dentre os métodos há um tipo muito importante que é o chamado de construtor, o qual é executado automaticamente toda vez que uma nova instância de uma classe é criada.
- Classes, que contêm os atributos e os métodos e ainda são responsáveis pela estrutura dos objetos.

- Objetos, os quais são as instancias das classes e através deles podemos manipular os métodos e atributos de um objeto.

De acordo com (CLARO; SOBRAL, 2008) a grande vantagem de uma linguagem OO está na possibilidade de encapsulamento, ou seja permitir que um método externo a uma classe só faça alterações na mesma utilizando os métodos internos dela. Isso é possível graças ao uso dos modificadores de acesso, que são:

- *private*, o qual permite acesso apenas dentro da própria classe, fazendo com que só possa ser acessado externamente por meio dos métodos da mesma.
- *public*, o qual permite acesso para todas as classes. Os métodos para manipulação interna da classe geralmente recebem esse modificador de acesso.
- *protected*, o qual permite acesso a todas as subclasses dessa classe.

## 2.6 Sistema Operacional *Android*

De acordo com (GONÇALVES, 2013), o Sistema Operacional (OS) *Android* foi desenvolvido por uma empresa chamada *Android Inc.* e posteriormente foi adquirido pela empresa *Google* em 2005, a qual amadureceu o projeto e tornou essa plataforma a primeira do tipo *open source* para desenvolvimento de aplicativos para dispositivos móveis. Ainda de acordo com (GONÇALVES, 2013), os dispositivos móveis são baseados em um núcleo de *Linux* e as aplicações são escritas em linguagem Java, as quais são compiladas em *bytecodes Dalvik* e executadas em uma Máquina Virtual denominada Máquina Virtual *Dalvik*.

Segundo (GONÇALVES, 2013), o *software* utilizado para desenvolver as aplicações em *Android* é o *Android SDK*, que possui um emulador para simular o dispositivo móvel, ferramentas utilitárias e uma interface de programação de aplicativos (API) completa com todas as classes necessárias para desenvolver as aplicações.

Os principais componentes, segundo (GONÇALVES, 2013), da estrutura das aplicações desenvolvidas para *Android* são:

- *AndroidManifest.xml*, é o arquivo de manifesto escrito em XML, obrigatório e único para a aplicação, onde são feitas as configurações gerais dela e dos componentes usados por ela.
- *Activities*, que são as classes que formam a base para o desenvolvimento visual de uma aplicação, realizam os tratamentos de eventos da tela e definem as *Views*. Geralmente é usado uma *activity* para cada tela de uma aplicação.

- *Intents*, que são as mensagens utilizadas para fazer a ligação entre as *activities* da aplicação ou de aplicações diferentes ou então para realizar outras operações agendadas dentro do código.
- *Views*, que são os elementos usados para definir objetos gráficos exibidos na tela, como botões, caixas de diálogo, entre outros.
- *Layouts*, que são os arquivos escritos em Linguagem de Marcação Estendida (XML), a qual de acordo com (ALMEIDA, 2002) é uma linguagem que descreve um dado para armazenamento, transmissão ou processamento por um programa, e que provêm toda a parte gráfica das telas da aplicação.

Para a persistência dos dados, pode-se utilizar o *SQLite* que, segundo (GONÇALVES, 2013), é um banco de dados do tipo *open source* incorporado ao *Android* e que suporta recursos de banco de dados relacionais padrões, como sintaxe padrão, transações e instruções preparadas e, além disso, requer pouca memória em tempo de execução.

## 3 Materiais e Métodos

PARA atingir os objetivos propostos neste trabalho a fim de se provar a hipótese levantada, faz-se necessário o detalhamento dos materiais a serem utilizados e os métodos para se desenvolver o mesmo. Portanto, neste capítulo serão expostos os materiais e métodos utilizados para o desenvolvimento do sistema proposto.

### 3.1 Dispositivo para aquisição dos sinais de EEG

O sistema para aquisição de sinais para esse projeto foi idealizado de modo a obter sinais de um eletrodo seco de EEG e enviá-los via comunicação serial *Bluetooth*. Após realizada a pesquisa sobre os produtos disponíveis no mercado, foi escolhido o Kit para iniciantes chamado *Think Gear ASIC Module* (TGAM) desenvolvido pela empresa *Neurosky* e vendido ao custo de cinquenta e seis (56) dólares americanos. A sua escolha se deu pelos motivos abaixo:

- A capacidade do TGAM se conectar diretamente a um eletrodo seco e aos eletrodos de orelha para referência;
- O TGAM ser capaz de captar sinais de baixa amplitude graças a um amplificador interno;
- O TGAM realizar conversão Analógico-Digital do sinal e as filtragens de Passa-baixa, Passa-alta e os dois bloqueios das frequências de rede (50 e 60 Hz) antes de enviá-lo para o módulo *bluetooth*;
- O fato de o TGAM possuir uma taxa de amostragem de 512 sps e converter os sinais com 16 *bits* de precisão.
- O TGAM ter a sua eficácia como medidor de ondas cerebrais comprovada de acordo com (APEKSHA; PRATHIBHA, 2015)

#### 3.1.1 Montagem do kit

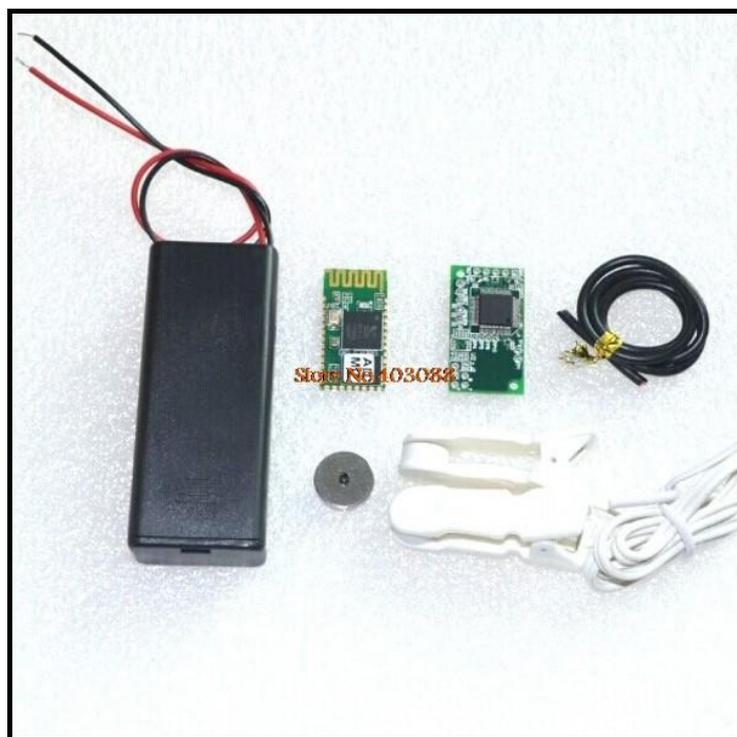
O kit adquirido para esse projeto foi o comercializado pela empresa *Wuxi Sichiray Tech Co.*, que pode ser visto na Figura 10. Esse kit contém:

- Um TGAM;
- Um eletrodo seco de aço inoxidável;
- Um cabo blindado de 30 cm;

- Um porta-baterias com chave de liga-desliga;
- Dois eletrodos de orelha;
- Um módulo de comunicação serial via *Bluetooth* 2.0 FT06.

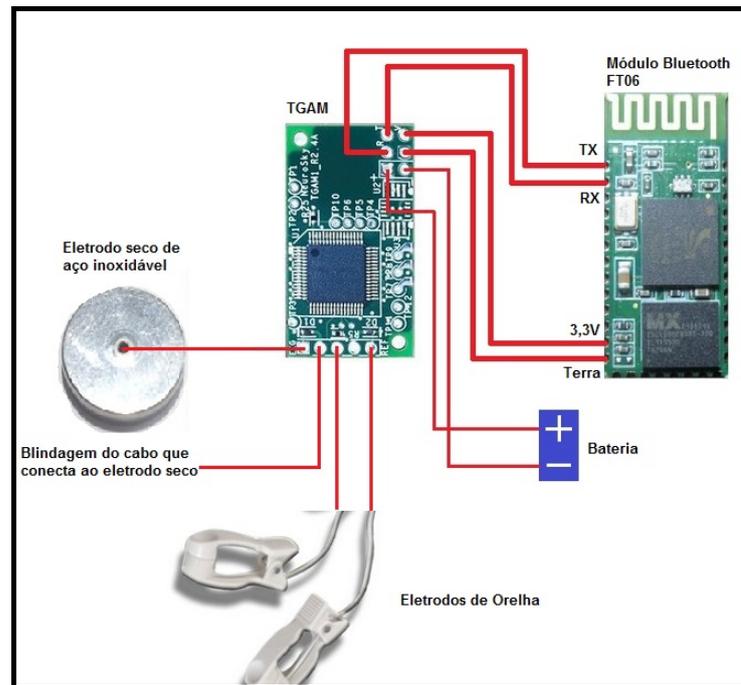
Após a chegada do kit adquirido, foi realizada a soldagem dos componentes elétricos no laboratório da Escola Superior de Tecnologia (EST) da Universidade do Estado do Amazonas (UEA). O esquemático do circuito final pode ser visto na Figura 11. Após a soldagem dos componentes, foi feito um teste simples para determinar se o circuito estava enviando dados pela comunicação serial por meio da conexão do mesmo com uma Porta Serial de um computador e visualização dos dados em um Terminal de acesso com *baud rate* de 9600 bps.

Figura 10 – Kit para iniciantes TGAM



Fonte: Página da empresa *Aliexpress*

Figura 11 – Esquemático do circuito para aquisição de sinais de EEG



Fonte: Autoria do autor

Após a realização do teste de conectividade do circuito, foi realizada a compra de uma bandagem para a cabeça no valor de vinte (20) reais. Foi feita então uma adaptação por meio de costura, a fim de comportar o circuito elétrico e também posicionar o eletrodo seco na posição Fp2 do sistema internacional 10/20. A bandagem adaptada pode ser visualizada na Figura 12. Com a conclusão do dispositivo, foi possível a realização dos primeiros testes reais e aquisição dos primeiros sinais.

Figura 12 – Bandagem para a cabeça adaptada com o circuito de aquisição de sinais de EEG



Fonte: A autoria do autor

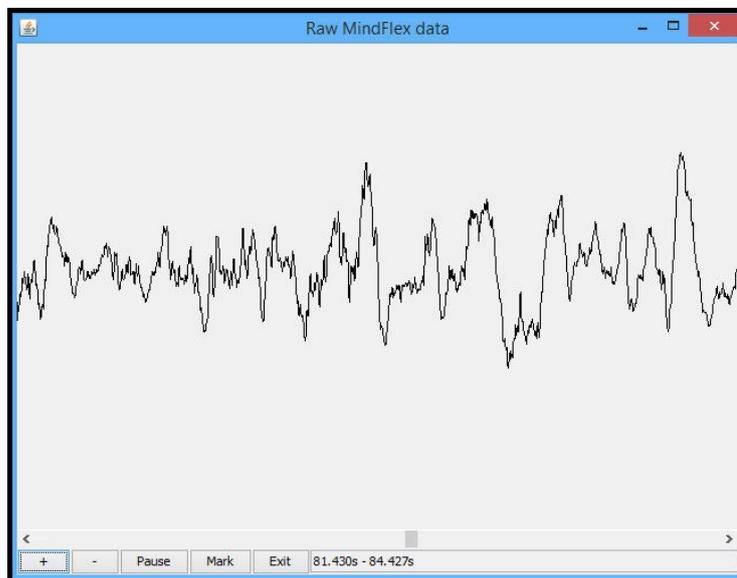
### 3.1.2 Aquisição dos primeiros sinais de EEG

Com a aquisição dos primeiros sinais, verificou-se que o TGAM encontrava-se bloqueado para o envio de sinais de EEG brutos, o mesmo só enviava informações predefinidas de fábrica. Para resolver o problema, foram buscadas soluções e duas foram encontradas: fazer uma modificação no circuito do TGAM a nível de soldagem ou fazer o envio de um *byte* de comando por meio de um *software* conectado ao circuito via comunicação serial *Bluetooth*. As duas soluções foram analisadas e a primeira foi descartada, pois a modificação era bastante delicada de ser realizada e poderia danificar o TGAM permanentemente. Então a segunda solução foi escolhida.

Para a mudança do TGAM via *software*, era necessário enviar o *byte* "0x02" para o TGAM em cada inicialização com um *baud rate* de 9600 bps. Como o módulo de comunicação serial possui um *baud rate* fixo de 57600 bps, foi necessário efetuar um cálculo para que ele pudesse enviar esse *byte* de forma correta. Observando-se que 57600 dividido por 9600 é igual a 6, enviando cada *bit* seis vezes, seria o mesmo que enviar o *byte* com o *baud rate* reduzido. Para o teste dessa solução, foi utilizado um código em Linguagem Java de fonte aberta fornecido pela empresa *Arpruss*. A Figura 13 mostra o sinal obtido com o

uso do *software* fornecido. Após a mudança ser realizada, o TGAM passou a enviar sinais de EEG brutos com *baud rate* de 57600 bps compatível com o módulo de comunicação serial.

Figura 13 – Resposta obtida após a modificação do TGAM via *software*



Fonte: A autoria do autor

## 3.2 Desenvolvimento do Aplicativo para *Android*

Uma vez que o *hardware* se encontrava devidamente montado e testado, foi dado início ao desenvolvimento do aplicativo para o OS *Android*. A primeira coisa a se fazer para o desenvolvimento do aplicativo foi a sua modelagem com a definição de suas regras de funcionamento. As regras definidas foram:

- O aplicativo deve ter controle de acesso por meio de uma tela de *Login*, a qual coleta o *login* e senha do usuário, verifica se o mesmo se encontra cadastrado e então o acesso ao aplicativo é liberado.
- No caso de um novo usuário para o sistema, deverá haver uma tela de cadastro de novos usuários, a qual deverá coletar dados pessoais como: nome completo, *login*, senha, data de nascimento e tipo de deficiência caso houver. Isso é importante para que os dados de usuários sejam armazenados em um banco de dados para futura análise de preferências por usuário.
- Quando na tela principal do aplicativo, o usuário poderá escolher se quer ou não utilizar o aplicativo conectado à Interface Cerebral, pois o mesmo deverá atender também as pessoas que sofrem com mudez ou afonia além das pessoas com doenças degenerativas como a ELA.

- O aplicativo deverá fornecer ao usuário as opções de escrever palavras e frases, editá-las, reproduzi-las sonoramente e enviá-las por *e-mail* e essas opções deverão estar disponíveis para o aplicativo conectado ou não à Interface Cerebral.
- O aplicativo contará com o armazenamento das palavras digitadas por cada usuário e, em tempo real, auxiliará o mesmo no processo de digitação fornecendo até três sugestões para tornar o processo de uso do aplicativo mais rápido com o tempo.
- A ferramenta para envio de *e-mails* deverá ser automática, sem que o usuário tenha que ser redirecionado para outra tela para concretizar o envio da mensagem. Isso é importante porque pacientes com ELA não possuem a capacidade de interagir com outras telas do dispositivo móvel.

Após todas as regras definidas para o aplicativo, foi feita a escolha do dispositivo móvel a ser utilizado para o desenvolvimento e testes. O *tablet* escolhido foi o da marca *Samsung* e modelo *Galaxy Tab S SM-T705M*, o qual utiliza a versão 5.0.2 do OS *Android*, possui 3 GB de memória RAM, um processador de 1,9 GHz e uma tela de 8,4". O dispositivo tem custo em torno de mil e quinhentos (1500) reais e pode ser visto na Figura 14.

Figura 14 – *Tablet* escolhido para o desenvolvimento e testes do aplicativo



Fonte: Página da empresa *Samsung Updates*

### 3.2.1 Desenvolvimento dos *Layouts* das telas

Com a escolha do dispositivo móvel, foi dado início ao desenvolvimento dos *layouts* das três telas do aplicativo. Para isso foi utilizada IDE *Android Studio* e os mesmos foram desenvolvidos em Linguagem XML. A tela de *Login* possui o ícone obtido gratuitamente no domínio público da empresa *The Noun Project* e o nome escolhido para o aplicativo, "Mind Talks", que significa em português "Mente fala". A tela também possui dois campos para preenchimento de *login*, senha e dois botões, um para entrar no aplicativo e outro para se encaminhar à tela de Cadastro de Novos Usuários. A tela pode ser vista na Figura 15.

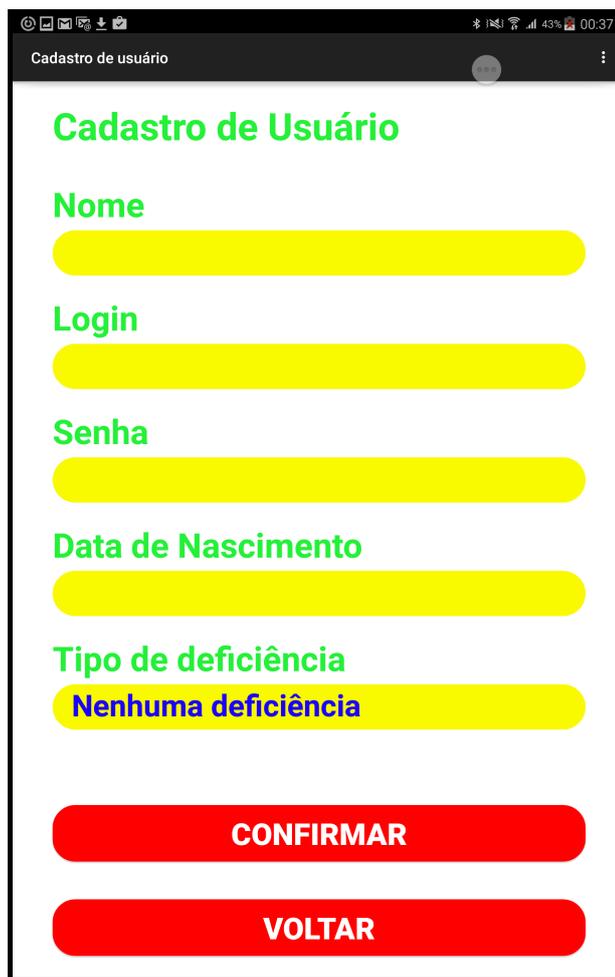
Figura 15 – Tela de Login do aplicativo "Mind Talks"



Fonte: A autoria do autor

A segunda tela a ser desenvolvida foi a tela de Cadastro de Novos Usuários. Essa tela possui campos para preenchimento do nome do usuário, *login*, senha, data de nascimento e um campo de tipo de deficiência com três opções de seleção: "Nenhuma deficiência", "Mudez ou Afonia" e "Esclerose Lateral Amiotrófica". Nessa tela as informações do usuário são coletadas para serem salvas no banco de dados e permitir acesso desse usuário ao sistema. A tela encontra-se ilustrada na Figura 16.

Figura 16 – Tela de Cadastro de Novos Usuários do aplicativo ”Mind Talks”



Cadastro de usuário

### Cadastro de Usuário

**Nome**

**Login**

**Senha**

**Data de Nascimento**

**Tipo de deficiência**

**Nenhuma deficiência**

**CONFIRMAR**

**VOLTAR**

Fonte: A autoria do autor

Após a conclusão das duas primeiras telas, foi desenvolvida a tela principal do aplicativo. Essa tela se baseou em um modelo de Quadro Alfabético utilizado em Comunicação Suplementar e Alternativa (AAC) para pessoas que não podem se comunicar devido a doenças degenerativas motoras e Paralisia Cerebral. A tela principal pode ser vista na Figura 17.

Na tela principal há seis cores para escolha: verde, amarelo, azul, rosa, preto e vermelho. Para cada cor há um grupo de letras possíveis de serem escolhidas correspondentes a essa cor. Há também na tela:

- A caixa de texto principal onde aparecem as letras digitadas pelo usuário;
- O botão ”Ligar/Desligar Interface Cerebral” para ligar ou desligar a Interface Cerebral;
- O botão ”Cores” para ir para o modo de Seleção de Cores;
- O botão ”Ok” para finalizar e armazenar uma palavra digitada;

- O botão "Botões" para ir para o modo de Seleção de Botões;
- O botão "Apagar" para apagar letras escritas na caixa de texto principal;
- O botão "Enter" para iniciar uma nova linha;
- O botão "Reproduzir" para emitir sonoramente o que está escrito na caixa de texto principal;
- O botão "Enviar e-mail" para enviar por *e-mail* a mensagem escrita na caixa de texto principal;
- Três caixas de texto para as sugestões de palavras a serem fornecidas pelo aplicativo.

Figura 17 – Tela Principal do aplicativo "Mind Talks"



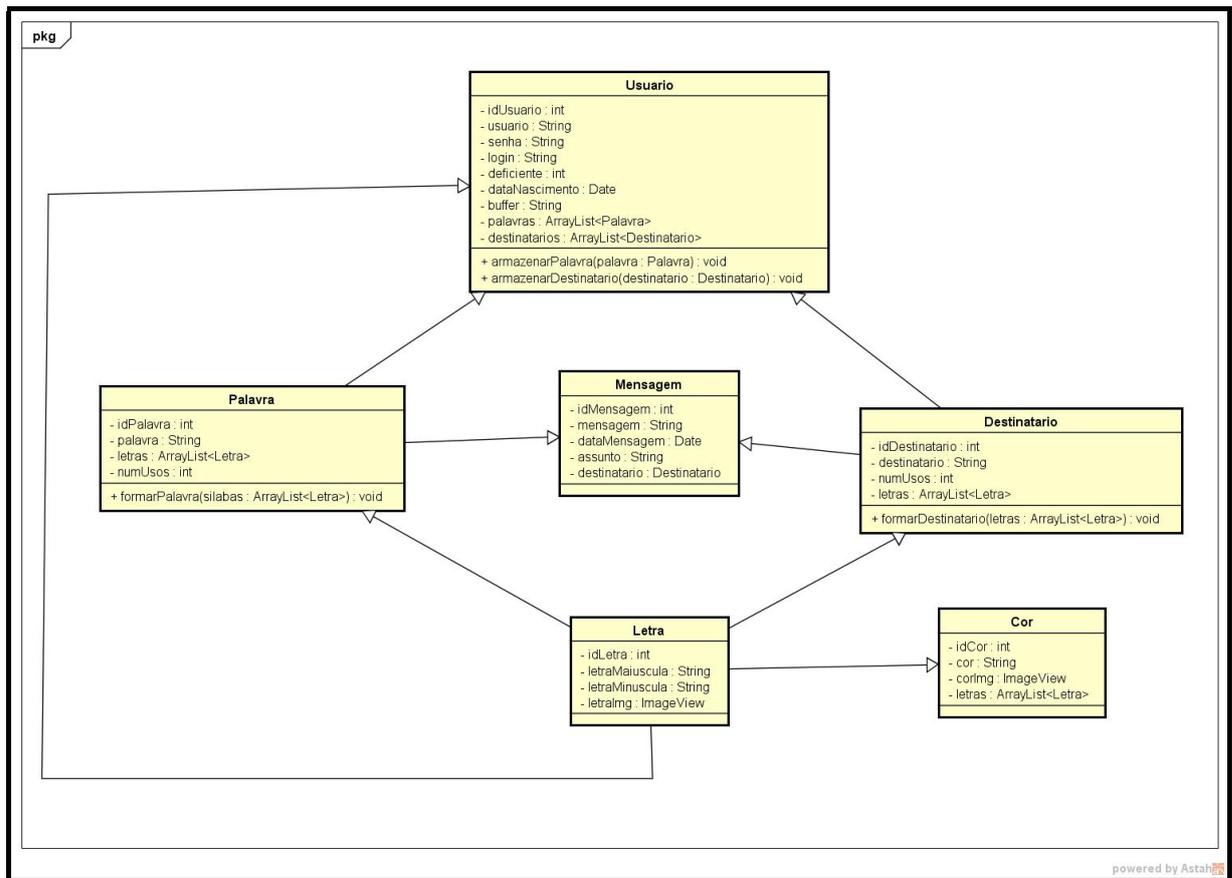
Fonte: Autoria do autor

Para cada uma das telas criadas, foram desenvolvidas duas versões para as possíveis orientações do aplicativo, pois o *tablet* pode ser manuseado tanto em orientação "Retrato" como em "Paisagem".

### 3.2.2 Modelagem do aplicativo e conexão com o Banco de Dados

A modelagem de qualquer aplicativo, seja ele para *Desktop*, *Web* ou *Android*, é tipicamente realizada utilizando-se a Linguagem de Modelagem Unificada (UML). Essa linguagem, de acordo com (COSTA, 2001), "define um conjunto básico de diagramas e notações que permitem representar as múltiplas perspectivas (estruturais / estáticas e comportamentais / dinâmicas) do sistema sobre análise e desenvolvimento". Para o aplicativo em questão foi desenvolvido, utilizando o *software Astah Professional*, um Diagrama de Classes para atender às regras acima esclarecidas, o qual pode ser visto na Figura 18.

Figura 18 – Diagrama de Classes do aplicativo ”Mind Talks”



Fonte: Autoria do autor

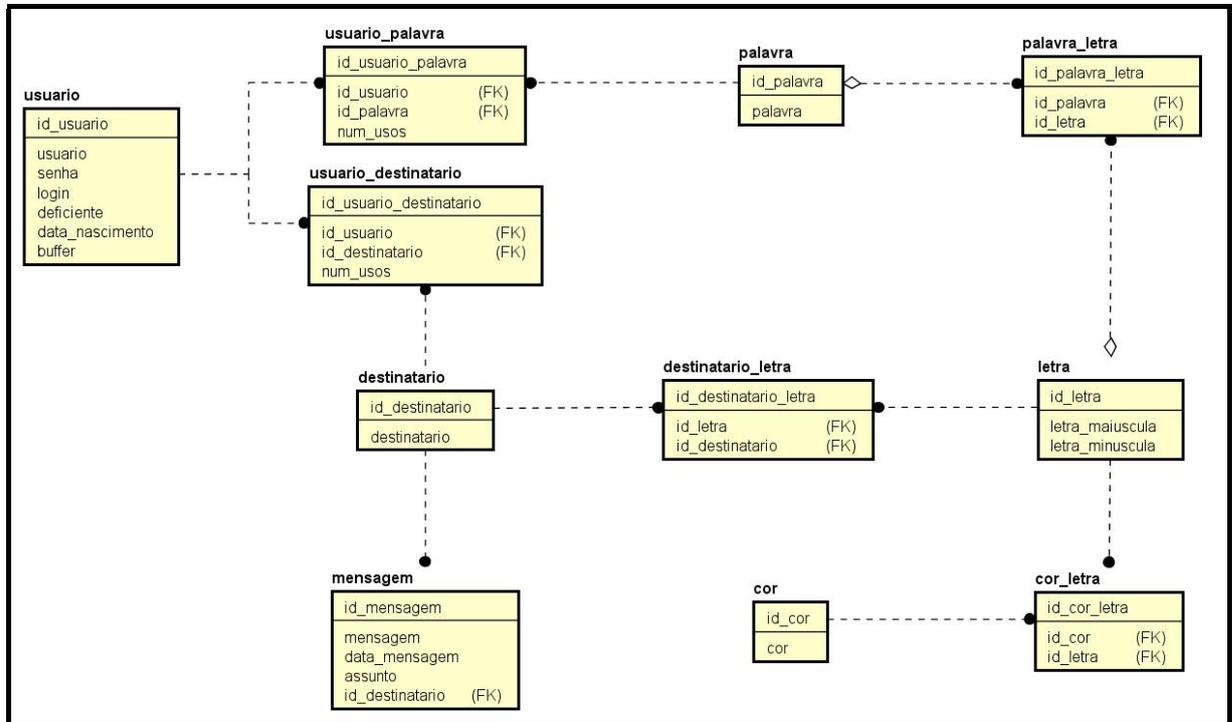
As classes criadas foram:

- Cor, a qual controla e faz a ligação da cor com suas letras correspondentes;
- Letra, a qual armazena as informações das letras e outros caracteres;
- Palavra, a qual controla as palavras e faz a conexão das letras com as palavras;
- Usuário, a qual contém todas as características dos usuários;
- Destinatario, a qual faz o controle dos dados dos destinatários dos *e-mails* a serem enviados;
- Mensagem, a qual controla as informações das mensagens a serem enviadas por *e-mail*.

Após a modelagem das Classes necessárias para o funcionamento do aplicativo, foi feita a modelagem do Banco de Dados (BD) a ser utilizado. O BD escolhido foi o *SQLite*, já embutido em todos os dispositivos móveis com SO *Android*. Para armazenar todas as informações de maneira correta foram definidas onze (11) tabelas a serem criadas, as

quais visam também permitir o relacionamento entre as classes. Na Figura 19 o Diagrama Entidade Relacionamento (DER) pode ser visto.

Figura 19 – DER do aplicativo "Mind Talks"



Fonte: Autoria do autor

Após realizada a modelagem completa do aplicativo, chegou a hora de começar a escrever o algoritmo em Linguagem Java. O primeiro passo foi criar o BD utilizando a IDE *Android Studio*. Para isso, foi criado um objeto chamado Objeto de Acesso aos Dados (DAO). De acordo com (ORACLE, ), o DAO é um padrão de projetos que tem como propósito encapsular todo o acesso ao BD e gerenciar essa conexão. Para a criação do BD e dos métodos de pesquisa, inserção, edição e exclusão de dados das tabelas foram utilizados conhecimentos básicos de Linguagem de Consulta Estruturada (SQL). O SO *Android* possui uma classe chamada "*SQLiteOpenHelper*", a qual deve ser estendida pelo DAO para criação e manipulação de BDs em *SQLite*.

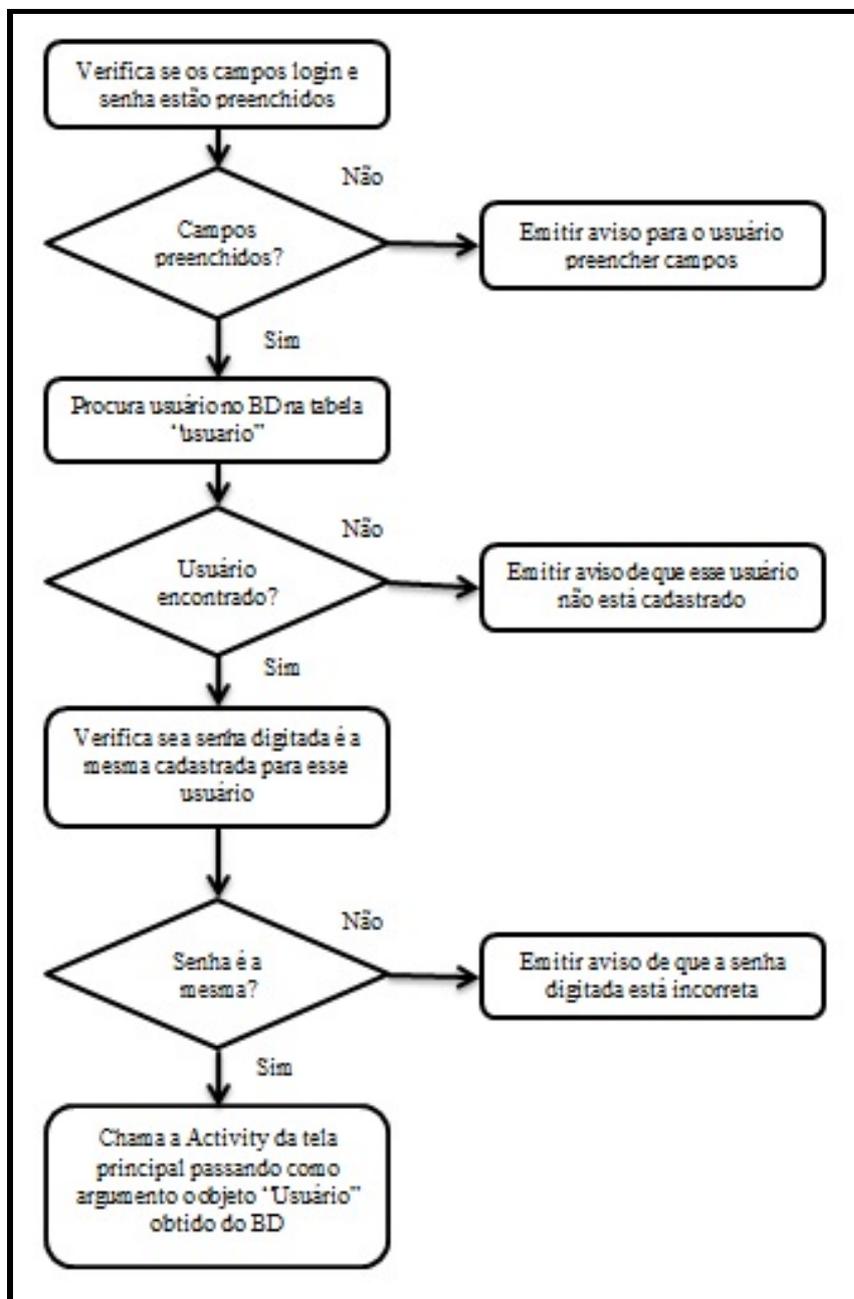
### 3.2.3 Criação das *Activities*

Após concluída a fase de modelagem e preparação da programação de fundo do aplicativo, foi a hora de utilizar os elementos criados para dar vida ao aplicativo. Isso foi realizado com a criação das chamadas *Activities*. Ao todo foram desenvolvidas três (3), uma para cada tela.

A primeira a ser criada foi a "*LoginActivity*", a qual faz o controle da tela de *login*. Essa *Activity* possui duas caixas de texto para o usuário digitar o *login* e a senha e dois botões, "*Confirmar*" e "*Cadastrar-se*". Se o usuário apertar o primeiro botão, acontecerá

o processo mostrado no fluxograma da Figura 20. Se o usuário apertar o segundo botão, o aplicativo chamará a *Activity* de Cadastro de Novos Usuários.

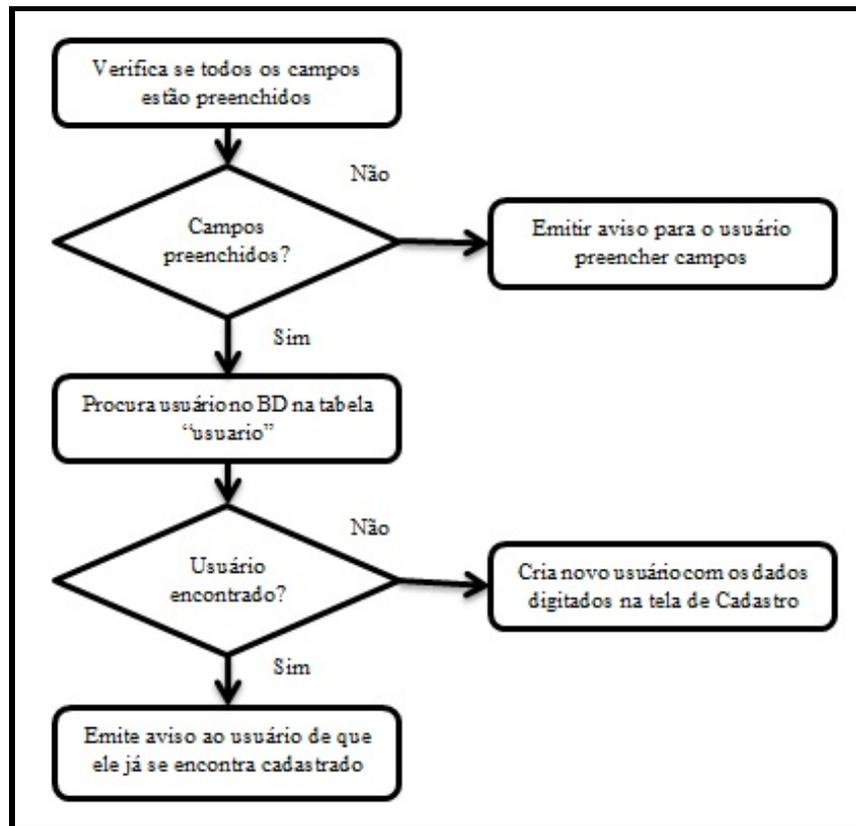
Figura 20 – Lógica para o botão "Confirmar" da tela de Login



Fonte: A autoria do autor

A segunda *Activity* a ser desenvolvida foi a "CadastroActivity" para efetuar o cadastro de novos usuários no aplicativo. Ela possui os seguintes campos: nome do usuário, *login*, senha, data de nascimento e tipo de deficiência se houver. Ela também contém os botões "Confirmar" e "Voltar". Se o usuário apertar o primeiro botão, acontecerá o processo mostrado no fluxograma da Figura 21. Caso o segundo botão for pressionado o aplicativo chama a *Activity* de *login* para que o novo usuário possa adentrar o aplicativo com seu *login* e senha cadastrados.

Figura 21 – Lógica para o botão "Confirmar" da tela de Cadastro de Novos Usuários

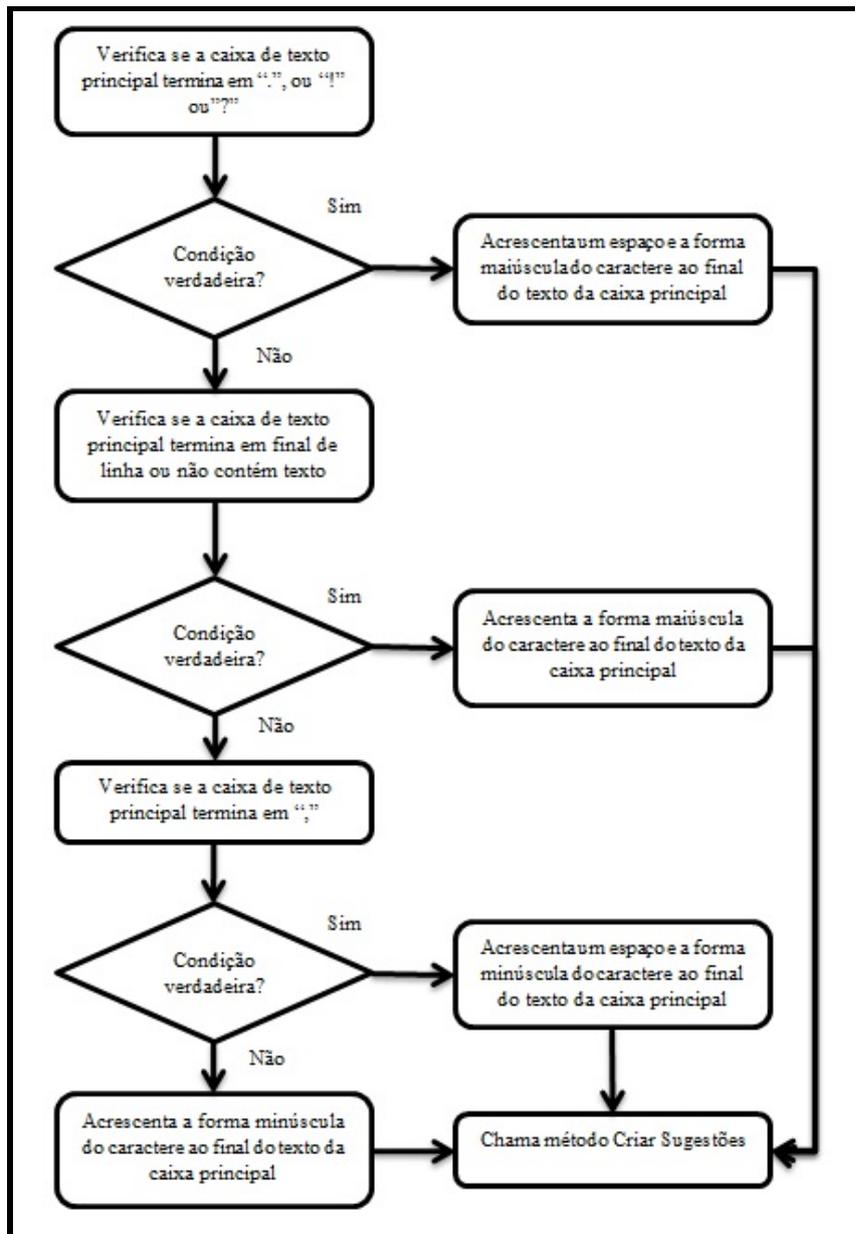


Fonte: A autoria do autor

A terceira *Activity* a ser desenvolvida foi a "MainActivity" para controlar a tela principal do aplicativo. Ela controla todos os processos principais e para cada um de seus caracteres ou botões pressionados pelo usuário há uma lógica a ser realizada.

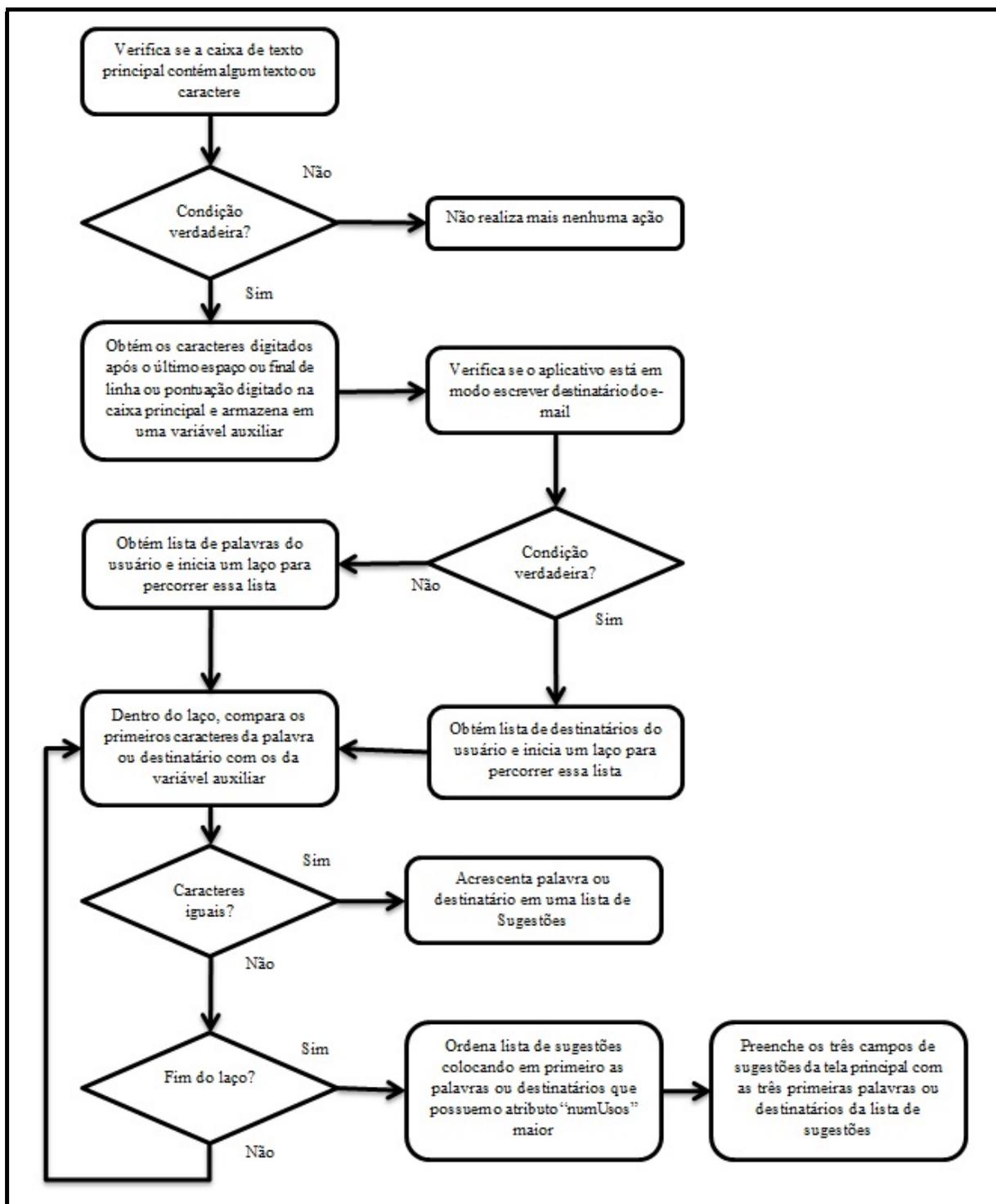
Quando o usuário pressiona qualquer um dos caracteres disponíveis na tela principal, ocorre a lógica descrita no fluxograma da Figura 22. O método "Criar Sugestões" se refere às três sugestões dadas pelo aplicativo para facilitar a digitação pelo usuário. A lógica para a sua implementação se encontra no fluxograma da Figura 23.

Figura 22 – Lógica para quando o usuário pressiona qualquer caractere da tela principal



Fonte: Autoria do autor

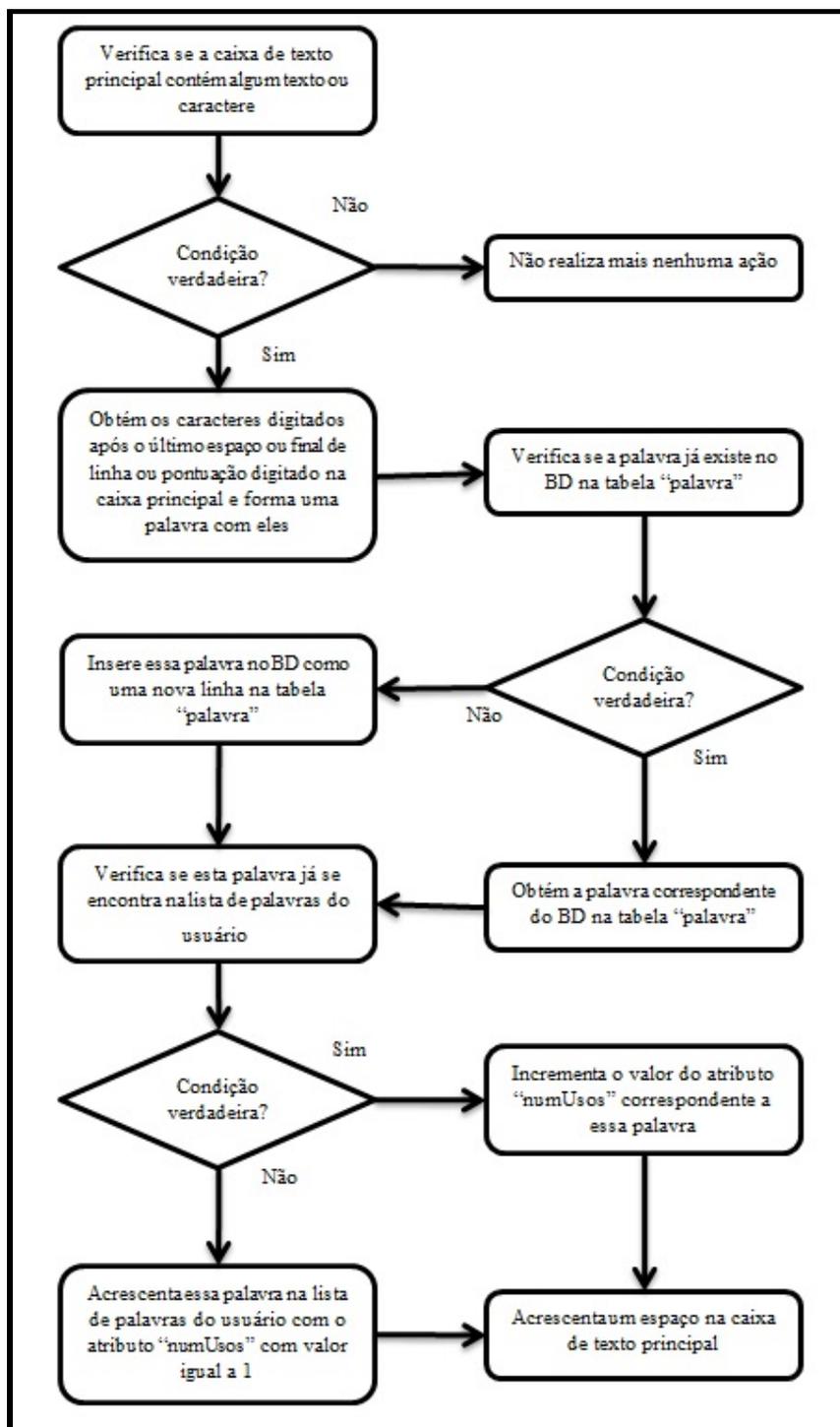
Figura 23 – Lógica para o método Criar Sugestões



Fonte: Autoria do autor

Ao pressionar o botão "Ligar Interface Cerebral" o aplicativo habilita o funcionamento da Interface Cerebral, a qual será tratada mais adiante. Os botões "Cores" e "Botões" fazem parte da Interface Cerebral e seu funcionamento também será descrito nessa seção do texto. O botão "Ok" deve ser pressionado quando o usuário terminar de digitar uma palavra e sua lógica de implementação encontra-se explicada no fluxograma da Figura 24.

Figura 24 – Lógica para quando o usuário apertar o botão "Ok" da tela principal



Fonte: Autoria do autor

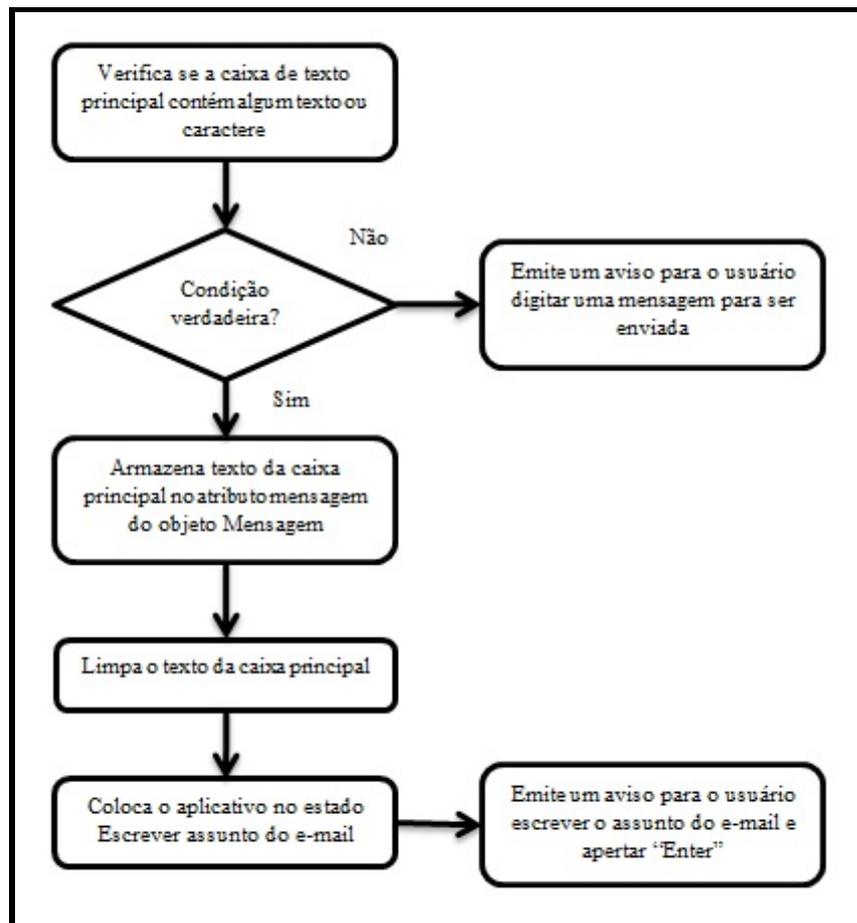
Quando o usuário apertar o botão "Apagar" da tela principal, o último caractere da caixa de texto principal é apagado. Se o usuário segurar o botão apertado, toda a caixa de texto é limpa. Quando o botão "Reproduzir" é pressionado, o aplicativo utiliza a ferramenta fornecida pelo SO *Android* chamada *Text-to-Speech* para reproduzir sonoramente o texto que estiver presente na caixa de texto principal. Essa ferramenta foi criada e disponibilizada na API 4 do SO *Android*, de acordo com (ANDROID, c).

O aplicativo possui um modo de funcionamento para enviar *e-mails*. Esse modo possui três possíveis estados:

- Normal, onde o aplicativo não se encontra no modo de envio de *e-mails*;
- Escrever assunto do *e-mail*, onde o texto digitado na caixa de texto principal será posteriormente salvo em um objeto do tipo Mensagem como o assunto do *e-mail* a ser enviado;
- Escrever destinatário do *e-mail*, onde o texto digitado na caixa de texto principal será posteriormente salvo em um objeto do tipo Mensagem como o destinatário do *e-mail* a ser enviado.

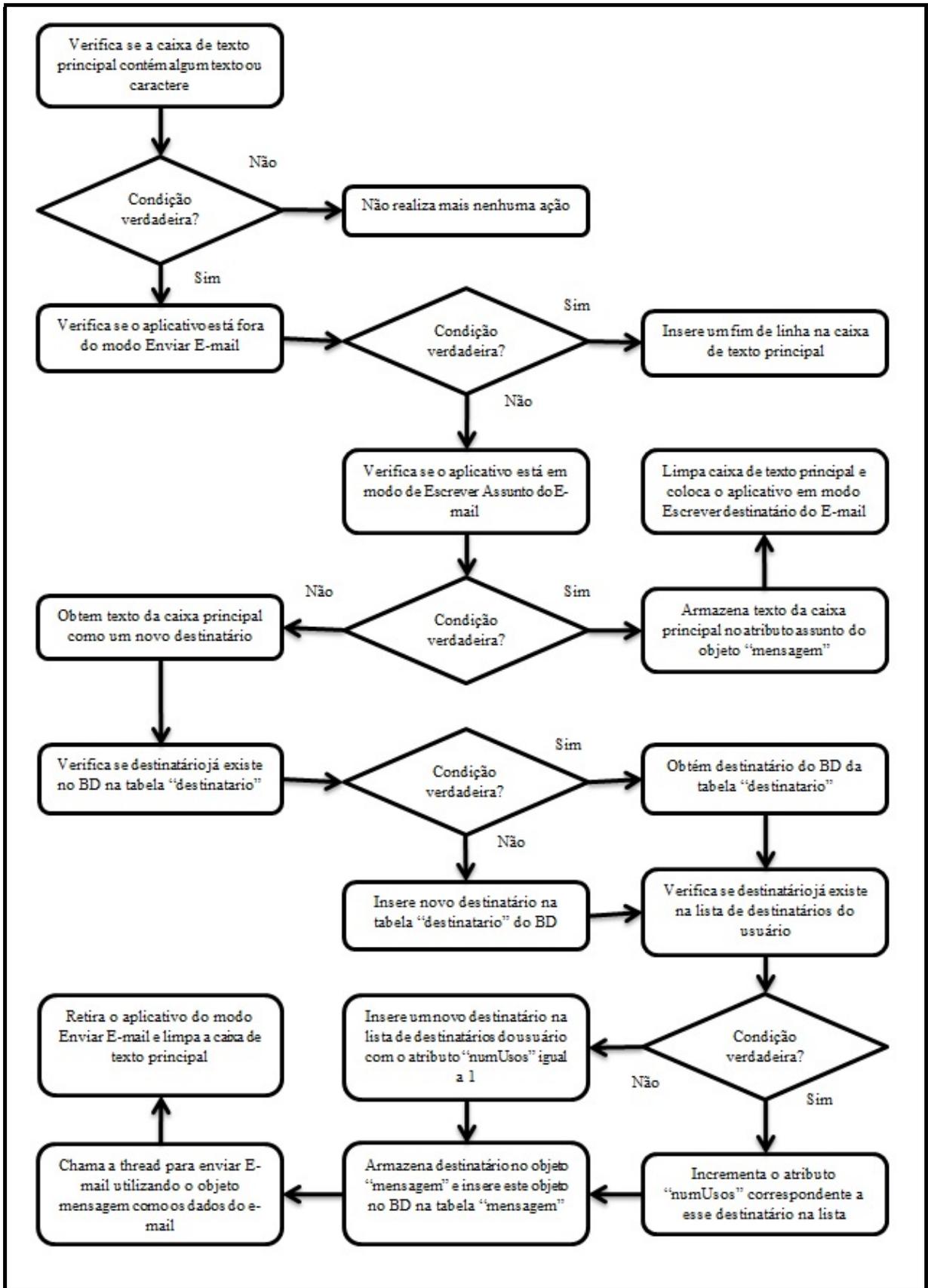
O modo de envio de *e-mails* é controlado pelos botões "Enviar E-mail" e "Enter". As lógicas implementadas para quando o usuário apertar cada um dos dois botões estão nos fluxogramas das Figuras 25 e 26.

Figura 25 – Lógica para quando o usuário pressiona o botão "Enviar e-mail" da tela principal



Fonte: Autoria do autor

Figura 26 – Lógica para quando o usuário pressiona o botão "Enter" da tela principal



Fonte: Autoria do autor

Para o envio automático dos *e-mails* o aplicativo utilizou uma ferramenta do SO *Android* chamada "AsyncTask", que nada mais é do que uma ferramenta fornecida por esse SO na API 3 para realizar funções na *thread* principal sem ter que se preocupar com manipulação de outras *threads*, conforme explicado em (ANDROID, a). Com ela é possível chamar uma tarefa a qualquer momento da execução do aplicativo. A tarefa realizada por essa ferramenta foi o envio de um *e-mail* utilizando um servidor "SMTP" padrão fornecido pela ferramenta "Gmail" da empresa *Google*. Esse servidor acessa uma conta de *e-mail* criada exclusivamente para uso do aplicativo "Mind Talks" e dela envia o *e-mail* com os dados contidos no objeto mensagem obtido da tela principal.

Ao pressionar uma das três sugestões dadas na tela principal, a palavra que o usuário está digitando na caixa de texto principal é completada com a da sugestão selecionada por ele.

### 3.3 Desenvolvimento do algoritmo para interpretar os sinais adquiridos

Uma vez que o aplicativo se encontrava com todas as suas funcionalidades implementadas e testadas ainda sem conexão com o TGAM, chegou a hora de desenvolver o algoritmo para obter o sinal bruto do dispositivo cerebral e traduzi-lo para comandos que pudessem ser lidos pelo aplicativo "Mind Talks". Por questões de praticidade e para não alterar o código do aplicativo com um algoritmo ainda não devidamente testado, um novo projeto no *Android Studio* foi iniciado.

O primeiro estudo a ser realizado foi como conectar o TGAM ao *tablet* por meio de *bluetooth*. Para isso foi utilizada uma classe própria do SO *Android* chamada "BluetoothAdapter". Com essa classe é possível criar um adaptador virtual a ser utilizado pelo TGAM para se conectar ao aplicativo. Além disso, o aplicativo também tem que pedir a permissão para utilizar o *bluetooth* do dispositivo móvel no arquivo de manifesto desse projeto.

Depois foi realizado o estudo da biblioteca fornecida pela empresa *Neurosky* para se trabalhar com o TGAM em dispositivos móveis com SO *Android*. Essa biblioteca possui, entre outras classes, uma chamada "TGDevice", a qual possui todas as opções de aquisição de dados de um TGAM já conectado ao dispositivo móvel via *Bluetooth*. Dentre as opções fornecidas, duas foram utilizadas no algoritmo desenvolvido, a de aquisição de sinais brutos e a de tipo de sinal. A função de tipo de sinal foi utilizada apenas para constatar que o TGAM estava no modo de envio de sinais brutos e não no modo de envio de informações diversas. E a função de aquisição de sinais brutos é uma função que recebe os sinais brutos enviados e os converte para números inteiros decimais.

Essa classe "TGDevice" recebe como parâmetros em seu construtor, o adaptador virtual de *bluetooth* utilizado para conectar o TGAM ao aplicativo e uma variável do tipo

”Handler”, que é uma classe especializada em lidar com mensagens de resposta de uma *Thread* a qual deve estar ligada, como explicado em (ANDROID, b). Dentro da instância dessa variável é que o algoritmo tem seu desenvolvimento. A classe ”TGDevice” então inicia uma *thread* de conexão com o TGAM e envia retornos na mesma velocidade da taxa de envio de sinais dele, no caso dos sinais de EEG a taxa é de 512 amostras por segundo.

Para testar a conectividade do TGAM com o *tablet*, inicialmente foi criado apenas um algoritmo para escrever na tela os valores de tipo de sinal e do sinal bruto convertido. Ao energizar o TGAM e antes de realizar esse teste, o mesmo tinha que ser desbloqueado para enviar sinais brutos, para isso foi utilizado um aplicativo do tipo Código Aberto fornecido pela empresa *Arpruss*. Após desbloqueado, foi feito o teste para envio de sinais. O teste mostrou que os valores obtidos, após convertidos para valores em Volts utilizando a Equação 3.1 fornecida pela empresa *Neurosky*, estavam de acordo com valores reais de sinais de EEG.

$$ValorReal(V) = \frac{Valorbruto \times 1,8V}{\frac{4096}{2000}} \quad (3.1)$$

Foram feitos então testes de piscadas para verificar qual seria o melhor valor de base a ser utilizado para detecta-las. Após cinquenta (50) detecções se chegou ao valor de base - 1000 ou - 219.727 $\mu$ V, que se encontra fora do intervalo de amplitudes normais de sinais de EEG, o qual de acordo com (TEPLAN, 2002) varia em módulo de 0,5 a 100  $\mu$ V. Em aquisições de sinais de EEG para outras aplicações, os sinais obtidos das piscadas são geralmente encarados como ruídos e facilmente descartados e só os sinais dentro do range normal são utilizados. O valor de base deve ser muito maior que o range normal em módulo pois o sistema não deve gerar falsos positivos ao detectar outros movimentos dos olhos.

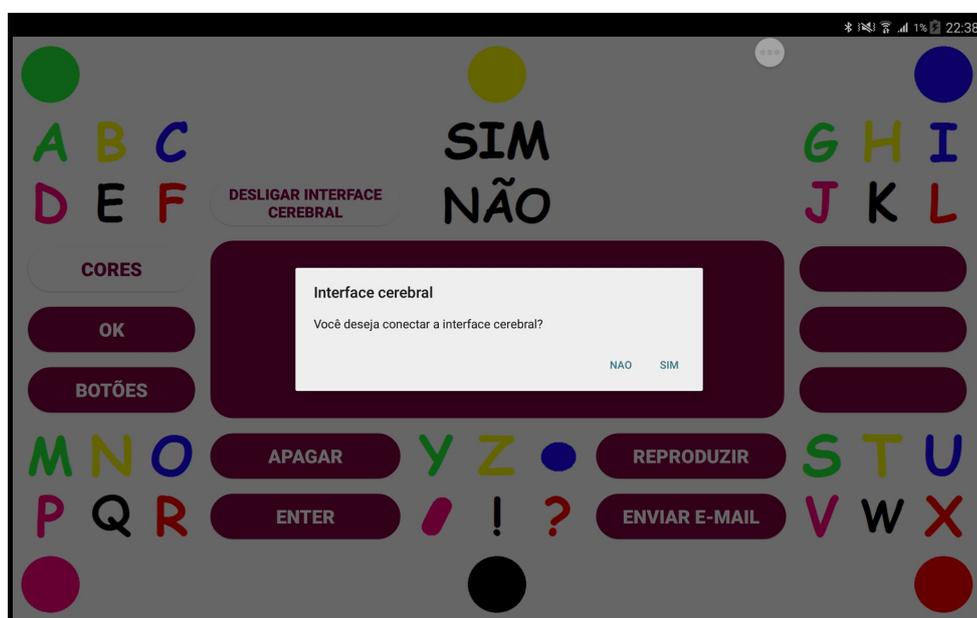
Para a detecção do número correto de piscadas, foi utilizado como base um intervalo de tempo de até 1 milissegundo para a detecção de cada uma delas. Dessa forma, toda vez que o valor bruto ficava abaixo do valor de base por até 1 milissegundo uma piscada era contabilizada pelo algoritmo.

### 3.4 Integração do aplicativo com o algoritmo para interpretação dos sinais

Após verificada a eficácia da detecção de piscadas com o algoritmo de teste, era chegada a hora de integrar o sistema como um todo. Para isso ainda era necessário desenvolver a Interface Cerebral do aplicativo ”Mind Talks”. A primeira coisa a ser feita foi o *Layout* de uma tela auxiliar de conexão. Após isso, essa tela passou a ser um *layout* dentro do *layout* da tela principal que ficava invisível até ser chamado pela *Activity* principal.

Todo o código para a Interface Cerebral foi integrado na "MainActivity", começando pela criação de uma Mensagem de Alerta que aparece quando o usuário acessa a tela principal. Nessa mensagem encontra-se a pergunta: "Você deseja conectar a interface cerebral?" e o usuário pode escolher entre dois botões, um afirmativo e um negativo. Em caso afirmativo, a *activity* torna a tela de conexão visível e o usuário pode então estabelecer a conexão com o TGAM utilizando o algoritmo de aquisição de sinais já integrado, essa tela pode ser vista na Figura 27. Para determinar quando o TGAM se encontra conectado e enviando sinais brutos, o aplicativo verifica se o tipo de sinal enviado permanece o de sinais brutos por dez retornos da *thread* de conexão para então retornar a visibilidade da tela principal e iniciar a Interface Cerebral automaticamente.

Figura 27 – Mensagem de Alerta para o usuário escolher se deseja ou não conectar o aplicativo ao TGAM



Fonte: Autoria do autor

Para a criação da Interface Cerebral na tela principal, foi criado o *layout* de uma borda para envolver os componentes passíveis de serem selecionados. Foram também definidos quatro possíveis estados da Interface Cerebral:

- Desligada, no qual só o usuário só pode interagir com o aplicativo com uso de toques na tela.
- Cores, no qual a interface alterna entre as seis (6) cores presentes na tela principal para que o usuário selecione uma.
- Letras, no qual já há uma cor selecionada e a interface alterna entre os caracteres pertencentes a ela para que o usuário selecione um deles.
- Botões, no qual a interface alterna entre os botões "Apagar", "Enter", "Reproduzir", "Enviar e-mail" e os das três sugestões para que o usuário selecione uma opção.

Em todos os estados da Interface Cerebral, o usuário tem a opção de ir para o estado "Cores" ou o "Botões", para isso utilizando os botões "Cores" e "Botões". Após a definição dos estados possíveis da Interface Cerebral, foi definido o tempo que um componente deveria ficar passível de ser selecionado como sendo de 1,5 s. Para fazer com que a interface fizesse essa contagem foram utilizadas duas variáveis de tempo, uma para obter a hora atual do *tablet* ao início da obtenção de sinais e ao final de cada contagem de 1,5 s e a outra para salvar a hora atual do sistema dentro da instância da variável que lida com o retorno da *thread* de conexão e comparar seu valor com a variável de tempo anterior para mudar a seleção quando essa se encontrar 1,5 s à frente.

Para lidar com a mudança de seleções na tela principal, foram criadas duas listas, uma para salvar as cores e seus respectivos caracteres da tela principal e uma para salvar os botões possíveis de serem selecionados. Para percorrer as listas, toda vez que o tempo de 1,5 s é atingido, o índice da lista correspondente ao estado atual da Interface Cerebral é incrementado para que a borda de seleção passe para o próximo item a ser selecionado.

Após a escrita do algoritmo para a criação da Interface Cerebral, o mesmo foi testado e alguns ajustes foram realizados até ser verificado que as condições propostas estavam sendo atendidas corretamente. Após isso, fez-se necessário adaptar as funções de cada caractere e de cada botão à sua seleção com o uso da Interface Cerebral ao invés do toque da tela. Outros testes foram realizados utilizando um botão auxiliar criado na tela apenas para esse fim e constatou-se que todas as funções do aplicativo encontravam-se agora também disponíveis para a seleção com a Interface Cerebral.

Com todos os testes realizados com a Interface Cerebral criada, finalmente a detecção de piscadas foi integrada à *activity* principal também dentro da instância da variável que lida com o retorno da *thread* de conexão. Essa detecção foi implementada para detectar duas piscadas dentro do intervalo de tempo do componente da Interface Cerebral passível de seleção no instante em que elas ocorrerem. Uma vez detectadas as piscadas, o aplicativo realiza a ação determinada pelo item selecionado e muda ou não o estado da Interface Cerebral de acordo com ele.

Foram realizados testes do sistema integrado e após alguns ajustes, constatou-se que o mesmo funcionava da maneira como foi proposto, atendendo a todas as regras e condições acima citadas.

## 4 Resultados e Discussão

ESTE capítulo descreve os resultados obtidos com os testes propostos no capítulo Materiais e Métodos deste trabalho.

### 4.1 Testes do aplicativo Mind Talks

Os primeiros testes realizados foram os das telas de *Login* e de Cadastro. Esses testes consistiram em tarefas de cadastro de novos usuários e tentativas de acesso com *logins* e senhas incorretos. Os resultados obtidos foram satisfatórios, considerando que mesmo após o desligamento do *tablet*, os registros salvos no BD persistiram.

Após isso foram realizados testes na tela principal do aplicativo. Primeiro foram feitos testes com todos os caracteres da tela em todas as situações previstas em sua lógica de implementação e verificou-se que todos estavam de acordo com o proposto. Ao mesmo tempo realizou-se o teste do botão "Ok" para armazenar palavras na lista do usuário e verificar se o método "Criar Sugestões" estava agindo corretamente. Todos os testes se mostraram satisfatórios como pode ser visto na Figura 28.

Figura 28 – Resultado do teste de caracteres combinado ao teste de Sugestões



Fonte: Autoria do autor

Após os testes com os caracteres, foi realizado o teste com o botão "Apagar", o qual também se mostrou satisfatório e com o botão "Reproduzir". Alguns problemas foram constatados com esse último teste pois a API instalada no *Android Studio* não continha o pacote de linguagem para Língua Portuguesa. Para resolver o problema, foi colocado no

código uma *Intent* para verificar se esse pacote se encontra instalado e, como resultado em caso negativo providenciar sua instalação. Após isso os testes se mostraram satisfatórios.

O último teste realizado com o aplicativo ainda não conectado ao TGAM foi o da rotina de envio de *e-mail*. Foi escrita uma mensagem de teste na caixa de texto principal e pressionado o botão "Enviar e-mail". O aplicativo então limpou a tela principal e solicitou que fosse agora escrito o assunto do *e-mail*. Foi então digitado o assunto "Teste" e novamente pressionado o botão "Enter". O aplicativo então apresentou-se como na Figura 29. O destinatário foi digitado corretamente e o botão "Enter" foi mais uma vez apertado, ao que o *e-mail* foi enviado. Ao se verificar a caixa de entrada do destinatário, como visto na Figura 30, constatou-se o sucesso do teste realizado.

Figura 29 – Tela principal no estado de Escrever destinatário do *e-mail*



Fonte: Autoria do autor

Figura 30 – Resultado do teste de envio de *e-mail*

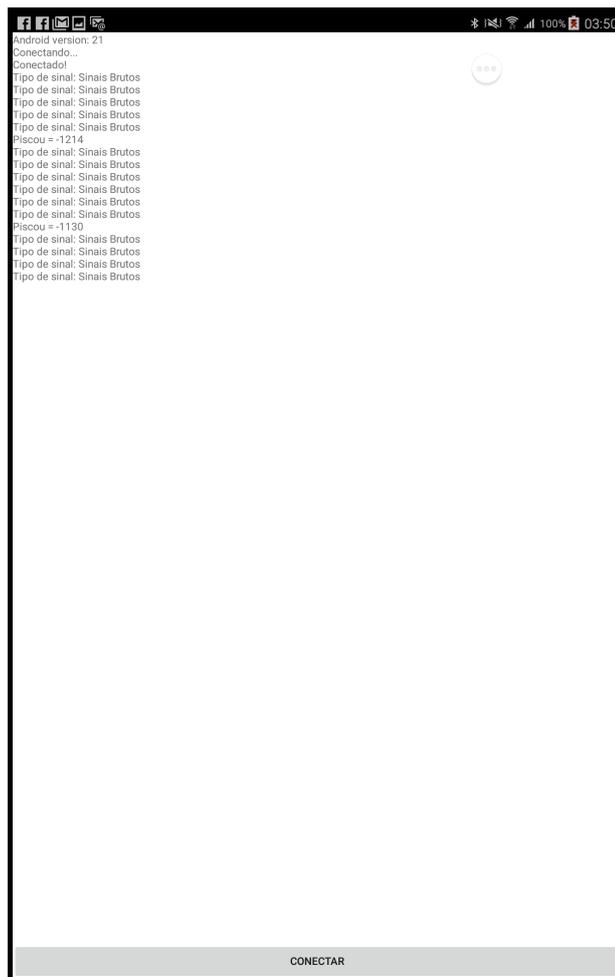


Fonte: Autoria do autor

## 4.2 Testes do algoritmo para tradução dos sinais do TGAM

Após o desenvolvimento do algoritmo para conexão do TGAM ao *tablet* e aquisição dos sinais bruto, foram realizados testes para se chegar ao valor adequado de base para detectar e o tempo para contabilizar as piscadas. Na Figura 31 pode ser vista a tela do aplicativo de teste no momento de duas piscadas contabilizadas.

Figura 31 – Tela do aplicativo de teste para aquisição dos sinais do TGAM



Fonte: A autoria do autor

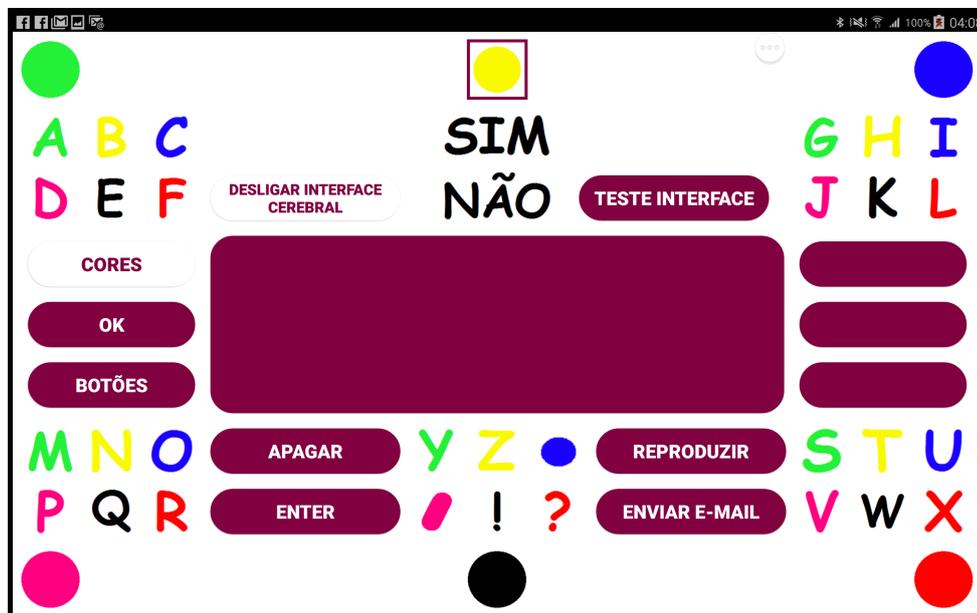
Pode-se ver que os valores obtidos com as piscadas estão abaixo do valor de base escolhido e o número de piscadas contabilizadas foi o de piscadas efetuadas, o que concluiu o teste como satisfatório.

## 4.3 Testes da Interface Cerebral sem o TGAM conectado

Após a definição de estados e criação da Interface Cerebral do aplicativo "Mind Talks", testes foram realizados utilizando um botão auxiliar na tela apenas para simular seleções com o TGAM antes de realmente conecta-lo ao aplicativo. Foram testados os quatro (4) possíveis estados da interface e se com a seleção de um caractere ou um botão, as

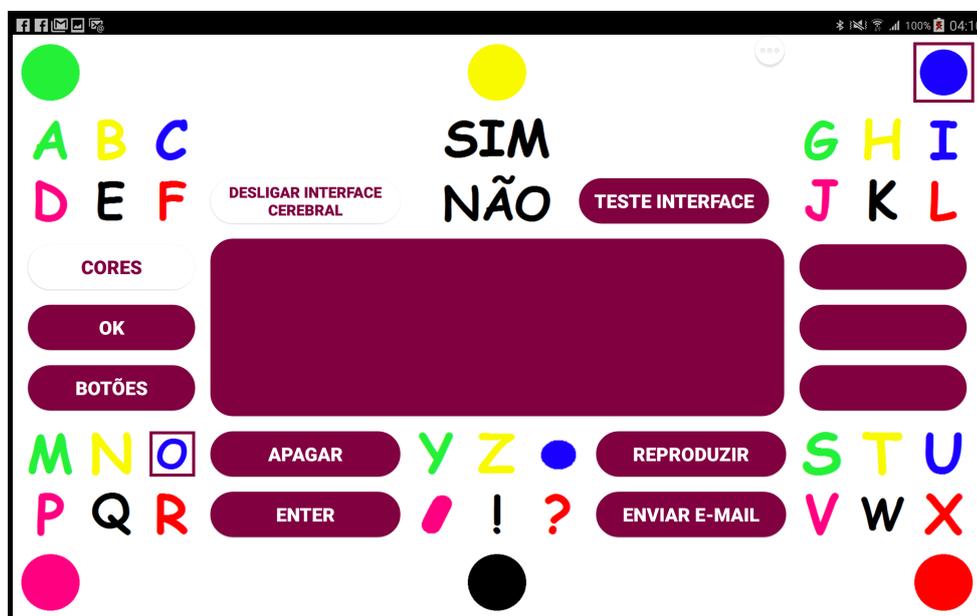
funcionalidades do aplicativo se mantinham as mesmas de antes da implementação da Interface Cerebral. Nas Figuras 32, 33 e 34 os resultados dos testes podem ser visualizados.

Figura 32 – Tela principal no estado Cores da Interface Cerebral



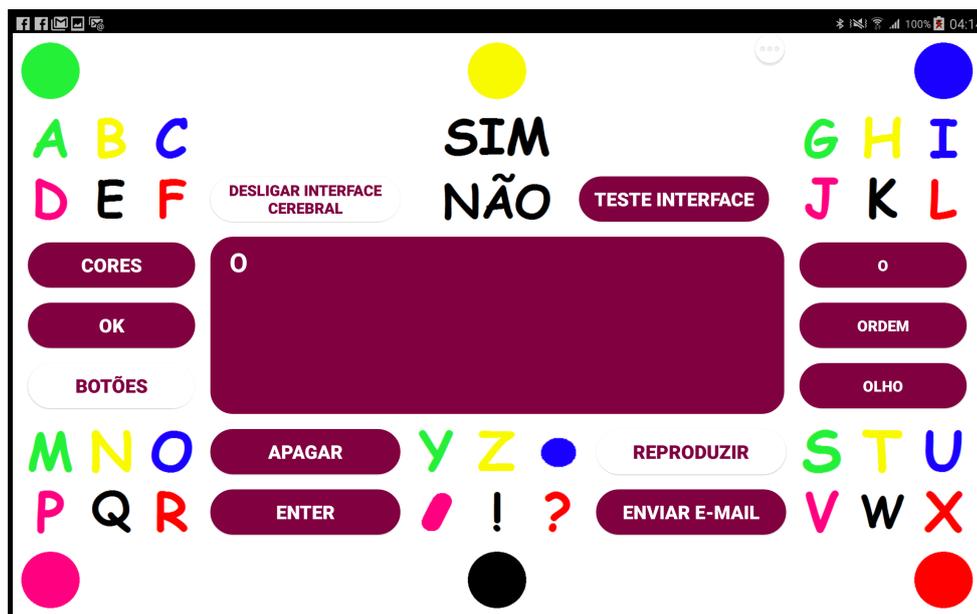
Fonte: Autoria do autor

Figura 33 – Tela principal no estado Letras da Interface Cerebral



Fonte: Autoria do autor

Figura 34 – Tela principal no estado Botões da Interface Cerebral



Fonte: Autoria do autor

#### 4.4 Testes da Interface Cerebral com o TGAM conectado

Ao integrar o algoritmo para conexão do TGAM ao aplicativo e detecção de piscadas, foi notado um problema no funcionamento da Interface Cerebral. O que ocorreu foi que a mesma havia sido desenvolvida utilizando uma ferramenta chamada "Timer Task" fornecida pelo SO *Android*, que de acordo com (ANDROID, d) é um tipo de *Task* que é executada em um tempo especificado. Porém a "Timer Task" é um tipo de *thread* que utiliza muito processamento e acabou entrando em conflito com a *thread* utilizada pelo algoritmo de conexão com o TGAM. Dessa forma o aplicativo não funcionava da forma que deveria.

Em vista de resolver o problema, todo o código da Interface Cerebral teve que ser repensado para que não tivesse que utilizar a "Timer Task" para controle do tempo. Foi então realizado um estudo sobre como obter o tempo atual do dispositivo. Após isso, com o uso de operações lógicas simples e a reformulação do código para ser integrado ao "handler" da *thread* de conexão, a Interface Cerebral pôde ser completamente integrada e voltou a funcionar corretamente.

Foram então realizados os testes com as piscadas e se mostraram bastante satisfatórios.

#### 4.5 Testes finais com o sistema integrado

Após a constatação de que o sistema integrado apresentava todas as funcionalidades propostas, foram realizados alguns testes finais. Um deles foi o teste de persistência da conexão do aplicativo com o TGAM. Foi realizada a conexão do TGAM com o aplicativo

e depois o usuário minimizou a aplicação para utilizar outras aplicações do *tablet*, as quais foram a Câmera e um aplicativo de *e-mails*. Depois o usuário retornou à tela do aplicativo "Mind Talks" e o mesmo tentou escrever o caractere "T" utilizando a Interface Cerebral. O teste foi satisfatório pois além de poder digitar o caractere, o que mostrou que o TGAM ainda se encontrava conectado corretamente, nenhum outro caractere indesejado havia sido digitado na tela, demonstrando que o aplicativo não havia captado nenhum falso positivo durante o teste.

Outro teste realizado foi o de estresse com o uso do aplicativo por um período de uma (1) hora. Várias palavras foram digitadas, reproduzidas e *e-mails* foram enviados apenas utilizando a Interface Cerebral para a seleção dos caracteres e botões. O teste se mostrou satisfatório e o tempo para o usuário digitar palavras e frases diminuiu com o tempo de uso.

O terceiro teste realizado foi o uso do aplicativo com seleções de caracteres e botões simultaneamente por toques na tela e pela Interface Cerebral. Esse teste foi realizado para assegurar que as duas formas de uso não entrariam em conflito em nenhum momento. Este teste também se mostrou satisfatório.

## 4.6 Discussão acerca dos testes realizados

Durante a realização dos testes, alguns pontos foram identificados como futuras melhorias ou modificações para tornar o aplicativo mais eficiente. Entre esses pontos está a ausência de números nas opções de digitação oferecidas. Isto ocorreu porque o aplicativo é baseado em um Quadro Alfabético para comunicação rápida falada que não contempla números. Em trabalhos futuros, uma forma de resolver esse problema será elaborada de forma que não interfira no tempo de uso do aplicativo e nem o torne mais difícil de ser utilizado.

Outro ponto levantado foi a possibilidade do piscar dos olhos se tornar um fator de estresse para o uso contínuo do aplicativo por pessoas que sofrem de ELA. Outras alternativas serão analisadas para uma futura melhoria no aplicativo nesse sentido.

O terceiro ponto seria uma melhoria na programação do aplicativo para tornar o valor de base da detecção de piscadas e do tempo de detecção mutáveis em tempo de execução. Isso é devido ao fato de que as pessoas que sofrem de ELA possuem vários níveis de degradação dos músculos, o que faz necessário que a detecção fornecida pelo sistema deva ser adaptável para cada paciente se tornando menos ou mais sensível de acordo com o caso.

## 5 Considerações Finais

NESTE trabalho foi proposto um novo sistema de comunicação para pessoas que sofrem de doenças degenerativas motoras como a ELA, mas que também pudesse ser utilizado por pessoas que sofrem de mudez ou afonia. A proposta foi de um sistema composto por um aplicativo para dispositivos móveis e um *hardware* simples, de baixo custo e que não causasse incômodo ao usuário.

Analisando as técnicas de aquisição de sinais de EEG atuais, os novos eletrodos que estão chegando ao mercado com preços mais acessíveis e o constante avanço da tecnologia móvel, foi possível desenvolver um sistema portátil e simples de ser usado utilizando um sensor de processamento de sinais de EEG conectado a um eletrodo seco que envia os sinais para um *tablet*, o qual os converte para comandos em um aplicativo de comunicação sonora e por *e-mails*.

Em vista dos resultados obtidos com os testes, pode-se concluir que o que foi proposto pode ser considerado um sistema de comunicação confiável para pessoas que sofrem de deficiências na fala.

Em relação ao *hardware*, por se tratar de apenas um canal de aquisição, ainda é necessário o piscar dos olhos para gerar comandos. Porém o custo-benefício deve sempre ser considerado, pois para se utilizar de técnicas mais avançadas de detecção de comandos em sinais de EEG, seria necessário uma ICM com mais canais e também mais custo envolvido.

Em relação ao aplicativo, o mesmo cumpre o seu papel de trazer a possibilidade de comunicação tanto para pessoas que sofrem de mudez ou afonia, quanto para pessoas com doenças degenerativas motoras como a ELA, além de trazer à pessoa com paralisia a capacidade de comunicação não apenas local com o uso do envio de *e-mails*.

Alguns pontos apresentam oportunidades de melhorias, porém o sistema criado seguindo os métodos propostos neste trabalho atingiu os objetivos provando as suposições apresentadas.

### 5.1 Trabalhos Futuros

Apesar do trabalho ter atingido os objetivos propostos e fornecer um sistema confiável de comunicação, há alguns pontos de melhoria observados para serem contemplados em trabalhos futuros, como:

- O estudo de técnicas mais complexas para aquisição e tradução de sinais de EEG utilizando mais canais de aquisição para isso. Uma proposta seria a de desenvolver um sistema de comunicação portátil com o uso de quatorze (14) eletrodos secos para

aquisição dos sinais e extrair deles ERPs para gerar comandos a fim de escrever palavras no aplicativo de comunicação.

- O estudo da melhor forma de colocar opções de digitação de números e de palavras com acentuação no aplicativo, considerando que o mesmo deverá atender as normas de comunicação da Língua Portuguesa.
- Modificar o aplicativo de forma a introduzir a opção do usuário alterar o valor de base para detecção de piscadas e o tempo de detecção em tempo de execução.
- Estudar uma forma de ampliar as possibilidades de comunicação para uso de redes sociais e outros aplicativos conhecidos de comunicação utilizando a ICM já desenvolvida neste trabalho.
- Utilizar o aplicativo aliado a outras peças de *hardware* para fazer o acionamento de equipamentos externos e até mesmo o controle de uma cadeira de rodas motorizada.
- Obter a aprovação do Comitê de Ética em Pesquisa para realizar um teste de usabilidade do aplicativo, com um grupo de voluntários possuindo a deficiência e outros que não a possuem, a fim de verificar a eficácia do sistema e chegar a um valor de base para detecção das piscadas utilizando métodos estatísticos.
- Realizar estudo sobre a autonomia do sistema para determinar a melhor fonte de alimentação a ser utilizada.
- Transformar o protótipo desenvolvido em um produto a ser comercializado.

## Referências

- ADUR, R. *Sistema de processamento de sinais biomédicos: Módulo didático de eletroencefalograma*. Dissertação (Mestrado) — Universidade Federal de Santa Catarina, 2008. Citado 2 vezes nas páginas 19 e 23.
- ALMEIDA, M. B. Uma introdução ao xml, sua utilização na internet e alguns conceitos complementares. *Scielo*, Universidade Federal de Minas Gerais, 2002. Citado na página 29.
- ANATEL. *Perfil dos Usuários de Celular*. Acesso em 01 mai. 2016. Disponível em: <[http://www.teleco.com.br/ncel\\_usu.asp](http://www.teleco.com.br/ncel_usu.asp)>. Citado na página 15.
- ANDROID. *Android Developer - AsyncTask*. Acesso em 29 mai. 2016. Disponível em: <<https://developer.android.com/reference/android/os/AsyncTask.html>>. Citado na página 48.
- ANDROID. *Android Developer - Handler*. Acesso em 29 mai. 2016. Disponível em: <<https://developer.android.com/reference/android/os/Handler.html>>. Citado na página 49.
- ANDROID. *Android Developer - Text to Speech*. Acesso em 29 mai. 2016. Disponível em: <<https://developer.android.com/reference/android/speech/tts/TextToSpeech.html>>. Citado na página 45.
- ANDROID. *Android Developer - Timer Task*. Acesso em 04 jun. 2016. Disponível em: <<https://developer.android.com/reference/java/util/TimerTask.html>>. Citado na página 56.
- APEKSHA, R. H. M.; PRATHIBHA, K. A novel method for analysis of eeg signals using brain wave data analyzer. *International Journal of Soft Computing and Engineering (IJSCE)*, Blue Eyes Intelligence Engineering and Sciences Publication Pvt. Ltd., 2015. Citado 2 vezes nas páginas 15 e 30.
- BARBOSA, A. F. *Interface Cérebro-Computador: uma abordagem exploratória*. Dissertação (Mestrado) — Centro de Tecnologia, Universidade Federal do Rio Grande do Norte, 2012. Citado 5 vezes nas páginas 15, 18, 20, 21 e 22.
- CALADO, E. F. *O corpo silenciado e a expressão da vida psíquica no sujeito acometido pela Esclerose Lateral Amiotrófica*. Dissertação (Mestrado) — Universidade Católica de Pernambuco, 2010. Citado na página 14.
- CLARO, D. B.; SOBRAL, J. B. M. *Programação em Java*. [S.l.]: Copley Pearson Education, 2008. Florianópolis, SC. Citado 2 vezes nas páginas 27 e 28.
- COSTA, C. A. A aplicação da linguagem de modelagem unificada (uml) para o suporte ao projeto de sistemas computacionais dentro de um modelo de referência. *Scielo*, 2001. Citado na página 38.
- CRONIN, S.; HARDIMAN, O.; TRAYNOR, B. J. Ethnic variation in the incidence of als: a systematic review. *Neurology*, 2007. Citado na página 14.

CRUZ, R. R.; JUNIOR, S. D. d. N. Biometria aplicada a segurança residencial. *Trabalho de Conclusão de Curso como requisito para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas*, Faculdades Integradas do Vale do Ivaí, 2013. Citado na página 26.

DONCHIN, E.; COLES, M. G. H. Is the p300 component a manifestation of context updating? behavioral and brain sciences. Cambridge, 1988. Disponível em: <<http://dx.doi.org/10.1017/S0140525X00058027>>. Citado na página 21.

DONCHIN, E.; SPENCER, K. M.; WIJESINGHE, R. The mental prosthesis: assessing the speed of a p300-based brain-computer interface. *Rehabilitation Engineering, IEEE Transactions on*, 2000. Disponível em: <<http://dx.doi.org/10.1109/86.847808>>. Citado na página 21.

GONÇALVES, J. C. Uso da plataforma android em um protótipo de aplicativo coletor de consumo de gás natural. *Monografia (Especialização em Tecnologia Java)*, Faculdades Integradas do Vale do Ivaí, 2013. Citado 2 vezes nas páginas 28 e 29.

HAAK, M. et al. Detecting stress using eye blinks and brain activity from eeg signals. *Man-Machine Interaction Group*, Delft University of Technology, Delft, Netherlands, 2009. Citado na página 22.

HARLAND, C. J.; CLARK, T. D.; PRANCE, R. J. Remote detection of human electroencephalograms using ultrahigh input impedance electric potential sensors. *Applied Physics Letters*, 2002. Disponível em: <<http://dx.doi.org/10.1063/1.1516861>>. Citado na página 23.

HAWKING, S. *The Computer*. Acesso em 01 mai. 2016. Disponível em: <<http://www.hawking.org.uk/the-computer.html>>. Citado na página 14.

HE, B. *Neural Engineering*. [S.l.]: Kluwer Academic / Plenum Publishers, 2005. Minneapolis. Citado 4 vezes nas páginas 18, 19, 20 e 21.

KANDA, P. A. M. *Análise de wavelets com máquina de vetor de suporte no eletrencefalograma da doença de Alzheimer*. Tese (Doutorado) — Faculdade de Medicina, Universidade de São Paulo, 2012. Citado na página 18.

LAZZARI, C. D. *Estudos de Eletroencefalografia em um ambiente de trânsito simulado*. Dissertação (Mestrado) — Escola de Engenharia, Universidade Federal do Rio Grande do Sul, 2013. Citado 2 vezes nas páginas 15 e 20.

LOPEZ-GORDO, M. A.; SANCHEZ-MORILLO, D.; VALLE, F. P. Dry eeg electrodes. *Sensors*, 2014. Disponível em: <<http://www.ncbi.nlm.nih.gov/pubmed/25046013#>>. Citado 3 vezes nas páginas 15, 22 e 23.

MANOILOV, P. Eeg eye-blinking artifacts power spectrum analysis. *CompSysTech*, 2006. Citado na página 22.

MITCHELL, J. D.; BORASIO, G. D. Amyotrophic lateral sclerosis. *Lancet*, 2007. Citado na página 14.

ORACLE. *Core J2EE Patterns ? Data Access Object*. Acesso em 29 mai. 2016. Disponível em: <<http://www.oracle.com/technetwork/java/dataaccessobject-138824.html>>. Citado na página 40.

PEREIRA, S.; PIZZATO, R.; CARATI, E. G. Conversão de sinais analógicos para digitais. *Pato Branco*, UTFPR, 2010. Citado 2 vezes nas páginas 23 e 24.

ROSA, D. L. *Sistema de processamento de sinais biomédicos: filtragem de sinais de eletroencefalograma*. Dissertação (Mestrado) — Universidade Federal de Santa Catarina, 2009. Citado 4 vezes nas páginas 20, 24, 25 e 26.

SANTOS, H. B.; SILVA, M. V.; STEDILE, R. Sistema de identificação de padrões de ondas cerebrais. *Trabalho de Conclusão de Curso de Graduação como requisito para obtenção do título de Engenheiro Industrial Elétrico*, Departamento Acadêmico de Eletrônica, Universidade Tecnológica Federal do Paraná, 2014. Citado na página 21.

SILVA, D. F. Sistema de comunicação bluetooth utilizando microcontrolador. *Monografia apresentada como requisito parcial para obtenção do diploma de Bacharel em Engenharia da Computação*, Escola Politécnica de Pernambuco, Universidade de Pernambuco, 2009. Citado na página 26.

SILVA, F. A. *Recuperação com base em checkpointing: Uma abordagem orientada a objetos*. Dissertação (Mestrado) — Universidade Federal do Rio Grande do Sul, 2002. Citado na página 27.

SORIANO, D. C. Análise de diferentes técnicas de extração de características em interface cérebro-máquina. *Uberlândia*, Trabalho apresentado ao 24 Congresso Brasileiro de Engenharia Biomédica, 2014. Citado na página 20.

TEPLAN, M. Fundamentals of eeg measurement. *MEASUREMENT SCIENCE REVIEW*, Institute of Measurement Science, Slovak Academy of Sciences, 2002. Citado na página 49.