

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE COMPUTAÇÃO
CURSO DE ENGENHARIA DE COMPUTAÇÃO

BRUNO DA SILVA BRAZIEL

**APLICATIVO MÓVEL PARA REPRESENTAÇÃO GRÁFICA DE
SINAIS ELETROMIOGRÁFICOS**

TRABALHO DE CONCLUSÃO DE CURSO

CORNÉLIO PROCÓPIO
2020

BRUNO DA SILVA BRAZIEL

APLICATIVO MÓVEL PARA REPRESENTAÇÃO GRÁFICA DE SINAIS ELETROMIOGRÁFICOS

Trabalho de Conclusão de Curso apresentado ao Curso de Engenharia de Computação da Universidade Tecnológica Federal do Paraná, como requisito parcial para a obtenção do título de Bacharel em Engenharia de Computação.

Orientador: Prof. Dr. Paulo Rogério Scalassara
Universidade Tecnológica Federal do Paraná

Coorientador: Prof. Dr. Alexandre Rômolo Moreira Feitosa
Universidade Tecnológica Federal do Paraná

CORNÉLIO PROCÓPIO
2020



TERMO DE APROVAÇÃO

Aplicativo Móvel para Representação Gráfica de Sinais Eletromiográficos

por

Bruno da Silva Braziel

Este Trabalho de Conclusão de Curso de graduação foi julgado adequado para obtenção do Título de Bacharel em Engenharia de Computação e aprovado em sua forma final pelo Programa de Graduação em Engenharia de Computação da Universidade Tecnológica Federal do Paraná.

Cornélio Procópio, 29/09/2020

Prof. Dr. Paulo Rogério Scalassara

Professor Orientador

Prof. Dr. Henrique Yoshikazu Shishido

Professor membro da banca

Prof. Dr. Wagner Endo

Professor membro da banca

À toda comunidade preta que lutou durante décadas para que eu pudesse chegar a este momento.

AGRADECIMENTOS

Agradeço primeiramente aos meus pais, que me deram suporte financeiro e emocional durante toda a graduação. Aos colegas que viraram amigos, coletivos de diversidades e professores que me acompanharam, muito obrigado pela força.

Agradeço também à Universidade Tecnológica Federal do Paraná, instituição por me formar um acadêmico e profissional dedicado, bem como por financiaram parcialmente o desenvolvimento deste trabalho trabalho.

Pode haver 100 pessoas em uma sala, e 99 não acreditarem em você, mas você só precisa de uma para mudar sua vida.

- Stefani Germanotta

RESUMO

BRAZIEL, Bruno. Aplicativo Móvel para representação gráfica de Sinais Eletromiográficos. 2020. 32 f. Trabalho de Conclusão de Curso – Curso de Engenharia de Computação, Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2020.

Os avanços na tecnologia afetaram diversos setores na indústria e na vida humana. Com a utilização do monitoramento à distância das funções do corpo humano na medicina, profissionais na área de saúde e pesquisadores são capazes de acompanhar o estado de pacientes sem de fato estarem presentes. A difusão dos smartphones possibilitou que a sociedade se prontificasse para a implementação dos conceitos acerca deste tópico. A eletromiografia é um exemplo de processo que utiliza a leitura e estudo de atividade muscular em diversos campos esportivos e da medicina moderna. Levando tais tópicos em consideração, este trabalho consistiu no estudo de uma solução móvel de análise corporal envolvendo a implementação de um aplicativo capaz de visualizar sinais biológicos. Sua utilização envolve cenários genéricos de análise de estrutura física humana, como por exemplo, simples monitorações eletromiográficas ou eletrocardiográficas.

Palavras-chave: Aplicativo. Android. Processamento de Sinais. Eletromiografia.

ABSTRACT

BRAZIEL, Bruno. Mobile Application to represent Eletromyographic Signals in charts. 2020. 32 f. Trabalho de Conclusão de Curso – Curso de Engenharia de Computação, Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2020.

The technology progress affected several aspects in industry and human life. Due to the use of body function monitoring in medicine field, health specialists and researchers are now capable of monitoring patients' status without being there. The smartphone spreading allowed that society got ready to the implementation of concepts around this topic. Electromyography is an example of procedure that uses muscular activity acquisition and analysis in fields such as physical therapy, dentistry and sports. Taking those topics into consideration, this work intended on studying a body analysis under a mobile solution evolving an application implementation able to present biologic signals. Its utilization involved generics scenarios of human's fisic structure such as simple electromyography or electrocardiography monitoring.

Keywords: Application. Android. Signal Processing. Electromyography.

LISTA DE FIGURAS

Figura 1 – Esquema de captação e visualização do sinal EMG.	2
Figura 2 – Visão da estrutura interna de uma fibra muscular.	4
Figura 3 – Potencial de ação de uma fibra muscular em decorrência do tempo.	5
Figura 4 – Processo de tratamento e análise recomendado a ser feito no sinal bruto.	6
Figura 5 – Interface do sistema Android 9.0	8
Figura 6 – Estrutura do Sistema Android disposta em camadas.	9
Figura 7 – Circuito para recepção e transmissão dos sinais.	11
Figura 8 – Módulo AD8232 juntamente dos eletrodos utilizados para a leitura.	12
Figura 9 – Posicionamento do sistema de três eletrodos (a) e o de 12 derivações (b).	12
Figura 10 – Módulo Bluetooth HC-06	13
Figura 11 – Arduino Uno	14
Figura 12 – Interface de desenvolvimento Android Studio.	15
Figura 13 – Página inicial do aplicativo.	19
Figura 14 – Aplicativo durante (a) e após (b) a leitura do sinal	20
Figura 15 – Lista de leituras realizadas pelo usuário. É possível arrastar para a esquerda para remover uma leitura, e para a direita para exportar os dados no formato .csv.	21
Figura 16 – Representação gráfica de uma leitura que foi exportada no formato .csv e a tabela de valores.	22
Figura 17 – Tela de configurações do aplicativo (a) e seleção de dispositivo emparelhado (b).	23

LISTA DE QUADROS

Quadro 1 – Pinagens utilizadas entre os componentes do circuito.	11
--	----

LISTA DE ABREVIATURAS E SIGLAS

BFB	Biofeedback
ECG	Eletrocardiograma
EMG	Eletromiografia
IDE	<i>Integrated Development Environment</i> Ambiente de Desenvolvimento Integrado
MFAP	<i>Muscle Fiber Action Potential</i> Potencial de Ação da Fibra Muscular
MFCV	<i>Muscle Fiber Conduction Velocity</i> Velocidade de Condução da Fibra Muscular
RMS	<i>Root Mean Square</i> Valor Eficaz
SEMG	Eletromiografia Superficial
SO	Sistema Operacional
USB	<i>Universal Serial Bus</i> Porta Universal

SUMÁRIO

1 – INTRODUÇÃO	1
1.1 OBJETIVOS	2
1.2 ORGANIZAÇÃO DO TRABALHO	2
2 – FUNDAMENTAÇÃO TEÓRICA	3
2.1 ELETROMIOGRAFIA	3
2.1.1 CONTEXTO HISTÓRICO	3
2.1.2 POTENCIAL DE AÇÃO DA FIBRA MUSCULAR	3
2.1.3 COLETA DE DADOS ELETROMIOGRÁFICOS E TRATAMENTO DO SINAL	5
2.1.4 APLICAÇÕES RELACIONADAS A ESTE TRABALHO	6
2.2 APLICAÇÕES MÓVEIS	7
2.2.1 Sistema Operacional Android	7
3 – MATERIAIS E MÉTODOS	10
3.1 RECEPÇÃO E TRANSMISSÃO DOS SINAIS	10
3.1.1 MÓDULO CARDÍACO AD8232	11
3.1.2 MÓDULO BLUETOOTH	13
3.1.3 ARDUINO UNO	13
3.2 DESENVOLVIMENTO DA APLICAÇÃO MÓVEL DE VISUALIZAÇÃO DOS DADOS	14
3.2.1 ANDROID STUDIO	14
3.3 ANÁLISE E DESENVOLVIMENTO	15
4 – ANÁLISE E DISCUSSÃO DOS RESULTADOS	17
4.1 RECEPÇÃO E TRANSMISSÃO DOS SINAIS	17
4.2 DESENVOLVIMENTO DA APLICAÇÃO MÓVEL PARA VISUALIZAÇÃO DOS DADOS	17
5 – CONCLUSÃO	25
5.1 TRABALHOS FUTUROS	25
Referências	27

Apêndices	31
APÊNDICE A–Código Arduino para recepção de sinais	32

1 INTRODUÇÃO

A evolução da tecnologia possibilitou o desenvolvimento de avanços na ramo da medicina e permitiu, conseqüentemente, sua utilização em diversas técnicas para monitoramento, análise e recuperação de pacientes. Um exemplo é a técnica de biofeedback (BFB), que consiste em acompanhar as informações fisiológicas de um paciente e instruí-lo em uma tomada específica de decisão no intuito de melhorar sua saúde física, emocional e neurológica (GIGGINS; PERSSON; CAULFIELD, 2013). O biofeedback é um termo do final dos anos 1960 que remete ao processo de aprendizado da informação fisiológica que é monitorado e “alimentado” (*fed back*) através de instrumentos de BFB (MCKEE, 2008). Isto significa que o paciente que faz uso desta técnica é o principal encarregado de seu bem-estar, uma vez que ele é o agente que servirá para proveniência de dados e irá concomitantemente controlar suas funções comportamentais para aprimorar sua saúde.

Esta técnica pode ser utilizada na fisioterapia, por exemplo, para aliviar dores crônicas (CORNEL et al., 2005) bem como para prática fisioterapêutica na recuperação de mobilidade física após um Acidente Vascular Cerebral (WOODFORD; PRICE, 2007). Há também projetos no meio desportivo, como o estudo de monitoramento e análise de batidas cardíacas durante a prática de esportes (CROCE, 1986) e o efeito do treinamento mental com biofeedback na natação (BAR-ELI; BLUMENSTEIN, 2004). É possível também trabalhar com biofeedback na odontologia (WIESELMANN-PENKNER et al., 2001), na psicologia (REEVES; MEALIEA, 1975) e psicoterapia (ADLER; ADLER, 1976).

Os exemplos citados tomam como um dos meios de aquisição de dados o estudo eletromiográfico. A eletromiografia (EMG) representa a atividade da musculatura de acordo com a manifestação elétrica provinda da ação das fibras musculares (BASMAJIAN, 1962). Este sinal é descrito em termos de amplitude, frequência e fase, em função do tempo (REAZ; HUSSAIN; MOHD-YASIN, 2006).

Com o crescente uso de smartphones, surgem também aplicações móveis capazes de exibir e interpretar sinais de EMG. Em 2018 foi publicado um artigo que descreveu a implementação de um detector de sinais EMG utilizando a porta USB (TSAI; KO, 2018). Um ano depois, foi publicado outro artigo que desenvolveu uma aplicação móvel para estimar o período de latência muscular (KARIMPOUR et al., 2019). O primeiro projeto possui a limitação de que a comunicação entre o sensor e o smartphone é feita através de conexão USB. O segundo projeto, embora trabalhe com comunicação Bluetooth, foi desenvolvido exclusivamente para tratar a latência da musculatura.

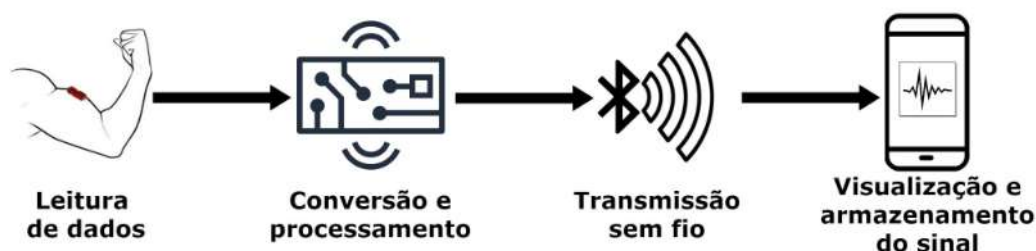
Neste contexto, o propósito deste trabalho foi a implementação de um aplicativo no sistema operacional Android que receba os dados do sensor muscular e apresente um gráfico do sinal lido em tempo real. Ao fim da execução do programa, é exportado um arquivo contendo informações da leitura realizada para que os dados possam ser estudados posteriormente, se

necessário. A intenção deste projeto foi desenvolver um aplicativo que permita a leitura de sinais EMG e não seja necessário o auxílio de um computador, contando apenas com um transmissor de dados sem fio acoplado ao sensor muscular.

1.1 OBJETIVOS

O principal objetivo deste trabalho foi desenvolver um aplicativo compatível com o sistema operacional Android no intuito de representar graficamente sinais eletromiográficos. Um sensor capta a ação muscular e envia os dados via conexão Bluetooth para o aplicativo no *smartphone*. Este, por sua vez, apresenta um gráfico das informações em tempo real e salva o sinal na memória do dispositivo, como representado na [Figura 1](#).

Figura 1 – Esquema de captação e visualização do sinal EMG.



1.2 ORGANIZAÇÃO DO TRABALHO

O [Capítulo 2](#) irá revisar os principais conceitos acerca do tema deste trabalho. Esta seção irá explanar sobre o histórico da Eletromiografia, como o músculo sofre contração e como o sinal de EMG é obtido através deste. Também será discutido como os dados são coletados e tratados, além de apresentar aplicações das técnicas. Em relação ao aplicativo Android, esta seção discorre sobre a arquitetura do sistema operacional e irá apresentar o contexto histórico dos dispositivos vestíveis. O [Capítulo 3](#) apresenta como o projeto foi desenvolvido, sendo dividido na montagem do circuito de leitura dos sinais e desenvolvimento do aplicativo. O [Capítulo 4](#) evidencia as atividades executadas durante o trabalho e, por fim, o [Capítulo 5](#) apresenta as considerações finais acerca do projeto.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta os conceitos relevantes para o trabalho. Iniciando pelos termos acerca da eletromiografia, desde o contexto histórico, até à coleta de dados musculares. Tem-se, também, a definição de aplicações móveis, especialmente no escopo do sistema operacional Android.

2.1 ELETROMIOGRAFIA

A eletromiografia pode ser definida como o estudo da atividade da unidade motora, isto é, um neurônio motor e todas as fibras musculares ramificadas por ele (BUCHTHAL; SCHMALBRUCH, 1980), por meio da análise do sinal elétrico provindo das membranas da musculatura (OLIVEIRA et al., 2012). Tal estudo permite facilitar o entendimento sobre as características bioelétricas da fisiologia humana, as quais são comumente complexas. Consequentemente, a distinção da ação muscular de um conjunto de variáveis, como o tipo de fibra do músculo ou seu comprimento e diâmetro, se torna mais evidente. (KAMEN; GABRIEL, 2010).

2.1.1 CONTEXTO HISTÓRICO

A eletromiografia, em especial medida sobre a superfície da pele (SEMG), teve início com os estudos de movimentos dinâmicos, na década de 1940 (CRAM, 2003). Em 1944, três pesquisadores realizaram uma pesquisa envolvendo a observação da atividade muscular em movimentos de ombro (INMAN; ABBOTT et al., 1996). Onze anos depois, os estudos em EMG alavancaram ainda mais com a publicação de um artigo que analisou a função do músculo *erector spinae* em movimentos específicos e na postura humana (FLOYD; SILVER, 1955). Tal pesquisa auxiliou em estudos na área biomédica, em específico a de dores nas costas, onde com o auxílio dos conceitos de biofeedback foi realizada uma avaliação nos tratamentos de tal condição (WOLF; BASMAJIAN, 1978);(WOLF; NACHT; KELLY, 1982).

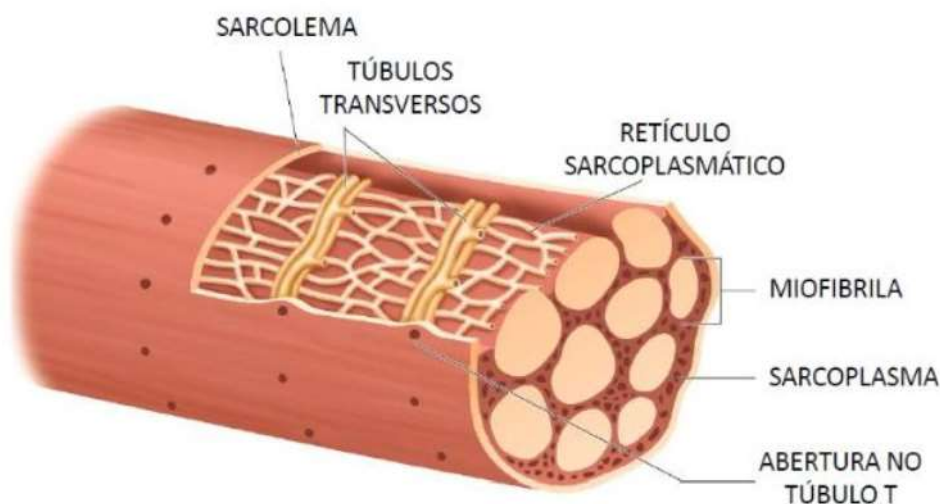
No fim dos anos 60, brevemente após o nascimento do biofeedback, um grupo de pesquisadores clínicos estudaram a unidade motora para captação de estímulos no sistema neuromuscular (BASMAJIAN, 1963), sendo posteriormente aplicada para desenvolver técnicas de relaxamento profundo (GREEN et al., 1969). Alguns anos depois, os conceitos de SEMG foram utilizados no tratamento de dores de cabeça (BUDZYNSKI; STOYVA; ADLER, 1970) e, desde então, as pesquisas envolvendo este campo vêm se expandindo (CRAM, 2003).

2.1.2 POTENCIAL DE AÇÃO DA FIBRA MUSCULAR

A fibra muscular é composta por um tecido envolto por uma membrana chamada sarcolema que, em intervalos regulares de tempo sofre interrupção pelo sistema tubular

transversal (Túbulo-T), conforme a [Figura 2](#). O gradiente de tensão (medido em milivolts) do sarcolema em repouso é de aproximadamente -90mV em relação à parte externa da fibra, sofrendo variação de acordo com os níveis de concentração de sódio, potássio, cloro e outros ânions que cercam a membrana. Uma medida pode ter sinal negativo visto que é utilizado o gradiente de tensão, portanto a direção é levada em consideração ([KAMEN; GABRIEL, 2010](#)).

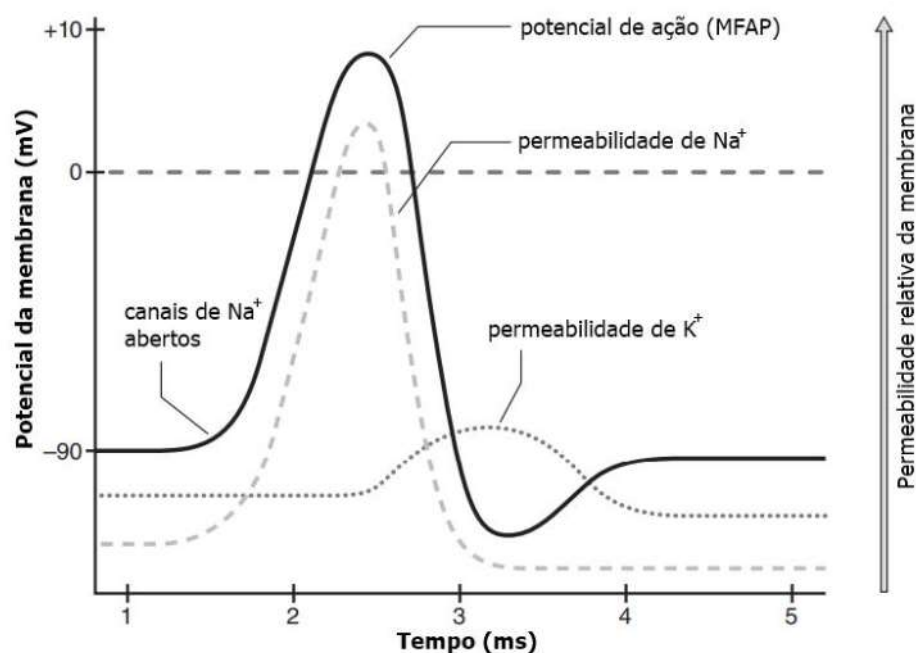
Figura 2 – Visão da estrutura interna de uma fibra muscular.



Fonte: ([WILMORE; COSTILL; KENNEY, 2001](#))

O potencial de ação da fibra muscular (MFAP) é uma resposta à despolarização (para aproximadamente 10mV ou mais) das fibras musculares. Isto é, assim que gerado nas junções neuromusculares, este potencial eleva a permeabilidade de sódio (Na^+) que percorre o músculo e reverte a polaridade das células, tornando o gradiente de tensão 10mV positivo. Ao fim do aumento de Na^+ , ocorre a elevação nos níveis de permeabilidade do potássio (K^+), que resulta no retorno do potencial da membrana ao seu estado de descanso ([KAMEN; GABRIEL, 2010](#)), como ilustrado na [Figura 3](#).

Figura 3 – Potencial de ação de uma fibra muscular em decorrência do tempo.



Fonte: (KAMEN; GABRIEL, 2010)

2.1.3 COLETA DE DADOS ELETROMIOGRÁFICOS E TRATAMENTO DO SINAL

A aquisição das informações EMG se dá através do registro da despolarização das fibras musculares juntamente de sinais irrelevantes, denominados ruídos (APARICIO, 2005). Isto é, a coleta eletromiográfica pode ser considerada uma prática de monitoramento dos potenciais de ação das membranas (MFAP) por meio de um sinal que sofre influência de variáveis fisiológicas, neurológicas, instrumentais, musculares e anatômicas. Alguns exemplos de fatores que interferem na formação do sinal EMG é o efeito da Velocidade de Condução da Fibra Muscular (MFCV), que é a velocidade pela qual o potencial de ação irá percorrer pela fibra (BECK, 2006), a escolha da unidade motora a ser analisada e a variância de MFAP nas unidades motoras (ENOKA, 2000).

O principal dispositivo para tal prática é o eletromiógrafo, que irá apresentar um sinal EMG de formato analógico (contínuo no tempo) a ser convertido para digital (sinal com valores em intervalos definidos de tempo). É importante levar em consideração cinco fatores para tratamento do sinal (MARCHETTI; DUARTE, 2006):

- Frequência de amostragem, para definir a leitura de um valor do sinal de frequência em um determinado momento. Embora o SEMG possua frequências na faixa de 400 e 500 Hz, a frequência de amostragem mínima considerada é cerca de 1000 Hz. Isto ocorre devido ao teorema de Nyquist, que diz que para reconstruir devidamente o sinal EMG, deve-se utilizar uma frequência de amostragem que seja o dobro da sua maior frequência;
- Eletrodos, que serão os dispositivos de recepção dos sinais elétricos. Estes podem ser aplicados como agulha ou fio (em casos de músculos profundos), na superfície da pele

(para músculos mais externos), ou de malha (para coletar características de propagação MFAP em um conjunto de fibras musculares);

- Amplificadores uma vez que o sinal EMG possui uma baixa amplitude, faz-se uso de amplificadores. Entretanto, deve-se atentar às características do sinal na intenção de não modificá-lo a tal ponto de amplificar também os ruídos, que são os sinais indesejáveis;
- Filtros para diminuir as variações indesejadas nas frequências e permitir que características relevantes para análise sejam evidenciados. Estes são úteis em situações onde o sinal sofreu interferência, possui ruído ou teve contaminação com outro sinal, bem como em casos de distorção, onde o mesmo deve ser reparado;
- Conversor analógico/digital para discretizar as amplitudes encontradas pelo sensor. Os conversores são os responsáveis na definição da frequência de amostragem.

Ao fim destes fatores podem ser realizadas diferentes técnicas de análise do sinal:

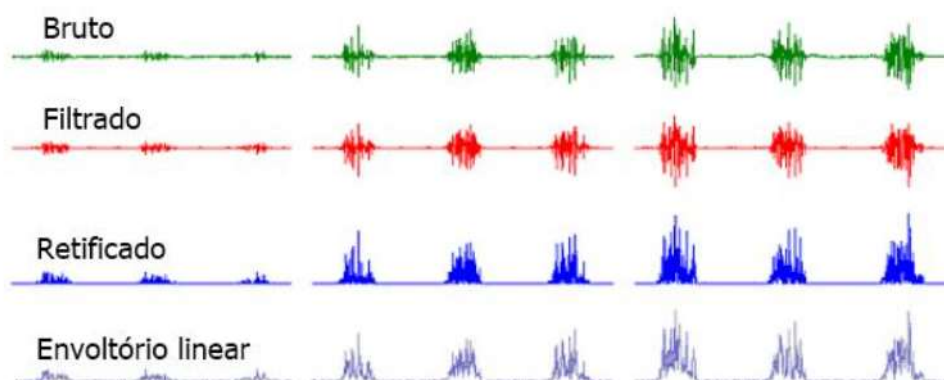
- Retificação designada para remover os valores negativos do sinal bruto.
- Envoltório linear que irá indicar a magnitude do sinal EMG.
- *Root Mean Square* (RMS), como mostrado na [Equação \(1\)](#), avalia o nível de atividade do sinal, analisando suas mudanças em função do tempo, não sendo necessário retificação.

$$RMS = \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2} \quad (1)$$

Sendo N a quantidade de amostras e x_i os valores de amplitude.

Um processo similar foi realizado no exemplo dado na [Figura 4](#).

Figura 4 – Processo de tratamento e análise recomendado a ser feito no sinal bruto.



Fonte: ([ALTIMARI et al., 2012](#))

2.1.4 APLICAÇÕES RELACIONADAS A ESTE TRABALHO

As aplicações eletromiográficas na tecnologia moderna, em especial nos aplicativos móveis, auxiliam a melhora da saúde humana de diferentes formas. Em 2014 foi publicado um artigo sobre um aplicativo móvel que faz uso de análise ECG (eletrocardiograma) e EMG para praticantes de ciclismo ([RICHER et al., 2014](#)). Há também uma rede corporal baseada no sistema

operacional Android com o intuito de registrar sinais ECG e enviá-los para aplicações de saúde (ALTINI; PENDERS; ROEBBERS, 2010). Tais projetos podem capacitar o monitoramento remoto de pacientes alavancando a prática da telemedicina (SARAVANAN, 2014).

A eletromiografia e os aplicativos móveis também podem auxiliar na acessibilidade humana, permitindo que uma pessoa paraplégica fique em pé e controle seus passos pelo celular (AHMED et al., 2018). Estudos similares foram capazes de auxiliar no processo de reabilitação de pacientes através de biofeedback, permitindo que a interação paciente-doutor ocorra através de um aplicativo Android (YASSIN et al., 2017).

2.2 APLICAÇÕES MÓVEIS

A crescente utilização de smartphones permitiu uma maior versatilidade no desenvolvimento de aplicações em diversas áreas. Atrelada à tecnologia *wearable*¹, os dispositivos são capazes de monitorar diversas informações de seus usuários, como localização, contagem de passos, batimentos e pressão cardíaca (PATEL; ASCH; VOLPP, 2015).

Com o lançamento em 2007 do iPhone pela Apple, o uso de aplicativos se tornou onipresente devido à experiência dos usuários de utilizarem os recursos de um computador pessoal (como calculadora, calendário e bloco de notas) em um dispositivo que cabe no bolso (RAKESTRAW; EUNNI; KASUGANTI, 2013). Entretanto, o crescimento na distribuição e desenvolvimento das aplicações aconteceu apenas com a entrada de outras empresas no ramo.

Desta forma os smartphones foram introduzidos ao mercado com o intuito de oferecer funcionalidades além de chamadas telefônicas e envio de mensagens de texto. Atualmente, o sistema operacional móvel mais utilizado no ramo é o Android, com cerca de 2.5 bilhões dispositivos dentre smartphones, tablets, vestíveis e dispositivos para TV (THE VERGE, 2019).

2.2.1 Sistema Operacional Android

Baseado no kernel do Linux, o Projeto Open Source Android é um sistema operacional (SO) desenvolvido pela empresa Google que teve como principal foco os dispositivos touchscreen (ALIFERI, 2016). Ela foi originada pela startup de mesmo nome e, em 2005 a Google comprou o projeto e contratou seu time de desenvolvimento (DIMARZIO, 2016). Este sistema é *open source*, isto é, seu código-fonte é disponibilizado sem custo de licença, sendo cedida sob uma licença criada pela empresa Apache Software Foundation (ASF). Este tipo de liberdade possibilita que terceiros utilizem o sistema em seus projetos, tanto internos quanto comerciais. Este é o caso da Motorola e da Sony Ericsson, por exemplo, que desenvolvem suas próprias customizações do Android. A Figura 5 apresenta a interface do Android em sua versão 9.0, denominada *Pie*.

¹Equipamentos tecnológicos vestíveis, como por exemplo, relógios inteligentes que podem ser conectados ao *smartphone* para leitura de notificações, monitoramento cardíaco, contagem de passos entre outras funcionalidades.

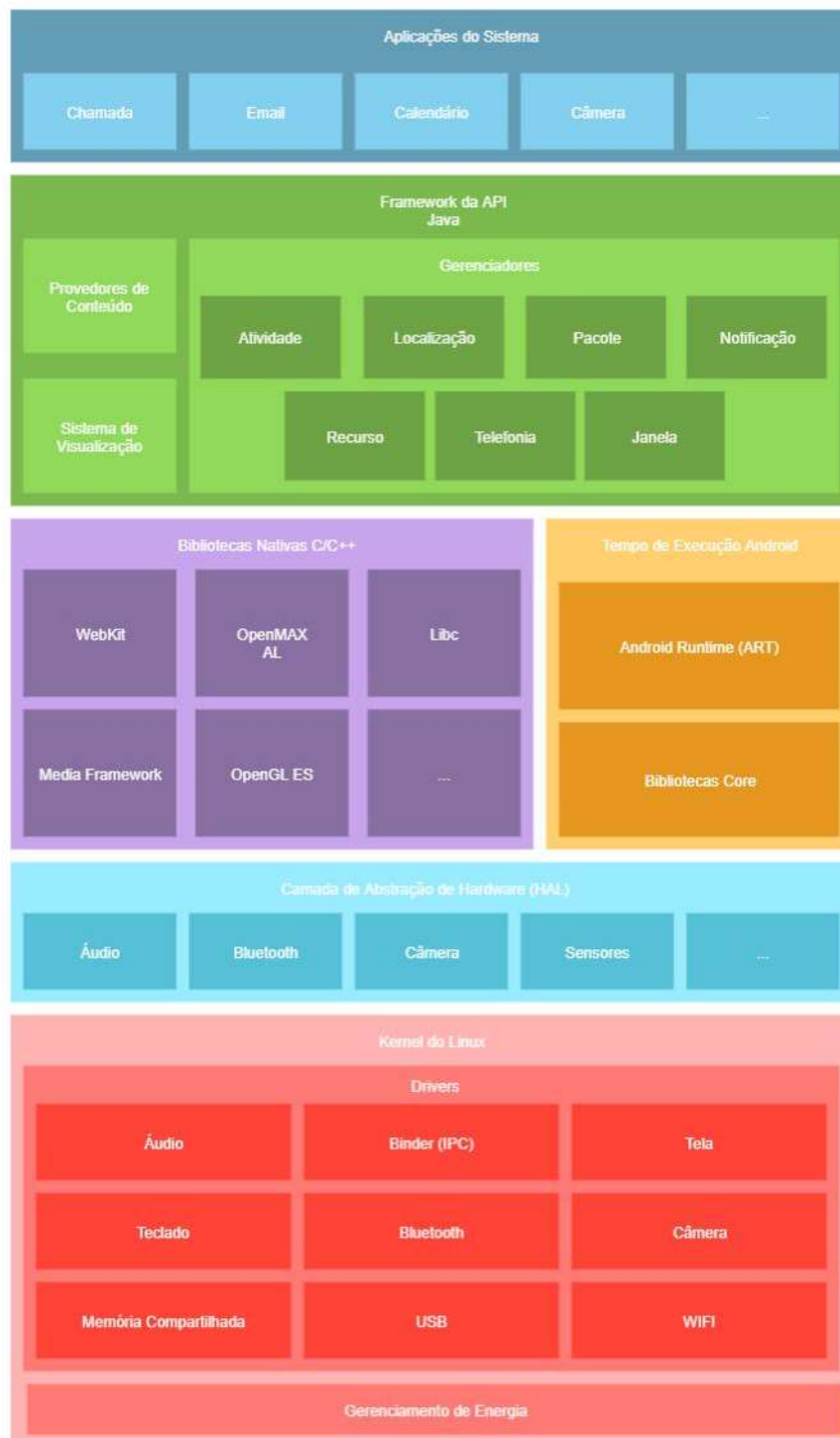
Figura 5 – Interface do sistema Android 9.0



Fonte: (TECNOBLOG, 2018)

Como mostrado na [Figura 6](#), sua arquitetura conta com cinco camadas: o kernel do Linux, que é onde o SO é baseado; a camada de abstração de hardware (HAC), que intermedia as camadas de software e hardware; as bibliotecas, que possuem os códigos dos principais recursos do sistema; o tempo de execução Android, que provêm um conjunto de bibliotecas para desenvolvimento de aplicativos; o framework de aplicação, que expõe todas as capacidades do sistema para os desenvolvedores de aplicações; e as aplicações em si, no topo da camada, que são os programas que serão executados dentro do SO ([GOOGLE LLC, 2019a](#)).

Figura 6 – Estrutura do Sistema Android disposta em camadas.



Fonte: (GOOGLE LLC, 2019a)

3 MATERIAIS E MÉTODOS

Este capítulo apresenta o desenvolvimento do trabalho proposto, dispondo dos materiais, técnicas e procedimentos realizados. O projeto foi dividido em duas partes: a recepção e transmissão dos sinais eletromiográficos e o desenvolvimento do aplicativo de visualização dos dados. A [Seção 3.1](#) apresenta a montagem do circuito e os seus componentes. A [Seção 3.2](#) dispõe dos materiais e recursos utilizados para desenvolvimento da aplicação. Por fim, a [Seção 3.3](#) descreve os passos de cada etapa do projeto.

3.1 RECEPÇÃO E TRANSMISSÃO DOS SINAIS

O processo de aquisição do sinal eletromiográfico foi previsto de acontecer com o Sensor Myoware, que seria conectado à uma placa de Arduino Uno no intuito de receber os dados. Entretanto, devido à fatores externos, o projeto não pôde ser desenvolvido utilizando o sensor muscular. Para se ater ao escopo de sinais biológicos, foi utilizado um sensor cardíaco, devido ao seu baixo custo quando comparado ao sensor muscular. Diante disso, foi utilizado o módulo AD8232 com três eletrodos.

A placa de Arduino, por sua vez, se comunica com um módulo Bluetooth HC06 para transmitir as informações para o celular, conforme mostra a [Figura 7](#). Entretanto, a tensão de operação do Arduino é de $5V$ enquanto o módulo trabalha com $3.3V$ no pino de transmissão R_x . Desta forma, é necessário realizar o cálculo de divisão de tensão.

Considerando T_a a tensão de operação do Arduino, e R_1 e R_2 as resistências do processo, tem-se a seguinte equação:

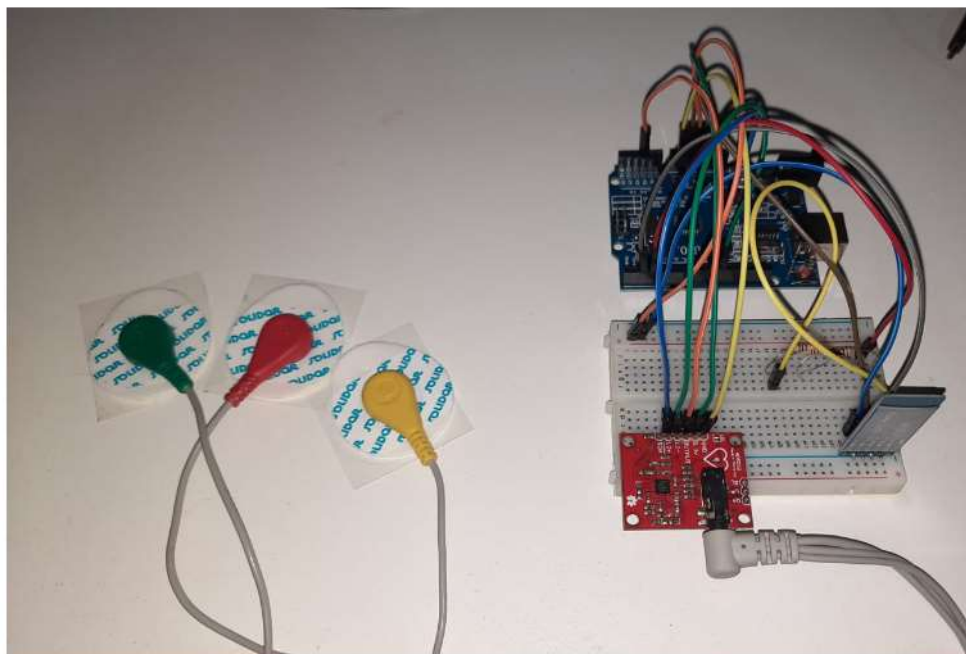
$$\frac{T_a \times R_2}{(R_1 + R_2)} = 3.3V \quad (2)$$

Utilizando um resistor de $1k\Omega$ e sabendo que $T_a = 5V$:

$$\frac{5V \times R_2}{(1k\Omega + R_2)} = 3.3V \quad (3)$$

Portanto, R_2 deve ser aproximadamente $2k\Omega$. Sendo assim, o segundo resistor utilizado no circuito terá este valor.

Figura 7 – Circuito para recepção e transmissão dos sinais.



Deve-se, também, atentar para qual pinagem do Arduino o sensor será conectado. A [Quadro 1](#) apresenta as pinagens entre os módulos e o Arduino UNO. É importante considerar a divisão de tensão entre o módulo Bluetooth e a placa.

Quadro 1 – Pinagens utilizadas entre os componentes do circuito.

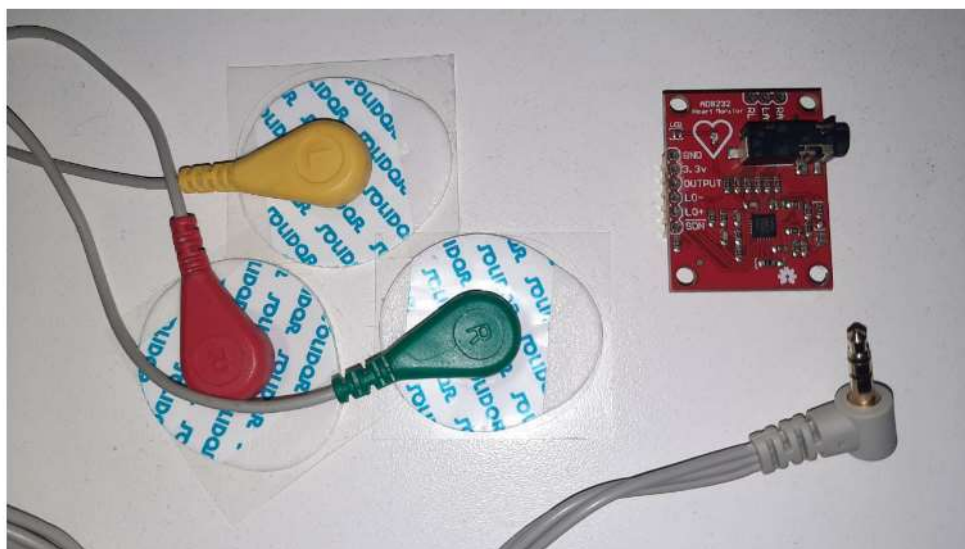
COMPONENTE	ORIGEM	DESTINO no Arduino UNO
Módulo Bluetooth HC-06	RXD	TX
	TXD	RX
	GND	GND
	VCC	5V
Módulo cardíaco AD8232	GND	GND
	3.3v	3.3V
	OUTPUT	A2 (Analógico)
	L0-	10 (Digital)
	L0+	11 (Digital)

3.1.1 MÓDULO CARDÍACO AD8232

O processo de tratamento do sinal dá-se através do módulo AD8232, apresentado na [Figura 8](#). Este sensor é utilizado para medir tanto a atividade elétrica do coração (sinais de eletrocardiograma - ECG) quanto outras respostas biopotenciais ([ANALOG DEVICES, 2012](#)). O valor lido é enviado em forma de sinal analógico para o amplificador que, posteriormente é filtrado para remover os ruídos. O sensor é capaz de lidar com mudanças bruscas de sinais sem afetar a performance do amplificador por meio de ajustes automáticos para um filtro mais

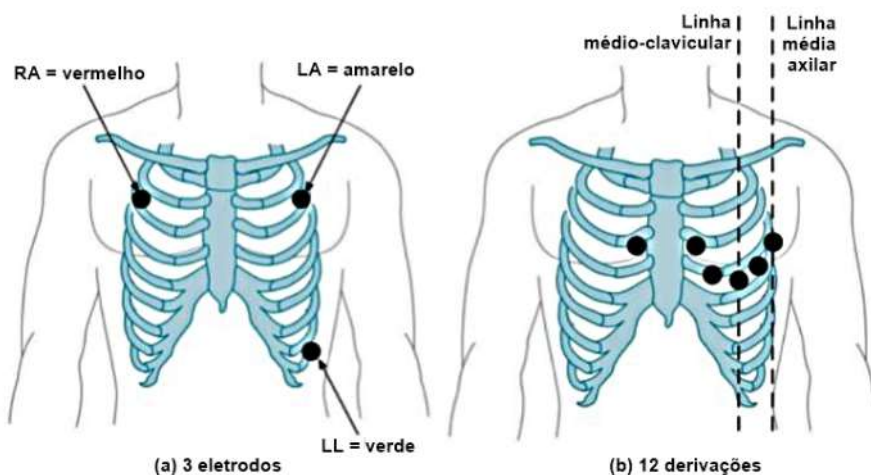
alto. Assim, o módulo inicia a leitura quase que no exato momento em que os eletrodos são conectados, se tornando um ótimo componente para aferição em cenários em que a leitura de sinais deve ser rápida.

Figura 8 – Módulo AD8232 juntamente dos eletrodos utilizados para a leitura.



Para a captura de sinais cardíacos é utilizado o sistema de três eletrodos, posicionados em regiões do peito que são equidistantes do coração, como pode ser visto na Figura 9. O padrão de localização dos componentes é seguido para garantir a acurácia da leitura dos sinais. No caso de testes mais complexos, pode ser utilizado o sistema de 12 derivações, capaz de monitorar todos os compartimentos do coração (POUNDS, 2018). As derivações permitem medir a diferença de potencial elétrico lido entre dois pontos.

Figura 9 – Posicionamento do sistema de três eletrodos (a) e o de 12 derivações (b).

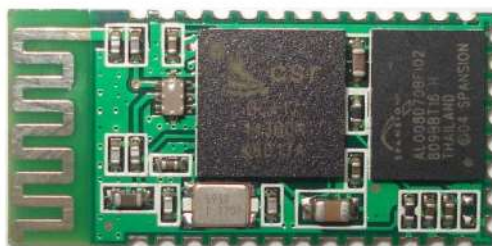


Fonte: (POUNDS, 2018)

3.1.2 MÓDULO BLUETOOTH

Para transmissão dos dados recebidos pelo Arduino, utiliza-se o módulo Bluetooth de modelo HC-06, representado na [Figura 10](#), que conta com a versão 2.0 do Bluetooth. Ele trabalha apenas no modo escravo, ou seja, os dispositivos bluetooth conseguem se conectar com ele, mas ele não consegue realizar o caminho contrário ([GUANGZHOU HC INFORMATION TECHNOLOGY, 2006](#)). O alcance de seu sinal é de até 10 metros e a taxa máxima de transmissão serial é de 1,3824 Mbps.

Figura 10 – Módulo Bluetooth HC-06



Fonte: ([GUANGZHOU HC INFORMATION TECHNOLOGY, 2006](#))

3.1.3 ARDUINO UNO

Arduino é uma plataforma de open-source utilizada para desenvolvimento de projetos eletrônicos ([MCROBERTS, 2015](#)). O seu funcionamento demanda de montagem da placa eletrônica e programação de ambiente de desenvolvimento integrado próprio, denominado Arduino Software. A linguagem de programação utilizada pelo ambiente é o C++, sendo possível trabalhar com classes e desenvolvimento de interfaces gráficas ([ARDUINO, 2019b](#)) .

O modelo utilizado neste projeto é o Uno, mostrado na [Figura 11](#). Baseado no microcontrolador ATmega328, a placa conta com uma entrada para a fonte de alimentação e uma saída para conexão USB. Ele é compatível com os protocolos de comunicação SPI e I2C, além de possuir comunicação serial nos pinos RX e TX. O diferencial deste modelo são suas dimensões de 68,6x53,4mm, modelado para projetos pequenos ([ARDUINO, 2019a](#)).

Figura 11 – Arduino Uno



Fonte: ([ARDUINO, 2017](#))

3.2 DESENVOLVIMENTO DA APLICAÇÃO MÓVEL DE VISUALIZAÇÃO DOS DADOS

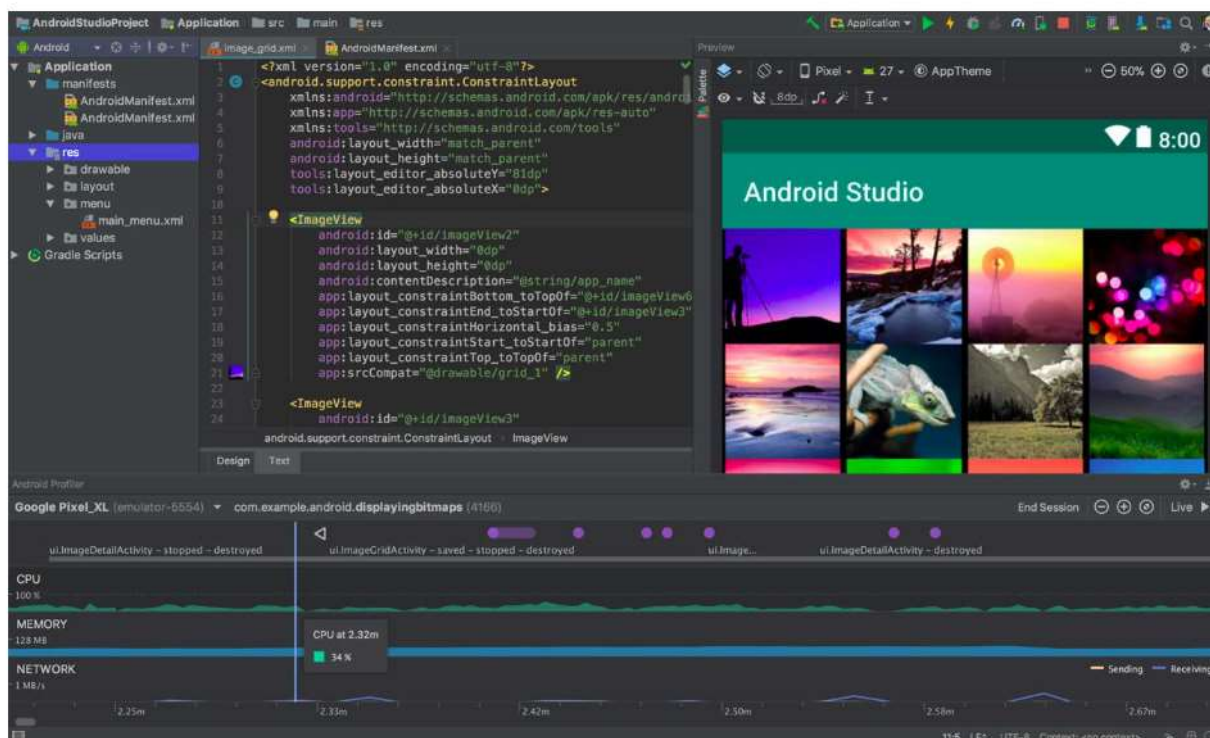
A visualização dos dados é feita através de um aplicativo móvel compatível com o sistema operativo Android. Através da comunicação Bluetooth, o aplicativo recebe os dados provindos do sensor e desenha o gráfico que representa os dados.

3.2.1 ANDROID STUDIO

O desenvolvimento do aplicativo foi planejado para acontecer na IDE Android Studio, assim como mostra a [Figura 12](#). Embora seja executável em todos os sistemas operacionais mais comuns, como Windows, Mac OS X e Linux, é recomendável que o computador possua no mínimo 4GB de memória RAM e 2GB de espaço no disco rígido ([GOOGLE LLC, 2019c](#)).

O Android Studio permite que a programação seja em Java ou Kotlin, além de utilizar o compilador Gradle para gerar o executável do projeto. Tanto o Gradle quanto o simulador do Android não necessitam do Android Studio para trabalho, ou seja, é possível compilar e executar tais projetos sem a IDE ([GOOGLE LLC, 2019b](#)).

Figura 12 – Interface de desenvolvimento Android Studio.



Fonte: (GOOGLE LLC, 2019c)

3.3 ANÁLISE E DESENVOLVIMENTO

Para o desenvolvimento do projeto proposto, foram seguidas as seguintes etapas:

- Montagem do circuito de recepção dos sinais;
- Desenvolvimento do código de transmissão do sinal via Bluetooth;
- Levantamento de requisitos para o aplicativo móvel;
- Desenvolvimento do projeto.

Na primeira etapa é montado o circuito que recebe os dados dos sinais e transmiti-los via comunicação sem fio. O sensor é conectado ao Arduino que, por sua vez, fez ligação à uma *protoboard* no intuito de realizar a divisão de tensão ao se utilizar o módulo Bluetooth. Dois resistores farão a ligação em série e, entre eles, o receptor do módulo.

A segunda etapa consistiu em desenvolver o código que recebe o sinal do sensor e o transmite via comunicação Bluetooth. O algoritmo será implementado através da plataforma IDE Arduino Software, utilizando a linguagem de programação C.

Na terceira etapa foram levantados quais recursos serão implementados no projeto. O propósito neste momento é estudar os requisitos da aplicação, ou seja, quais funções estarão disponíveis no aplicativo.

Finalizado o levantamento de requisitos, é iniciado o desenvolvimento do projeto. O plano foi a utilização o Android Studio como plataforma de programação, juntamente da linguagem de programação Java. O teste do aplicativo, ainda em desenvolvimento, se deu por

meio do *smartphone* do próprio autor do trabalho.

4 ANÁLISE E DISCUSSÃO DOS RESULTADOS

A execução deste trabalho seguiu, em sua grande parte, o cronograma estipulado. As etapas de montagem do circuito, desenvolvimento do código para receber os sinais e transmiti-los via conexão Bluetooth e teste do esquema foram realizados nos intervalos de Novembro de 2019 e Janeiro de 2020. A prototipação de tela, bem como a implementação do aplicativo para leitura de dados foi realizada de Fevereiro de 2020 até Maio de 2020. Os testes para analisar o comportamento da aplicação se deu durante todo o processo de desenvolvimento.

No decorrer do projeto, foram encarados desafios, tanto na esquematização do circuito, quanto na programação do aplicativo. Os ajustes realizados serão descritos nas próximas seções.

4.1 RECEPÇÃO E TRANSMISSÃO DOS SINAIS

O plano inicial foi a representação gráfica dos sinais provindos do sensor muscular Myoware. Entretanto, não foi possível utilizar os módulos provindos do laboratório. Isto se deu uma vez que o país se encontra em um estado de quarentena, impossibilitando a locomoção para testes. Para contornar a situação, é utilizado o sensor de frequência cardíaca AD8232, pelo motivo de ser financeiramente mais acessível que o sensor Myoware e por ser um módulo que ainda respeita a premissa da proposta deste trabalho. Desta forma, foi possível realizar testes do aplicativo em cenário real.

O código para recepção dos dados e transmissão via conexão Bluetooth foi implementado de forma que, quando possível utilizar o sensor muscular Myoware, nada precise ser alterado. Ele pode ser encontrado no [Apêndice A](#).

4.2 DESENVOLVIMENTO DA APLICAÇÃO MÓVEL PARA VISUALIZAÇÃO DOS DADOS

O desenvolvimento do aplicativo, embora proposto para ser utilizado utilizando a linguagem Java por meio do Android Studio, se mostrou desafiadora no momento de implementação do recurso de comunicação via Bluetooth. Isto se deu pois a biblioteca planejada a ser utilizada (BluetoothSPPLibrary ([GITHUB, 2014](#))) não recebe atualizações há mais de cinco anos, apresentando falhas na recepção dos sinais provindos do Arduino, fator crucial para a finalização do projeto. A documentação oficial do Android para desenvolvedores para utilização do componente Bluetooth se mostrou confusa em certos pontos, em especial no que diz respeito à recepção dos sinais, tratando vagamente do tópico. Outro ponto contra desta abordagem é a necessidade do projeto de trabalhar com *Threads*², uma vez que o aplicativo iria receber os dados por uma função e a representação do gráfico se daria em outra

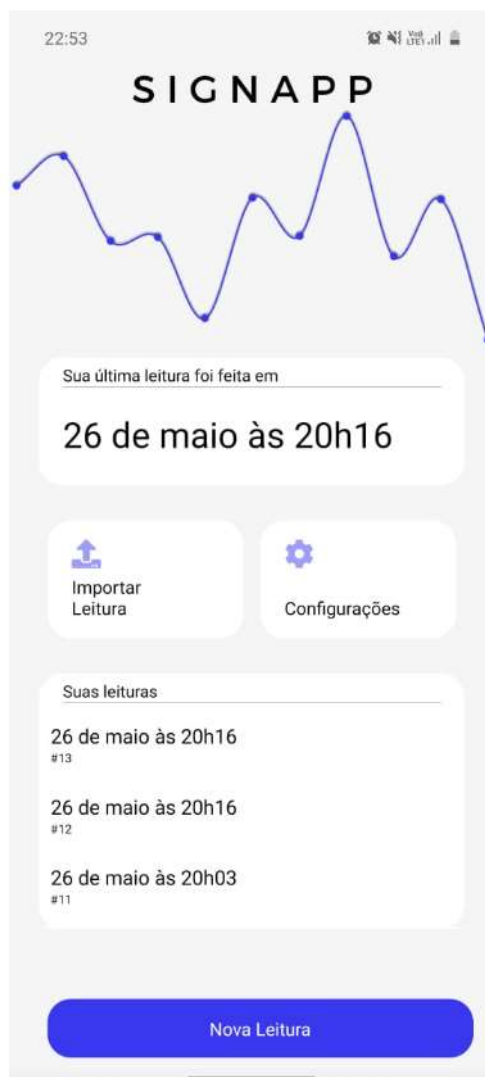
²Conceito de programação paralela onde o código é dividido em tarefas que são gerenciadas para executar de modo concorrente. Fonte: ([OAKS; WONG, 2004](#))

simultaneamente. Consequentemente, o nível de complexidade do código aumentaria, tornando difícil sua manutenção.

Sendo assim, o primeiro passo para a implementação do aplicativo foi encontrar um framework capaz de lidar com conexão Bluetooth de forma rápida e de fácil manutenção. Foi escolhido trabalhar com o plugin chamado react-native-bluetooth-serial-next (NPM, 2019) que, embora não receba mais atualizações de seu criador, se mostrou efetivo na recepção de dados. Este pacote de recursos foi desenvolvido para trabalhar com o framework React Native, biblioteca na linguagem Javascript capaz de criar aplicativos híbridos, ou seja, projetos de software capazes de serem executados tanto no sistema operacional Android quanto no iOS de forma nativa (NPM, 2020). Desta forma, o ambiente Android Studio foi retirado do escopo de desenvolvimento bem como a linguagem de programação Java, que seria utilizada para a criação do aplicativo.

A página inicial do aplicativo, como pode ser visto na [Figura 13](#), apresenta os principais componentes implementados. É possível saber quando foi realizada a última leitura (caso não seja a primeira, também será informado), importar leituras já feitas, acessar o menu de configurações e saber as três últimas leituras executadas. Ao final do aplicativo tem-se um botão para iniciar um novo processo de leitura.

Figura 13 – Página inicial do aplicativo.



Ao pressionar o botão de *Nova Leitura* [Figura 14](#) mostra a execução da leitura em tempo real, sendo possível parar a qualquer momento. Após finalizado, o aplicativo permite que uma nova execução seja feita, bem como o usuário pode optar por exportar um arquivo no formato `.csv`³ contendo as informações, visualizar os dados obtidos ou se direcionar à lista de leituras, apresentada na [Figura 15](#).

³*Comma Separated Values*: formato de arquivo onde os valores são separados por vírgula.

Figura 14 – Aplicativo durante (a) e após (b) a leitura do sinal



Tomando como base React Native, o banco de dados escolhido foi o Realm Database (REALM, 2014). O principal motivo é a sua capacidade de trabalhar com o conceito de offline-first, isto é, a habilidade de lidar com aplicações que não dependam de conexão com a internet. Sendo assim, caso haja alterações no banco de dados do aplicativo, as informações serão utilizadas na próxima vez em que o smartphone for conectado à Internet. No escopo deste projeto, o banco fica disponível no armazenamento móvel do celular, não sendo necessário conexão com a internet para gerenciar os valores contidos em sua base de dados.

Figura 15 – Lista de leituras realizadas pelo usuário. É possível arrastar para a esquerda para remover uma leitura, e para a direita para exportar os dados no formato .csv.



O recurso de exportação da leitura para o formato .csv permite que as informações lidas pelo sensor sejam utilizadas em outras plataformas, trabalhando com análise e tratamento de dados, por exemplo. Há também a capacidade de realizar a leitura de arquivos .csv que sejam exportados pelo aplicativo.

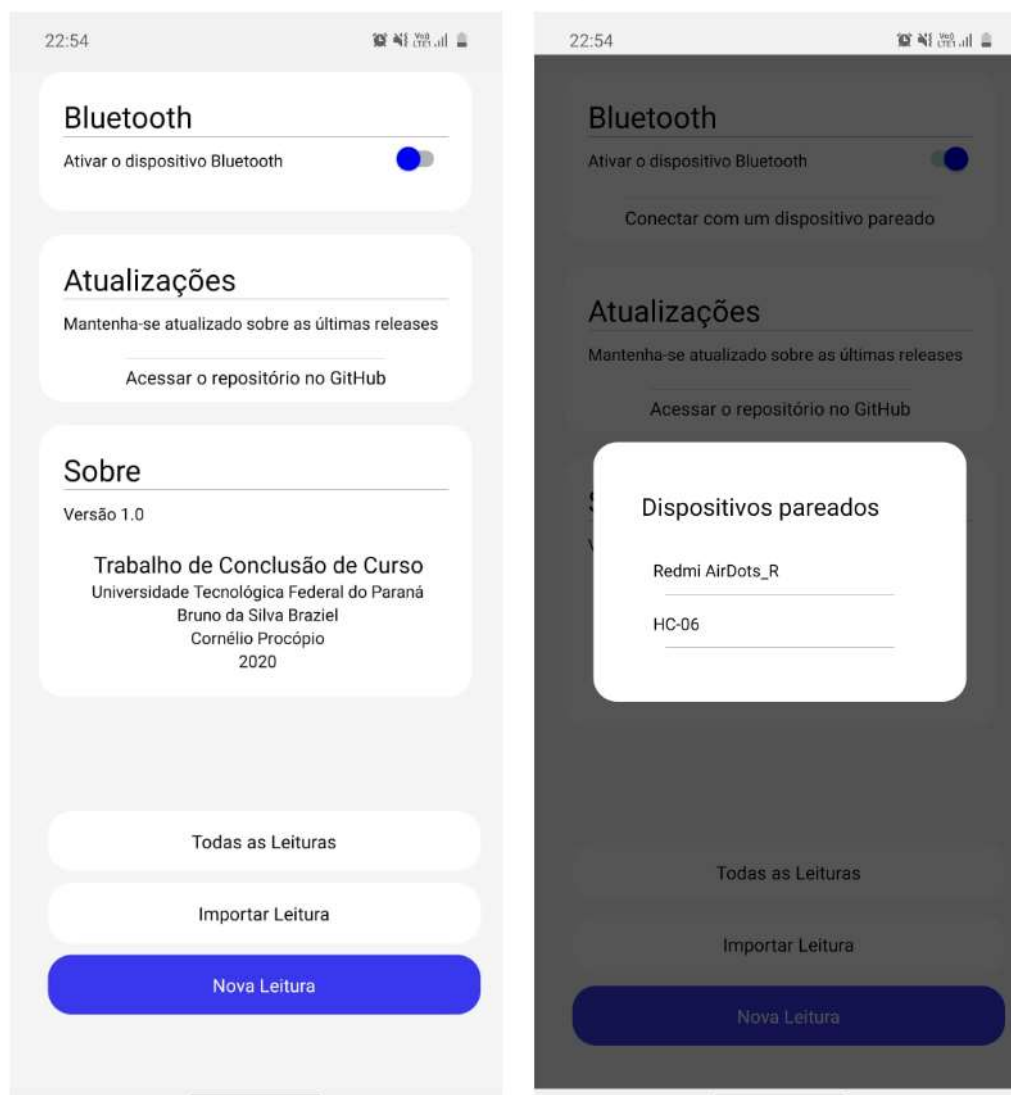
Arquivos exportados durante 20 segundos de leitura possuem tamanho de aproximadamente 2.4 Kilobytes e contêm cerca de 100 valores. Em leituras de 2 minutos de execução, por exemplo, o tamanho dos arquivos era de cerca de 20 Kilobytes. A [Figura 16](#) exibe um arquivo que foi exportado e posteriormente inserido para leitura. Foi notado, entretanto, que para arquivos com um número maior de valores, o processamento dos dados para representação gráfica se tornou mais lenta que o tempo padrão.

Figura 16 – Representação gráfica de uma leitura que foi exportada no formato .csv e a tabela de valores.



O componente de Ajustes, apresentado na [Figura 17](#), permite que o usuário habilite ou desabilite o componente de Bluetooth, se conecte ao módulo HC-06 e também que acesse o repositório do projeto na plataforma GitHub. O aplicativo é capaz de analisar se há uma conexão estabelecida com o módulo antes de abrir a tela de captura de dados.

Figura 17 – Tela de configurações do aplicativo (a) e seleção de dispositivo emparelhado (b).



A seguir são apresentados os principais impasses encontrados durante o desenvolvimento do projeto:

- Utilização de um módulo cardíaco ao invés de um módulo muscular. É interessante ressaltar que devido ao escopo genérico para leitura de dados, o aplicativo é capaz de trabalhar com qualquer tipo de sinal provindo através de uma placa Arduino com módulo Bluetooth.
- Lentidão apresentada durante e após a execução da leitura de dados em longos intervalos de tempo. Isto se dá pois a cada nova informação processada, a plotagem gráfica era executada novamente e, após minutos de leitura, a quantidade de dados a serem desenhados demandava tempo.
- A comunicação sem fio não transporta todos os dados. Diante do atraso de tempo entre a leitura do dado pelo módulo, sua transmissão via Bluetooth e a plotagem e inserção no banco de dados, certas informações eram lidas pelo sensor mas não eram processadas

pelo aplicativo. Ao diminuir este atraso, foi notado que o aplicativo não conseguia suprir a demanda de dados para processar, mesmo com a utilização de técnicas de processamento paralelo (*Threads*).

O projeto deste trabalho pode ser encontrado no repositório do GitHub denominado SignApp JS (disponível no link <https://github.com/brbrazil/signappjs>).

5 CONCLUSÃO

Levando em conta que a intenção do projeto é implementar uma aplicação capaz representar graficamente biológicossinais, o propósito deste trabalho foi cumprido. O aplicativo contribui não apenas para uma análise rápida por meio de sensores, mas também como intermediador no compartilhamento de tais dados entre diferentes plataformas, o que torna possível estudar e monitorar a atividade corporal humana.

Este trabalho se propôs a desenvolver uma aplicação móvel capaz de representar sinais EMG de forma gráfica. A intenção é levantar um projeto que unisse o escopo biológico (no caso, a contração muscular) com a praticidade da tecnologia. Devido a fatores externos, não foi possível testá-lo com sensores responsáveis pela captura de sinais musculares. Entretanto, o projeto se tornou efetivo quando tratando de sinais cardíacos. Isto significa que a intenção foi suprida com sucesso e, uma vez que o sensor muscular seja inserido ao circuito, o projeto permite que a execução do aplicativo siga sem erros ou alterações.

5.1 TRABALHOS FUTUROS

Os imprevistos enfrentados durante o desenvolvimento (como a transição de módulos e o transporte dos dados) abrem espaço para abordar itens que são vistos como pontos de melhoria.

O propósito inicial deste trabalho era utilizar o sensor muscular Myoware para leitura de sinais de EMG e posteriormente, sua representação gráfica no aplicativo. Embora a aplicação possua capacidade de processar dados de qualquer sensor, seria interessante utilizar, de fato, o sensor proposto.

A implementação do aplicativo deste trabalho contou com uma biblioteca Bluetooth que não possui atualizações pelo seu desenvolvedor. Sendo assim, a abordagem de diferentes pacotes capazes de trabalhar com comunicação sem fio é um trabalho relevante a ser considerado, uma vez que este fator pode mitigar os gargalos presentes no transporte de dados.

Seguindo as funcionalidades não obrigatórias que foram propostas a serem implementadas, restou, portanto, possibilitar com que as leituras sejam feitas em segundo plano, isto é, não seja necessário que o aplicativo esteja na janela principal para ser executado.

Outra sugestão é a utilização de uma biblioteca do Javascript que represente graficamente os dados de forma mais interativa. O pacote D3 (D3, 2019) se mostrou efetivo em casos de aplicativos que precisam lidar com gráficos estáticos. Entretanto, a implementação de dados em tempo real utilizando React Native se faz da mesma forma se comparado com dados já existentes.

Desenvolvendo soluções para os gargalos acima, o próximo passo seria a implementação de um serviço web capaz de apresentar os mesmos dados lidos. Desta forma, seria possível

analisar as leituras já realizadas, bem como aplicar modelos de predição para estudos de comportamento de acordo com os sinais obtidos.

Referências

ADLER, C. S.; ADLER, S. M. Biofeedback-psychotherapy for the treatment of headaches: A 5-year follow-up. **Headache: The Journal of Head and Face Pain**, Wiley Online Library, v. 16, n. 4, p. 189–191, 1976. Citado na página 1.

AHMED, S. F. et al. Mobility assistance robot for disabled persons using electromyography (emg) sensor. In: **2018 IEEE International Conference on Innovative Research and Development (ICIRD)**. Bangkok, Thailand: IEE, 2018. p. 1–5. Citado na página 7.

ALIFERI, C. **Android Programming Cookbook, Chryssa Aliferi**. [S.l.]: Java Code Geeks, 2016. (Android Programming Cookbook). Citado na página 7.

ALTIMARI, L. R. et al. Influence of different strategies of treatment muscle contraction and relaxation phases on emg signal processing and analysis during cyclic exercise. **Computational intelligence in electromyography analysis—a perspective on current applications and future challenges**, p. 97–116, 2012. Citado na página 6.

ALTINI, M.; PENDERS, J.; ROEBBERS, H. An android-based body area network gateway for mobile health applications. In: **Wireless Health 2010**. Eindhoven: Conference: Proceedings of Wireless Health, 2010. p. 188–189. Citado na página 7.

ANALOG DEVICES. **Single-Lead, Heart Rate Monitor Front End**. [S.l.], 2012. Disponível em: <<https://www.analog.com/media/en/technical-documentation/data-sheets/AD8232.pdf>>. Acesso em: 07 jun. 2020. Citado na página 11.

APARICIO, M. V. Electromiografía cinesiológica. **Rehabilitación**, Elsevier, v. 39, n. 6, p. 255–264, 2005. Citado na página 5.

ARDUINO. **Getting Started with Arduino and Genuino UNO**. 2017. Disponível em: <<https://www.arduino.cc/en/Guide/ArduinoUno>>. Acesso em: 20 set. 2019. Citado na página 14.

ARDUINO. **Arduino Uno Rev3**. 2019. Disponível em: <<https://store.arduino.cc/usa/arduino-uno-rev3>>. Acesso em: 20 set. 2019. Citado na página 13.

ARDUINO. **Download the Arduino IDE**. 2019. Disponível em: <<https://www.arduino.cc/en/main/software>>. Acesso em: 20 set. 2019. Citado na página 13.

BAR-ELI, M.; BLUMENSTEIN, B. Performance enhancement in swimming: The effect of mental training with biofeedback. **Journal of Science and Medicine in Sport**, Elsevier, v. 7, n. 4, p. 454–464, 2004. Citado na página 1.

BASMAJIAN, J. V. Muscles alive. their functions revealed by electromyography. **Academic Medicine**, LWW, v. 37, n. 8, p. 802, 1962. Citado na página 1.

BASMAJIAN, J. V. Control and training of individual motor units. **Science**, American Association for the Advancement of Science, v. 141, n. 3579, p. 440–441, 1963. Citado na página 3.

BECK, R. Muscle fiber conduction velocity. **Wiley Encyclopedia of Biomedical Engineering**, Wiley Online Library, 2006. Citado na página 5.

- BUCHTHAL, F.; SCHMALBRUCH, H. Motor unit of mammalian muscle. **Physiological Reviews**, v. 60, n. 1, p. 90–142, 1980. Citado na página 3.
- BUDZYNSKI, T.; STOYVA, J.; ADLER, C. Feedback-induced muscle relaxation: Application to tension headache. **Journal of Behavior Therapy and Experimental Psychiatry**, Elsevier, v. 1, n. 3, p. 205–211, 1970. Citado na página 3.
- CORNEL, E. B. et al. The effect of biofeedback physical therapy in men with chronic pelvic pain syndrome type iii. **European urology**, Elsevier, v. 47, n. 5, p. 607–611, 2005. Citado na página 1.
- CRAM, J. R. The history of surface electromyography. **Applied psychophysiology and biofeedback**, Springer, v. 28, n. 2, p. 81–91, 2003. Citado na página 3.
- CROCE, R. V. The effects of emg biofeedback on strength acquisition. **Biofeedback and Self-regulation**, Springer, v. 11, n. 4, p. 299–310, 1986. Citado na página 1.
- D3. **D3.js - Data-Driven Documents**. 2019. Disponível em: <<https://d3js.org/>>. Acesso em: 26 mai. 2020. Citado na página 25.
- DIMARZIO, J. **Beginning Android Programming with Android Studio**. Indianapolis, Estados Unidos da América: John Wiley & Sons, 2016. Citado na página 7.
- ENOKA, R. M. **Bases neuromecânicas da cinesilogia**. São Paulo, Brasil: Editora Manole, 2000. Citado na página 5.
- FLOYD, W.; SILVER, P. The function of the erectores spinae muscles in certain movements and postures in man. **The Journal of physiology**, Wiley Online Library, v. 129, n. 1, p. 184–203, 1955. Citado na página 3.
- GIGGINS, O. M.; PERSSON, U. M.; CAULFIELD, B. Biofeedback in rehabilitation. **Journal of neuroengineering and rehabilitation**, Springer, v. 10, n. 1, p. 60, 2013. Citado na página 1.
- GITHUB. **Android-BluetoothSPPLibrary**. 2014. Disponível em: <<https://github.com/akexorcist/BluetoothSPPLibrary>>. Acesso em: 26 mai. 2020. Citado na página 17.
- GOOGLE LLC. **Arquitetura da plataforma**. 2019. Disponível em: <<https://developer.android.com/guide/platform?hl=pt-br>>. Acesso em: 21 nov. 2019. Citado 2 vezes nas páginas 8 e 9.
- GOOGLE LLC. **Configure your build**. 2019. Disponível em: <<https://developer.android.com/studio/build>>. Acesso em: 20 set. 2019. Citado na página 14.
- GOOGLE LLC. **Download Android Studio**. 2019. Disponível em: <<https://developer.android.com/studio>>. Acesso em: 20 set. 2019. Citado 2 vezes nas páginas 14 e 15.
- GREEN, E. E. et al. Feedback technique for deep relaxation. **Psychophysiology**, Wiley Online Library, v. 6, n. 3, p. 371–377, 1969. Citado na página 3.
- GUANGZHOU HC INFORMATION TECHNOLOGY. **Bluetooth Module**. [S.I.], 2006. Disponível em: <<https://www.olimex.com/Products/Components/RF/BLUETOOTH-SERIAL-HC-06/resources/hc06.pdf>>. Acesso em: 21 set. 2019. Citado na página 13.

- INMAN, V. T.; ABBOTT, L. C. et al. Observations of the function of the shoulder joint. **Clinical Orthopaedics and Related Research**®, LWW, v. 330, p. 3–12, 1996. Citado na página 3.
- KAMEN, G.; GABRIEL, D. **Essentials of Electromyography**. Ontario, Canadá: Human Kinetics, 2010. ISBN 9780736067126. Citado 3 vezes nas páginas 3, 4 e 5.
- KARIMPOUR, M. et al. An android application for estimating muscle onset latency using surface emg signal. **Journal of Biomedical Physics & Engineering**, Shiraz University of Medical Sciences, v. 9, n. 2, p. 243, 2019. Citado na página 1.
- MARCHETTI, P. H.; DUARTE, M. **Instrumentação em eletromiografia**. São Paulo, Brasil, 2006. Citado na página 5.
- MCKEE, M. Biofeedback: an overview in the context of heart-brain medicine. **Cleveland Clinic Journal of Medicine**, v. 75, p. S31, 2008. Citado na página 1.
- MCROBERTS, M. **Arduino Básico-2ª edição: Tudo sobre o popular microcontrolador Arduino**. São Paulo, Brasil: Novatec Editora, 2015. Citado na página 13.
- NPM. **React Native Bluetooth Serial Next**. 2019. Disponível em: <<https://www.npmjs.com/package/react-native-bluetooth-serial-next>>. Acesso em: 26 mai. 2020. Citado na página 18.
- NPM. **React Native - A framework for building native apps using React**. 2020. Disponível em: <<https://reactnative.dev/>>. Acesso em: 26 mai. 2020. Citado na página 18.
- OAKS, S.; WONG, H. **Java Threads: Understanding and Mastering Concurrent Programming**. [S.l.]: "O'Reilly Media, Inc.", 2004. Citado na página 17.
- OLIVEIRA, D. C. S. d. et al. Análise eletromiográfica de músculos do membro inferior em exercícios proprioceptivos realizados com olhos abertos e fechados. **Revista Brasileira de Medicina do Esporte**, SciELO Brasil, v. 18, n. 4, p. 261–266, 2012. Citado na página 3.
- PATEL, M. S.; ASCH, D. A.; VOLPP, K. G. Wearable devices as facilitators, not drivers, of health behavior change. **Journal of the American Medical Association**, American Medical Association, v. 313, n. 5, p. 459–460, 2015. Citado na página 7.
- POUNDS, N. **A study into Vibro-acoustic stimulation methods to reduce heart-rate measured using an electrocardiogram to support research in relieving symptoms of stress**. Tese (Doutorado) — London South Bank University, 05 2018. Citado na página 12.
- RAKESTRAW, T. L.; EUNNI, R. V.; KASUGANTI, R. R. The mobile apps industry: A case study. **Journal of Business Cases and Applications**, Academic and Business Research Institute (AABRI), v. 9, p. 1, 2013. Citado na página 7.
- REALM. **Realm Database**. 2014. Disponível em: <<https://realm.io/products/realm-database>>. Acesso em: 26 mai. 2020. Citado na página 20.
- REAZ, M. B. I.; HUSSAIN, M.; MOHD-YASIN, F. Techniques of emg signal analysis: detection, processing, classification and applications. **Biological procedures online**, v. 8, n. 1, p. 11, 2006. Citado na página 1.

- REEVES, J. L.; MEALIEA, W. L. Biofeedback-assisted cue-controlled relaxation for the treatment of flight phobias. **Journal of Behavior Therapy and Experimental Psychiatry**, Elsevier, v. 6, n. 2, p. 105–109, 1975. Citado na página 1.
- RICHER, R. et al. Real-time ecg and emg analysis for biking using android-based mobile devices. In: IEEE. **2014 11th International Conference on Wearable and Implantable Body Sensor Networks**. Erlangen, Germany, 2014. p. 104–108. Citado na página 6.
- SARAVANAN, S. Remote patient monitoring in telemedicine using computer communication network through bluetooth, wi-fi, internet android mobile. **International Journal of Advanced Research in Computer and Communication Engineering**, v. 3, n. 7, 2014. Citado na página 7.
- TECNOBLOG. **O que há de novo no Android 9 Pie**. 2018. Disponível em: <<https://tecnoblog.net/254575/android-9-pie-novidades/>>. Acesso em: 20 out. 2019. Citado na página 8.
- THE VERGE. **There are now 2.5 billion active Android devices**. 2019. Disponível em: <<https://www.theverge.com/2019/5/7/18528297/google-io-2019-android-devices-play-store-total-number-statistic-keynote>>. Acesso em: 15 out. 2019. Citado na página 7.
- TSAI, Y.-R.; KO, J.-H. Implementation of a portable multi-channel emg signal detection system for android-based smartphones by using usb-otg interface. In: **2018 IEEE International Conference on Applied System Invention (ICASI)**. Taiwan, China: IEEE, 2018. p. 766–769. Citado na página 1.
- WIESELMANN-PENKNER, K. et al. A comparison of the muscular relaxation effect of tens and emg-biofeedback in patients with bruxism. **Journal of oral rehabilitation**, Wiley Online Library, v. 28, n. 9, p. 849–853, 2001. Citado na página 1.
- WILMORE, J.; COSTILL, D. L.; KENNEY, W. **Fisiologia do exercício e do esporte**. São Paulo, Brasil: Editora Manole, 2001. Citado na página 4.
- WOLF, S.; BASMAJIAN, J. Assessment of paraspinal electromyographic activity in normal subjects and in chronic back pain patients using a muscle biofeedback device. **International series on biomechanics VIB**, p. 319–324, 1978. Citado na página 3.
- WOLF, S. L.; NACHT, M.; KELLY, J. L. Emg feedback training during dynamic movement for low back pain patients. **Behavior Therapy**, Elsevier, v. 13, n. 4, p. 395–406, 1982. Citado na página 3.
- WOODFORD, H. J.; PRICE, C. I. Emg biofeedback for the recovery of motor function after stroke. **Cochrane Database of Systematic Reviews**, John Wiley & Sons, Ltd, n. 2, 2007. Citado na página 1.
- YASSIN, M. et al. Rehabilitation biofeedback using emg signal based on android platform. In: **2017 IEEE 30th International Symposium on Computer-Based Medical Systems (CBMS)**. Minya, Egito: IEEE, 2017. p. 475–480. Citado na página 7.

Apêndices

APÊNDICE A – Código Arduino para recepção de sinais

```

#include <SoftwareSerial.h>

SoftwareSerial bluetooth(2, 3); // RX, TX do Arduino
String command = "";
long f;
long t;
char freq[16];
char tempo[16];
const int heartpin=A2;
float sensorval = 0.0;

void setup()
{
  Serial.begin(9600);
  pinMode(heartpin, INPUT);
  pinMode(10, INPUT);
  pinMode(11, INPUT);

  bluetooth.begin(9600);
}

void loop()
{
  delay(1);

  sensorval = analogRead(heartpin);
  ltoa(sensorval, freq, 10);

  Serial.print(sensorval, 10);
  Serial.write("\n");

  bluetooth.print(sensorval, 4);
  bluetooth.write("\n");
}

```