

Desenvolvimento de um Middleware para tratamento de dados coletados por sensores de eletroencefalograma pessoais

Gustavo Fão Valvassori, Sylvio André Garcia Vieira

Centro Universitário Franciscano (UNIFRA)
Santa Maria – RS – Brazil

{faogustavo,sylviovieira}@gmail.com

***Abstract.** This paper presents development of a middleware to perform the collection, handle and normalize the data received from the electroencephalogram sensor. To develop this, we used the Javascript language supported by ElectronJS for the desktop application. The sensors which will be implemented are the Neurosky Mindwave and the TGAM. The middleware was able to connect multiple sensors on a single application using the same code.*

***Resumo.** Este artigo apresenta o desenvolvimento de um middleware para realizar a coleta, tratamento e padronização de dados providos por um sensor de eletroencefalograma. Para realizar este desenvolvimento, foi utilizada a linguagem Javascript com apoio do ElectronJS para desenvolvimento da aplicação desktop. Os sensores que tiveram seus módulos implementados foram o Neurosky Mindwave e um módulo TGAM. O middleware conseguiu integrar ambos sensores em uma aplicação para validação utilizando o mesmo código.*

1. Introdução

A complexidade e a variedade de comportamentos motores em seres humanos, é grandemente aumentada pela capacidade notável do sistema nervoso central para controlar movimentos. Interfaces cérebro máquinas (ICM – do inglês *brain-machine interfaces*) são sistemas híbridos em que se conectam sensores diretamente à pele do indivíduo, ou ainda, tecido cerebral diretamente a máquinas, para restaurar as funções motoras e sensoriais paralisadas [IFFT et al. 2016].

O avanço da ICM foi impulsionado por dois pontos fundamentais: investigação dos princípios fisiológicos que orientam o funcionamento de grandes conjuntos neurais e o desenvolvimento de dispositivos neuroprotéticos que poderiam restaurar os movimentos dos membros, ou ainda compreender os impulsos cerebrais para serem reproduzidos por computadores [IFFT et al. 2016]. Segundo Heidrich et al. (2016) a inclusão dessas interfaces junto a jogos pode proporcionar novas oportunidades no ensino de crianças portadoras de necessidades especiais.

A introdução de jogos para o ensino-aprendizagem com a utilização de interfaces cérebro máquina poderão auxiliar consideravelmente alunos com necessidades especiais motoras. A tecnologia pode permitir a inserção de pessoas cuja capacidade de manipulação de objetos esteja comprometida, contribuindo em sua autonomia e independência. A ciência já tem condições de identificar as ondas cerebrais e traduzir a

sua representação. Assim como exames podem ser feitos para avaliar a atividade cerebral, conhecidos por EEG (Eletroencefalograma), também é possível captar essas ondas e codificá-las, para que um computador interprete-as como comandos que venham a controlar objetos que sejam exibidos em sua tela.

Contudo, atualmente existem diversos fabricantes de sensores que realizam a leitura de ondas cerebrais. Porém, cada um deles utiliza técnicas de captura e kits de desenvolvimento diferentes (SDKs – *Software development kit*). Esse fato impede que o usuário de um dispositivo consiga utilizar a mesma aplicação com um equipamento diferente.

O objetivo deste trabalho foi desenvolver um middleware que realiza a coleta e formatação dos dados gerados pela plataforma do fabricante. Esses dados são coletados e formatados para uma estrutura de dados padrão, independente da escolha do sensor.

A partir do objetivo geral, pode-se destacar os seguintes objetivos específicos:

- Estudar o funcionamento do cérebro no que tange à emissão de ondas;
- Estudar aparelhos de eletroencefalograma pessoais;
- Estudar as plataformas de desenvolvimento dos fabricantes de sensores para EEG;
- Projetar o funcionamento e desenvolver o *middleware* que permita o recebimento de comandos de múltiplas plataformas e converta em uma única aplicável aos jogos.

2. Referencial Teórico

Este tópico visa apresentar alguns conceitos que foram importantes para o desenvolvimento do projeto.

2.1. Funcionamento do cérebro

Para entender que ondas são lidas pelos sensores, torna-se necessário estudar o funcionamento do cérebro humano. Como apresentado por Lelis (2014) e Gonçalves (2009), o cérebro é formado por dois tipos de células, que são os neurônios e as células da glia.

As células da glia, conforme Bear M, et al. (2007, p. 8), são responsáveis por proteger, isolar, sustentar e nutrir os neurônio. Já os neurônios, segundo Nishida (2013), têm como função emitir sinais bioelétricos, ou impulsos nervosos, que são utilizados como estímulos para diversas ações internas ou externas do corpo. Esses transmitidos têm o nome de sinapse e podem ser de duas formas, química e elétrica.

Nishida (2013) explica que as sinapses elétricas ocorrem com a ligação entre duas células. Essa ligação é chamada de *gap junction*. Dessa forma, as trocas são mais rápidas pois são enviados sinais elétricos direto de uma célula a outra. Entretanto, esse tipo de sinapse ocorre, em sua maioria, durante o desenvolvimento do sistema nervoso e, após o seu término, predominam as sinapses químicas.

Já as sinapses químicas, segundo Nishida (2013), se realizam através da liberação neurotransmissores na fenda sináptica. Os neurotransmissores são sintetizados pelo neurônio e armazenados em vesículas no terminal axônio. Para ocorrer uma sinapse química, o neurônio libera os neurotransmissores na fenda sináptica, e eles serão

absorvidos por outra célula como é apresentado na Figura 01.

Como apresentado por Gomes (2015), antes de ocorrer uma sinapse, a célula receptora, possui uma diferença de potencial interna de aproximadamente -70 milivolts. Quando um neurônio é estimulado, ou seja, ocorre uma sinapse, a membrana do receptor permite a passagem átomos carregados positivamente, que são os neurotransmissores, alterando sua diferença de potencial para aproximadamente 30mV. Os sensores de eletroencefalograma registram essas mudanças que ocorrem nas células.

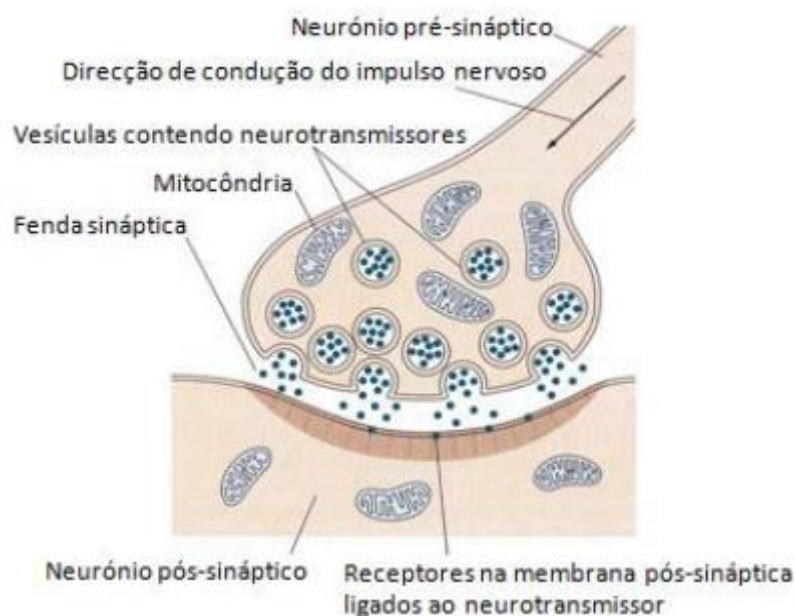


Figura 01 – Sinapse química [Gonçalves 2009, pg. 7]

2.2. Eletroencefalograma (EEG)

Segundo Zargerolame (2009), o eletroencefalograma (EEG) foi descoberto em 1929 pelo psiquiatra alemão Hans Berger. O EEG trata-se da leitura de ondas elétricas de baixa intensidade que são geradas pelo cérebro. Sua leitura é feita por meio do uso de eletrodos em determinadas regiões da cabeça (Figura 02) ligados a um amplificador.

Conforme apresentado por Gomes (2015), a leitura feita por Berger era capaz de captar dois tipos de ondas, que são a alfa (baixa frequência) e beta (alta frequência). Com a evolução das tecnologias, foi possível identificar a existência de quatro tipos de ondas distintos, cada um atuando em uma faixa de frequência específica, e a Beta, dividida em três tipos como apresentado na Tabela 01 [Zangerlone, 2009].

Tabela 01 - Ondas cerebrais e estados mentais correlatos [Zangerolame, 2009]

Onda	Frequência	Estado associado
Delta	0-4Hz	Sono, relaxamento profundo, cura
Theta	4-8Hz	Sono dos sonhos, criatividade, intuição
Alfa	8-12Hz	Relaxamento, calma, pensamento abstrato, estimulação do sistema imune

Beta Baixa	12-15Hz	Visualização, aumento da memória, movimento do corpo
Beta média	15-20Hz	Vigília, alerta, alta atividade mental, concentração
Beta alta	Acima de 20Hz	Agitação, ansiedade

Como procedimento de aquisição de sinais cerebrais, colocam-se eletrodos em posições pré-definidas sobre o couro cabeludo do paciente (ex: Fp1 e FP2, Figura 02), um amplificador aumenta a intensidade dos potenciais elétricos que posteriormente serão plotados num gráfico analógico ou digital, dependendo do equipamento [Zangerolame 2009, p. 17]

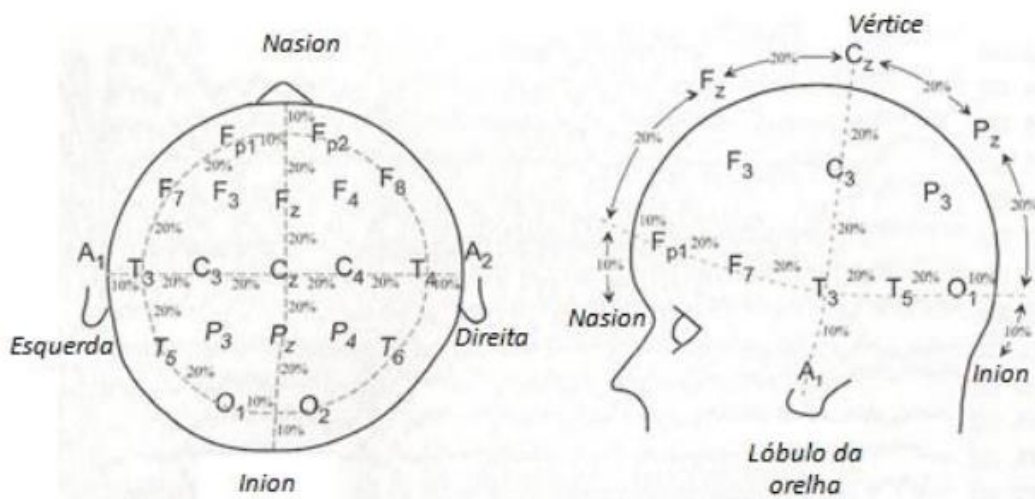


Figura 02 – Sistema internacional 10/20 para registro do EEG clínico [Gonçalves, 2009, pg. 12]

Porém, um aparelho que possibilite a aquisição dos dados possui um custo elevado, podendo passar dos R\$ 10.000,00. Com a evolução das tecnologias, foi possível desenvolver sensores pessoais com um custo mais baixo e que para uso pessoal satisfazem as necessidades dos usuários. Dentre os existentes no mercado, três deles se destacam, sendo eles o Mindwave da Neurosky e os sensores da Emotiv.

A Neurosky traz a proposta de um sensor de baixo custo buscando atingir a maior fatia do mercado [Neurosky]. Seu sensor possui apenas um eletrodo posicionado em uma haste ajustável que fica posicionada na testa, mais especificamente no ponto Fp1.

A Emotiv tem como objetivo trazer sensores com mais pontos para que possam ser detectadas mais informações. Seus sensores podem possuir 5 ou 16 eletrodos. Essa maior quantidade de sensores possibilita a aquisição de mais informações, como por exemplo o processamento auditivo e motor.

2.3. Controladores

Para que se possa controlar um personagem em um jogo, uma televisão, um computador

ou até um celular, é necessário um controlador. Esses controladores podem ser de diversas formas, e os mais comuns o teclado e mouse para computadores e joysticks para vídeo games e, celulares e controles para televisão.

Atualmente, pessoas com alguma necessidade especial podem ter dificuldade com os controladores existentes por necessitarem de uma ação física. Por outro lado, existem alguns que não são utilizados de forma física. Dentre eles é possível destacar o Move e o Kinect, que utilizam a captação de dados por imagem, as assistentes pessoais como a Siri, a Cortana e o Google Now, que realizam ações através de comandos de voz. Outro tipo são os aparelhos que realizam a leitura das ondas cerebrais, como os *headsets* da Emotiv, da Neurosky e da Muse.

3. Trabalhos correlatos

Este tópico apresentará alguns trabalhos semelhantes ao que será desenvolvido.

3.1. Análise de ondas eletroencefalográficas aplicada a Tecnologias Assistivas de Controle de ambiente

O trabalho desenvolvido por Ferreira e Roque (2014) estuda se é viável a utilização de sensores de eletroencefalograma para controle de ambientes através de equipamentos que realizam leitura de ondas cerebrais. Foi buscado trazer mais comodidade a vida de pessoas que possuem alguma necessidade especial.

Para o desenvolvimento do protótipo foi utilizado o Neurosky Mindwave para leitura e uma placa Arduino para interpretação dos sinais e execução dos comandos. Para realizar os comandos, era necessário que o usuário do sensor estivesse próximo a algum aparelho e ativasse algum estado mental, dessa forma o gatilho seria enviado ao equipamento. O gatilho definido foi um nível alto de concentração e disparado via sinal infravermelho como exibido na Figura 03.

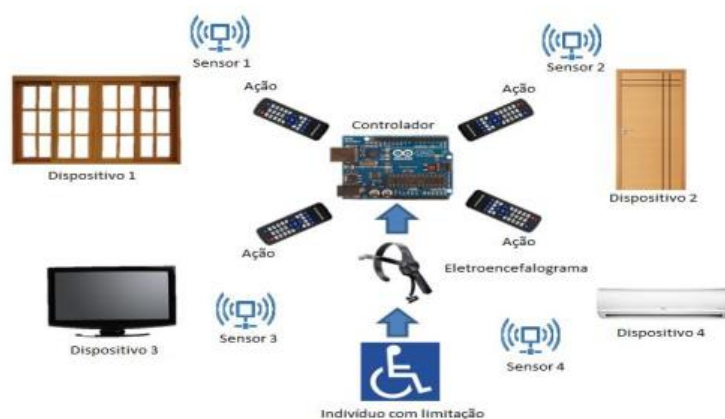


Figura 03 – Arquitetura de controle (Ferreira e Roque, 2014)

Com os resultados obtidos, foi possível compreender que as possibilidades são válidas não apenas para a tecnologia assistiva aplicada ao controle residencial, mas também para outras áreas do conhecimento.

3.2. Utilização de ondas cerebrais para controle de componentes eletrônicos

Este trabalho foi desenvolvido por Lelis (2014) como dissertação de mestrado. Ele tinha como objetivo desenvolver um sensor que realizasse a captação dos dados para controlar componentes eletrônicos. Esse sensor foi feito utilizando apenas o Chip da Neurosky retirado de um Mindflex. Também foi utilizado um Arduino para realizar a interpretação dos dados

Por utilizar o Chip da *Neurosky*, sua leitura é feita em apenas dois pontos, e o primeiro é o Fp1, exibido na Figura 04 e um grampo de orelha para remoção das interferências. Para realizar a validação dos dados lidos pelo sensor, foram desenvolvidas duas aplicações. A primeira realiza a apresentação gráfica dos dados coletados (Figura 04) e a segunda é o controle de um servo motor (Figura 05). Para desenvolver o sensor, foi definido o seguinte fluxo:

- Captação do sinal através dos eletrodos;
- Processamento desse sinal com Arduino para filtrar interferências e amplificar o sinal;
- Converter sinais em pulsos elétricos para controlar equipamentos.

O projeto trouxe uma alternativa aos sensores existentes, podendo construir um em casa. Porém, é necessário que o usuário

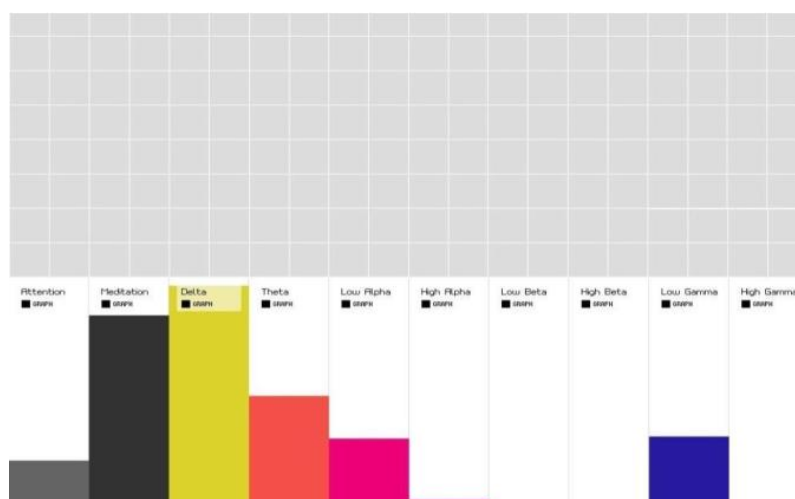


Figura 04– Gráficos gerados com a leitura do sensor

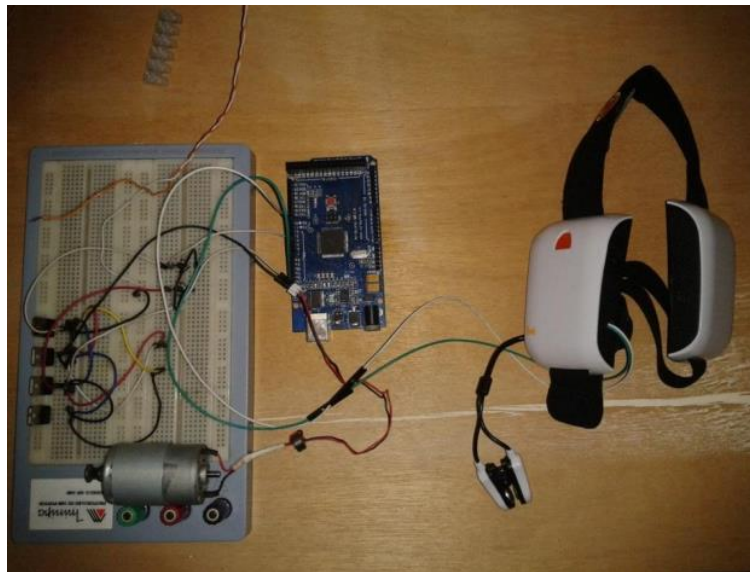


Figura 05 – Controle do servo motor

3.3. Desenvolvimento de um simulador controlado por Interface Cérebro-Computador não invasiva para treinamento na utilização de cadeiras de rodas.

Este trabalho foi desenvolvido por Schuch (2013) e teve como objetivo desenvolver um simulador para cadeirantes aprenderem utilizar cadeiras de rodas com a utilização de um sensor de EEG. Para isso, foi utilizada uma metodologia ICC (Interface Cérebro-Computador) que é bastante semelhante a IHC (Interface Homem-Computador) e é comparada a ações tomadas de forma motora pela Figura 06.

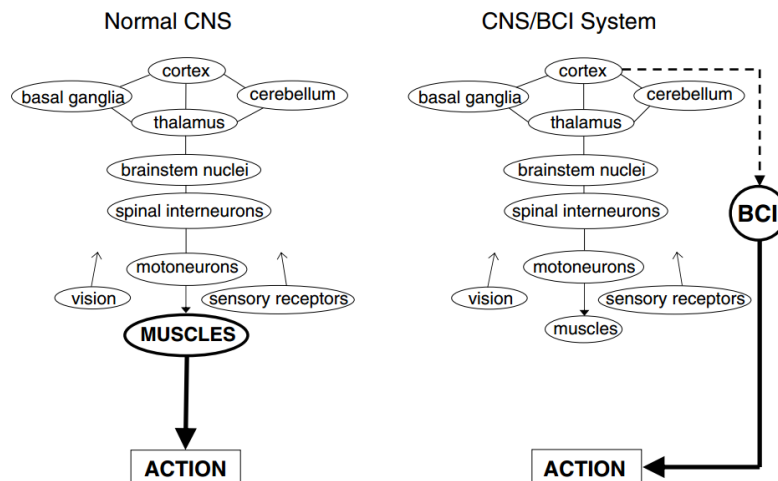


Figura 06 – Comparação entre comunicação normal do sistema nervoso central, e com a utilização de uma interface cérebro computador [Wolpaw, 2007 p. 4]

Conforme apresentado pelo trabalho, a aquisição dos dados pode ser feita de forma invasiva, com implantes no crânio, ou não invasivas, com o auxílio de eletrodos posicionados no escalpo do cérebro. Para realizar o desenvolvimento, foi utilizado o MindWave por se tratar de uma tecnologia não invasiva de baixo custo.

A comunicação entre o simulador foi feita utilizando o ThinkGear Connector, que é uma aplicação da NeuroSky para realizar a leitura e tratamento das informações. Para o desenvolvimento do simulador foi usada a GameEngine Unity, pois a empresa disponibiliza as bibliotecas necessárias para linguagem C#. Também foram desenvolvidos modelos 3D para personagens e ambientes. A interface deste trabalho pode ser vista na Figura 7.

Sua lógica de implementação foi usar o piscar dos olhos como gatilho para as ações. Foi verificado que o leitor conseguia identificar a intensidade do piscar dos olhos, retornando um valor inteiro entre 1 e 255. Assim, o controlador possui três tipos de ações (rotacionar a direita, ir em frente, rotacionar a esquerda) que são apresentadas em um display e são alternadas entre elas a cada dois segundos. Ao piscar os olhos, a ação é disparada e ao piscar novamente a ação é cancelada e o controlador volta a alternar entre as opções. Também foi adicionado um sensor de colisões na cadeira para ajudar na utilização do assento.

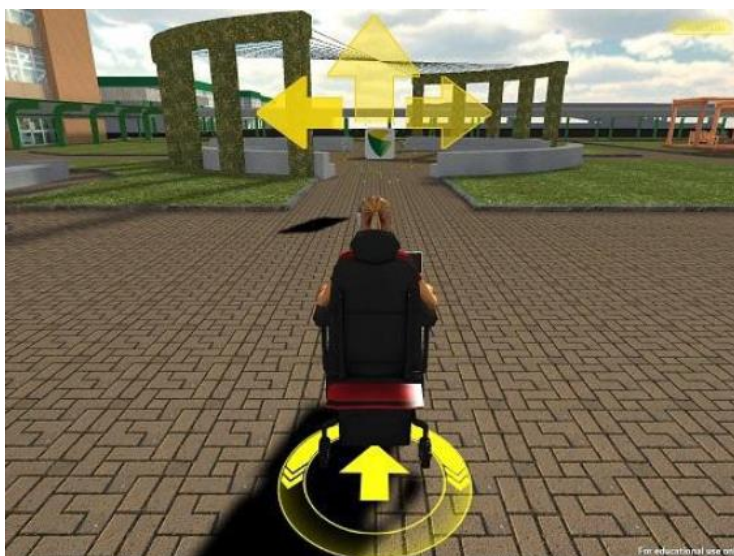


Figura 07 – Ambiente criado para simulação. [Schuch 2013, p. 6]

4. Metodologia

Para o desenvolvimento do projeto, foram utilizados conceitos da metodologia Features Driven Development (FDD – Desenvolvimento guiado por funcionalidades). Segundo Goyal (2007), essa é considerada uma metodologia ágil, sendo assim mais adaptativa às mudanças que podem aparecer no decorrer do desenvolvimento.

Para desenvolver o middleware, primeiramente foi desenvolvido uma visão geral do software (Fase 1 do FDD), representado através de um diagrama de domínio. O diagrama apresentava que os plug-ins teria um conjunto de configurações, utilizaria uma classe de comunicação para realizar o envio das informações, e gravar dados para simulação. Contudo, ele precisou ser adaptado durante o desenvolvimento para se adaptar melhor ao problema.

Logo depois foi realizada uma listagem com uma breve descrição das

funcionalidades que seriam desenvolvidas (Fase 2 do FDD). Além disso, foi desenvolvido um diagrama de casos de uso para realizar sua representação. As funcionalidades mais complexas foram detalhadas (Fase 3 do FDD) em uma tabela contendo dados como: qual caso de uso ele representava, fluxos que teria, algumas regras de negócio e caso fosse necessário alguns protótipos de interface.

A fase de Desenho por funcionalidade (Fase 4) foi removida, pois seus artefatos não trariam grandes benefícios durante o desenvolvimento e ainda diminuiria o tempo disponível para desenvolver o middleware. Para dar início a fase de construção por etapas (Fase 5), optou-se por ter todas as telas de interface com o usuário (UI – User Interface) previamente implementadas. Ao desenvolver o middleware dessa forma, foi possível ter uma visão mais abrangente de onde pretendia-se chegar, fazendo com que problemas que viriam a acontecer durante a construção fossem previamente visualizadas, estudadas e resolvidas.

4.1. Requisitos não funcionais

Como requisitos não funcionais, esse projeto teve os seguintes:

1. Portabilidade (Entre sistemas operacionais):
 - a. Utilizar a linguagem Javascript com o apoio do NodeJS para backend;
 - b. Utilizar a plataforma ElectronJS para gerar o aplicativo executável;
2. Portabilidade (Entre aplicações nativas e Web):
 - a. Utilizar os protocolos de comunicação TCP e UDP para aplicações nativas;
 - b. Utilizar o framework Socket.IO para utilizar sockets em páginas WEB.
3. Utilizar o framework ReactJS com apoio Redux para desenvolvimento de interfaces componentizadas e reativas a mudanças no estado da aplicação;
4. Utilizar formato de objetos JSON para enviar os dados ao destinatário final.

4.2. Desenvolvimento de interfaces de usuário

Neste trabalho, foram implementadas as interfaces do usuário que são adaptativas a cada sensor utilizado como pode ser visto na Figura 08. Foram utilizados conceitos e estilos do *material design* para deixar o uso mais fácil aos usuários.

Na Figura 08, é possível ver as três principais interfaces. A posição “A” apresenta a tela inicial. Nela o usuário configura os parâmetros que são necessários para os plug-ins de entrada e saída de dados operar. É nessa tela também que o usuário inicia e para os serviços e a reprodução e gravação de dados. Na posição “B” e “C” é onde o usuário configura os métodos de entrada e saída de dados respectivamente. Nela é possível ativar apenas um plug-in por vez.

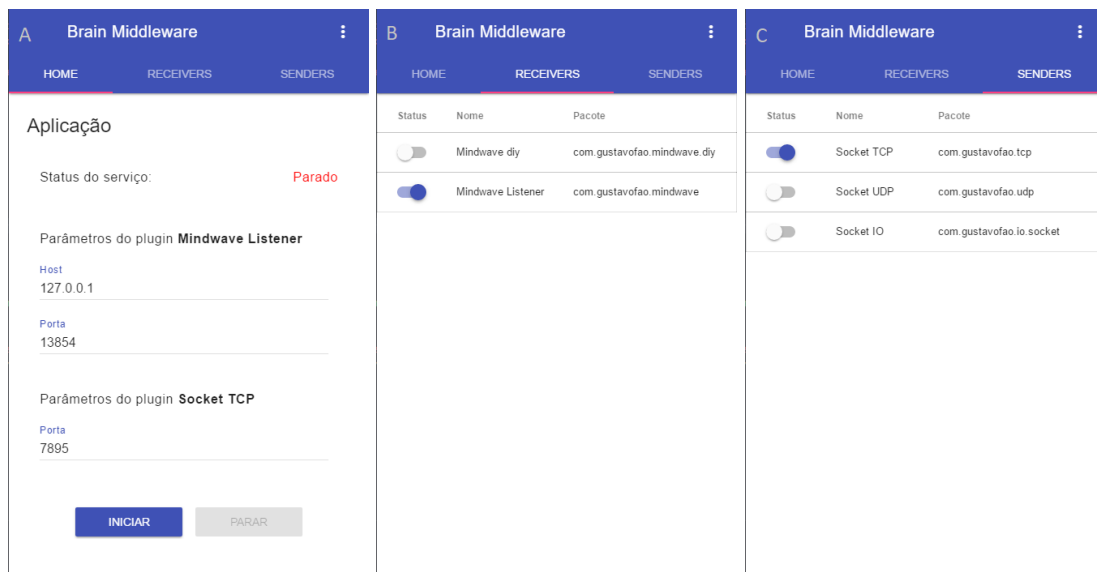


Figura 08 – Interfaces da aplicação.

4.3. Construção e utilização dos sensores

Em paralelo ao desenvolvimento das interfaces, foi iniciada a construção do sensor TGAM. O módulo TGAM é um chip desenvolvido pela Neurosky. Ele é utilizado para processamento de dados de eletroencefalografia. O chip também realiza o tratamento de dados de eletrocardiografia para anular eventuais ruídos que ocorrem nas leituras devido a circulação do sangue.

Para que seja possível utilizá-lo, é necessário: 2 clips para orelha, que serão utilizados para ignorar ruídos e 1 eletrodo, que realizará a leitura das ondas cerebrais, entrada de energia de 3.3V, que no nosso caso foi utilizado um case para duas baterias, e um chip bluetooth para comunicação com o computador. Ele montado pode ser visto na Figura 09.



Figura 09 – Módulo TGAM montado

Contudo, o sensor não funcionou conforme deveria. Era possível conectar nele

porém os dados não eram enviados ao computador. Por esse motivo, foi utilizado o sensor Neurosky Mindwave realizar a leitura de ondas cerebrais e desenvolvida uma pequena aplicação que formatava os dados no formato csv (dados separados por vírgula), utilizado pelo módulo TGAM. Logo após, esses dados foram adicionados em um array e compilado para um arduíno que foi utilizado para simular o funcionamento do módulo TGAM. Foi escolhido um arduíno por utilizar uma porta serial e o processo de conexão ser igual ao dispositivo bluetooth.

Já o Neurosky Mindwave não teve nenhum problema e sua implementação é bastante simples. O equipamento utiliza um programa proprietário chamando Thinkgear Connector que tem como principal função realizar a interpretação dos dados coletados pelo sensor. Ele possui uma porta TCP aberta para saída das informações lidas que é utilizada para o desenvolvimento de aplicações.

4.4. Construção por funcionalidades

O primeiro passo para construção do *Middleware* foi a implementação de regras de interface. Foram elas: a possibilidade de utilizar apenas uma forma de entrada e de saída de dados por vez e a ativação e desativação de modo desenvolvedor.

Com isso concluído, iniciou-se a integração com os plug-ins de entrada. Primeiramente foi adicionado o Neurosky Mindwave por ter uma integração mais simples. Para utilizá-lo foi apenas necessário conectar no host e porta especificada pelo fabricante e enviar um pacote de configurações que irá realizar a formação dos dados conforme é apresentado na Figura 10.

```
start({ host, port }) {
  return new Promise((resolve, reject) => {
    this.connector.connect(port, host, (err) => {
      if (err) {
        reject(err);
      } else {
        this._configure(resolve, reject);
      }
    });
  });
}

_configure(resolve, reject) {
  this.connector.write(JSON.stringify({
    enableRawOutput: false,
    format: 'Json',
  })), () => {
    this._setListeners();
    resolve();
  });
}
```

Figura 10 – Configurações para uso do Neurosky Mindwave

Logo após, foi iniciada a integração do TGAM. Por usar uma porta serial, sua configuração foi um pouco diferente. Além disso, foi necessário realizar uma recompilação do Electron devido a utilização de interfaces nativa que não eram providas por ele.

```

start({ com_port }) {
  console.log('Starting at', com_port);
  return new Promise((resolve, reject) => {
    this.serial = new SerialPort(com_port, {
      baudRate: 57600,
      parser: new SerialPort.parsers.Readline('\n'),
    });

    const parser = this.serial.pipe(
      new SerialPort.parsers.Delimiter({
        delimiter: Buffer.from('\n'),
      }),
    );

    this.serial.on('open', () => {
    });
  });
}
}

```

Figura 11 – Configuração da porta serial para uso do módulo TGAM

Com as duas formas de entrada de dados para o *Middleware*, ocorreu a integração dos métodos de saída. Foram implementadas três diferentes formas, que são: Utilização de *sockets* TCP e *sockets* UDP para aplicações nativas e com a biblioteca Socket.IO que realiza a comunicação com aplicações web.

Já para normalizar os dados, foi implementada um objeto modelo. Ele possui uma função que realizava o filtro dos valores recebidos pelo sensor. As ondas aceitas serão: *delta*, *theta*, *lowAlpha*, *alpha*, *highAlpha*, *lowBeta*, *mediumBeta*, *highBeta*, *lowGamma*, *gamma* e *highGamma*. A sua formatação para saída era feita também por essa model através de uma função que gerava um objeto JSON com os seguintes atributos:

- **Status:** valor inteiro que representa um *HTTP Status Code*;
- **events:** eventos recebidos, como meditação por exemplo ;
- **EEG:** que contém os valores lidos em cada ponto, como por exemplo as leituras feitas pelos sensors implementados são do ponto FP1.

As gravações dos sinais de entrada são realizadas e salvas no formato JSON em um arquivo. Dessa forma, para realizar uma simulação com dados previamente lidos era necessário apenas iterar entre cada registro salvo do arquivo. Cada iteração continha 3 atributos contendo: tempo decorrido em milissegundos da última gravação (*delay*), nome do método que deveria ser executado (*method*) e um *array* com parâmetros que deveriam ser passados ao método. Dessa forma, sua implementação poderia ser feita conforme é mostrado na figura 12.

```

const item = this.json[this.currentIndex++ % this.json.length];
setTimeout(() => {
  if (!this.started) {
    return reject();
  }

  this.eegController[item.method](...item.args);
  resolve();
}, item.delay);

```

Figura 12 – Algoritmo para implementação dinâmica de funções durante a simulação

4.5. Testes

Para testar se o projeto funcionava conforme esperado, foi desenvolvida uma interface que possibilitava a visualização dos dados gerados pelo sensor. Para isso, utilizou-se o framework de comunicação Socket.IO, que transmitia os dados até o destino final. Os gráficos foram plotados com a utilização da biblioteca Google Charts.

Para entrada de dados foi utilizado o Neurosky da Mindwave, que fornecia seus dados através de uma conexão TCP e a outra foi utilizando o módulo TGAM, que era lido através de uma porta serial no computador.

Na figura 13 é possível visualizar a apresentação em forma de gráfico das leituras feitas. Nela é apresentado os dados do bloco “eeg” do objeto recebido do middleware. Já na Figura 14 são apresentados os valores recebidos pelo bloco de eventos.

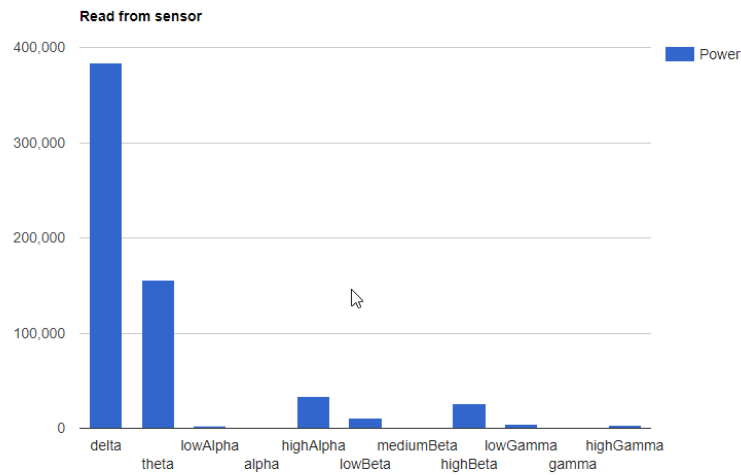


Figura 13 – Validação através de uma aplicação web.

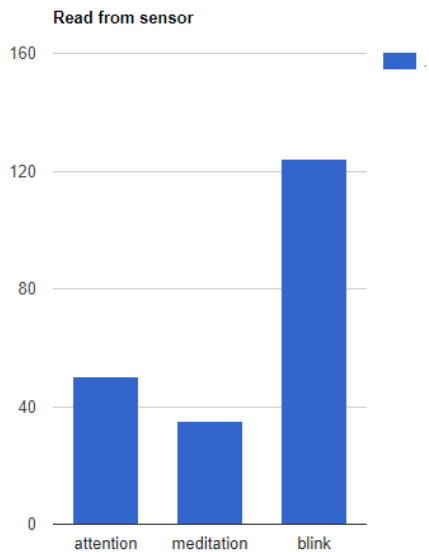


Figura 12 – Validação através de uma aplicação web.

5. Conclusão

Em grande parte das vezes que o ser humano necessita realizar uma ação, é necessário que utilize uma ação física. Contudo, pessoas com necessidades especiais podem ter algumas dificuldades. Uma das alternativas é a utilização de controladores não físicos, como os que realizam ações por voz, imagem ou pensamento.

Esse trabalho apresentou o desenvolvimento de um Middleware para apoiar o desenvolvimento de aplicações com sensores de eletroencefalograma pessoais. Foi identificado que esse tipo de equipamento era de difícil de utilização devido a falta de padrões. Cada fabricante de sensores possuía sua formatação de dados e protocolo de comunicação o que acaba tornando o software utilizável apenas com um sensor.

Com o desenvolvimento da plataforma, foi possível integrar dois diferentes métodos de entrada em uma mesma aplicação sem precisar replicar ou ajustar o software. Contudo, o fato de o módulo desenvolvido não funcionar conforme esperado trouxe alguns empecilhos ao seu desenvolvimento, principalmente por não conseguir validar com dois equipamentos reais.

Dentre as contribuições trazidas com o desenvolvimento desse middleware está: a facilidade de integração entre diversos sensores, tendo sua utilização principalmente para educação de pessoas com necessidades especiais, e a utilização da plataforma ElectronJS com a linguagem Javascript utilizando *NodeJS*, que é pouco explorada para trabalhos científicos. Além disso, foi aberta uma nova área de estudos na instituição.

Como desenvolvimento futuro, sugere-se a adição de novos sensores e a realização de melhorias na integração dos plug-ins com o middleware, visto que por rodar na plataforma *ElectronJS*, não é possível carregar módulos do *NodeJS* junto dele. Isso pode tornar-se um problema ainda maior quando utilizadas dependências que necessitem recompilar o produto.

Referencias

- Badcock, Nicholas A.; Mousikou, Petroula; Mahajan, Yatin; Lissa, Peter de; Thie, Johnson; McArthur, Genevieve. Validation of emotiv eeg gaming system for measuring research quality auditory erps. Sydney. 2013
- Bear M, et al., *Neuroscience – Exploring the Brain*, 3ª Edição, Lippincott Williams & Wilkins, Estados Unidos da América, 2007
- Campbell, Andrew T; Chodhury, Tanzeem; Hu, Shaohan; Lu, Hong; Mukerjee, Matthew K.; Rabbi, Mashfiqui, Raizada, Rajeev D. S. NeuroPhone: Brain-Mobile Phone Interface using a Wireless EEG Headset. Dartmouth College, Hanover, NH, USA. 2010
- EMOTIV, 2017. Disponível em: <<http://https://www.emotiv.com/>>. Acesso em: 20 jun. 2017.
- Falkembach, Gilse A. Morgental. O Lúdico e os jogos educacionais. Centro Interdisciplinar de Novas Tecnologias na Educação - UFRGS. Porto Alegre.
- Ferreira, Matheus Hentges; Roque, Alexandre dos Santos. Análise de ondas

- eletroencefalográficas aplicada a tecnologia assistiva de controle de ambiente. Universidade Regional Integrada do Alto Uruguai e das Missões. 2014
- Gomes, Fabio (2014) “Introdução ao FDD – *Feature Driven Development*” disponível em: <http://www.devmedia.com.br/introducao-ao-fddfeature-driven-development/27971>>. Acesso em: 27 mai. 2017.
- Gomes, Marleide de Mota “Bases fisiológicas do eletroencefalograma”, Instituto de Neurologia Deolindo Couto, Universidade Federal do Rio de Janeiro (UFRJ). 2015.
- Gonçalves, Miguel Vasco Rodrigues. Processamento de dados em aquisição simultânea de eeg/ifmr. Departamento de física, Universidade Nova de Lisboa. 2009
- Goyal, Sadhna major seminar on feature driven development. Technical University Munich. 2007
- Heidrich, Regina de O.; Brancho, Marsal A; Mossmann, João B.; Schuh, Anderson R.; Jensen, Emely. Jogos digitais para interação com *Brain Computer Interface* para auxiliar no processo de inclusão escolar de pessoas com paralisia cerebral. Congresso Brasileiro de Pesquisa e Desenvolvimento em Design. Belo Horizonte, Minas Gerais. 2016
- LELIS, Atahualpa Bastos. Utilização De Ondas Cerebrais Para Controle De Componentes Eletrônicos. Revista Científica Semana Acadêmica. Fortaleza, ano MMXIV, Nº. 000052, 31/03/2014.
- Lima, Edirlei E. Soares; De Paula, Pedro Luiz. 3D Game Builder: Uma game engine para criação de ambientes tridimensionais. Universidade do Contestado
- Muse, 2017. Disponível em: <<http://www.choosemuse.com/>>. Acesso em: 20 jun. 2017.
- Neurosky, 2017. Disponível em: <<http://neurosky.com>>. Acesso em: 20 jun. 2017.
- Nishida, Silvia M., Mecanismos de comunicação entre neurônios e dos neurônios com os órgãos efetadores. Material do Curso de Fisiologia do departamento de Fisiologia da Unesp. 2013
- Palmer, Felsing. A practical guide to feature driven development. 2002
- Salabun, Wojciech. Processing and spectral analysis of the raw EEG signal from MindWave. West Pomeranian University of Technology, Szczecin. 2014
- Schuch, Anderson R.; Lima, Alessandro Lima; Heidrich, Regina O.; Mossmann, João; Flores, Cecilia; Bez, Marta R. Desenvolvimento de um simulador controlado por interface cérebro-computador não invasiva para treinamento na utilização de cadeiras de rodas. Universidade Feevale. 2013
- Wolf, Mark J. P.. The vídeo game explosion – A history from pong to Playstation and beyond. 2008
- Wolpaw, Jonathan R. Brain-Computer interfaces as new brain output pathways. Laboratory of Nervous System Disorder, New York State Department of health and State University, Albany. 2007
- Zangelorame, Maria Fernanda. Monitoramento Analógico e Digital de Sinais Elétricos Cerebrais. Dissertação de mestrado. Centro Federal de Educação Tecnológica Celso Suckow da Fonseca. Rio de Janeiro. 2009.