



Pontifícia Universidade Católica do Rio de Janeiro

**Ambiente Virtual para Apoio à Reabilitação de Pessoas com
Dificuldades Físicas nos Membros Superiores**

Antônio Vasconcellos Chaves

PROJETO FINAL DE GRADUAÇÃO

CENTRO TÉCNICO CIENTÍFICO – CTC

DEPARTAMENTO DE INFORMÁTICA

Curso de Graduação em Engenharia da Computação.

Rio de Janeiro, julho de 2022



ANTÔNIO VASCONCELLOS CHAVES

**Ambiente Virtual para Apoio à Reabilitação de Pessoas com
Dificuldades Físicas nos Membros Superiores**

Relatório de Projeto Final, apresentado ao curso de Engenharia da Computação da PUC-Rio como requisito parcial para a obtenção do título de Engenheiro de Computação.

Orientadores: Alberto Barbosa Raposo e Greis Francy Mireya Silva-Calpa

Rio de Janeiro, julho de 2022

Agradecimentos

Aos meus orientadores, Alberto Raposo, Greis Silva, e ao professor Renato Cherullo por terem me auxiliado durante o projeto.

À PUC-Rio, pela formação e equipe de profissionais que me influenciaram diretamente na troca da minha carreira.

Aos professores Bruno Feijó, Sérgio Lifschitz, Ivan Mathias Filho, Jan K. S. Anna, pelo conteúdo de suas aulas e formas de ensino que me marcaram.

Aos meus amigos, em especial Jhonatha Neves e João Guilherme Alvarez, por terem despertado meu interesse na engenharia da computação.

Ao meu irmão, José Chaves, que esteve comigo sempre que podia.

Aos meus amigos, Felipe Leal, Daniella Espíndola, Dilan Kayne, Pedro Ivo, Pedro Raguenet e Juliana Lima principalmente nos últimos 3 anos, que se tornaram grandes amizades.

À instituição de ensino, Colégio de São Bento, onde passei a maior parte da minha vida, gostaria de agradecer a todo corpo docente por ter cuidado de mim minha infância inteira.

Aos professores Sebastião Tadeu Pereira e Fábio Bernardes, por todos os seus ensinamentos.

Ao basquete, e ao cheerleading, por ter mudado a minha vida.

Ao Marcus Santos, meu mentor na programação.

Aos meus amigos que me acompanharam na execução desse projeto, Marcos Paraizo e Clara Moreira.

À minha família, Ana Vasconcellos, Ronaldo Chaves, Carol Vasconcellos, Stephanie Chaves, Orlanzy Chaves e Ruy Chaves por todo suporte, oportunidades e conversas.

A um grupo em especial, Benson, por ter me acompanhado desde o ensino médio na minha vida.

E finalmente, à Marcílio Marinho, por todos os seus ensinamentos.

Resumo

Chaves, Antônio Vasconcellos. Raposo. Alberto Barbosa. Silva-Calpa, Greis Francy Mireya. Sleight of Hands - ambiente virtual para apoio à reabilitação de pessoas com dificuldades físicas nos membros superiores. Rio de Janeiro, 2022. Relatório de Projeto Final – Departamento de Informática. Pontifícia Universidade Católica do Rio de Janeiro.

Neste trabalho foi aprimorado um jogo cujo objetivo é criar um ambiente virtual para apoio à reabilitação de pessoas com dificuldades físicas nos membros superiores. O jogo foi desenvolvido em Unity para Oculus Rift e conta com a interação sendo feita diretamente com as mãos do usuário, para ela é utilizado o Leap Motion para detecção de movimentos e conta com a adição do Myo Armband para o rastreamento do esforço com a execução dos gestos. Nele, são utilizados exercícios de terapia tradicional como mecânicas. E foi desenvolvido buscando manter o usuário entretido, de forma que gere maior engajamento a pacientes e interesse às pessoas que não precisam de reabilitação. Foram realizados testes de experiência do usuário, a maioria, em pessoas sem qualquer tipo de deficiência nos membros superiores, nos quais os participantes declararam que gostariam de ter mais experiências com o jogo desenvolvido. Testes com pacientes em maior escala serão elaborados em trabalhos futuros.

Palavras-chave

Realidade virtual, jogos, reabilitação física.

Abstract

Chaves, Antônio Vasconcellos. Raposo. Alberto Barbosa. Silva-Calpa, Greis Francy Mireya. Sleight of Hands - virtual environment to support the rehabilitation process of people with physical difficulties in the upper limbs. Rio de Janeiro, 2022. Final Project Report – Department of Informatics. Pontifical Catholic University of Rio de Janeiro.

In this project, a game was improved in order to create a virtual environment to support the rehabilitation of people with physical difficulties in the upper limbs. The game was developed in Unity for Oculus Rift and has the interaction being done directly with the user's hands, for which Leap Motion is used to detect movements and with the addition of Myo Armband, we were able to track the effort within each gesture execution. In the game, traditional therapy exercises are used as mechanics and it was developed in a way to keep the user entertained, generating greater engagement on patients in need of treatment and interest on people that are not in need of rehabilitation. User experience tests are conducted, most of them, with people without any type of disability in the upper limbs, in which the participants declared that they would like to have more experiences with the improved game. Major tests with patients will be elaborated in future works.

Keywords

Virtual reality, games, physical rehabilitation.

Sumário

1. Introdução.....	8
2. Situação Atual.....	11
3. Objetivos do Trabalho.....	14
4. Atividades Realizadas.....	15
4.1. Definição das ferramentas e plataformas a serem utilizadas.....	15
4.2. Estudos Preliminares.....	18
4.3. Estudos conceituais e de tecnologia.....	18
4.4. Testes e protótipos para aprendizado e demonstração.....	19
4.5. Método.....	21
4.5.1. Definição da feature, seu comportamento e estrutura.....	21
4.5.2. Protótipo Inicial.....	22
4.5.3. Teste Piloto.....	23
4.5.4. Correções e Calibrações.....	23
4.5.5. Teste Final.....	24
4.6. Plano de Ação.....	24
5. Projeto e especificação do sistema.....	26
5.1. Versionamento e planejamento.....	26
5.2. Set up do projeto.....	26
5.3. Desenvolvimento.....	27
5.3.1. Unity.....	27
5.3.2. Exercícios.....	27
5.3.2.1. Punho.....	28
5.3.3. Inimigos.....	30
5.4. Sevidor.....	30
5.4.1. Estrutura e desenvolvimento.....	30
5.4.2. Estabelecer conexão.....	31
5.4.3. Utilitários ou serviços.....	31
5.4.4. Controllers.....	32
6. Implementação e Avaliação.....	32
7. Considerações Finais.....	33
8. Referências.....	34

Lista de Figuras

Figura 1: Myo Armband, por Thalmics Labs.....	15
Figura 2: Características do Myo, por Thalmic Labs.....	16
Figura 3: LEDs característicos do Myo e suas definições, por Thalmic Labs.....	16
Figura 4: Gestos reconhecidos pelo Myo, imagem por Myo Market: Diagnostics Page.....	17
Figura 5: Myo Diagnostics Page.....	17
Figura 6: Autômato que controla a detecção dos exercícios.....	27
Figura 7: Movimento do exercício do punho.....	29
Figura 8: Raio realista utilizado no jogo, obtido da Unity Asset Store.....	29

Lista de Tabelas

Tabela 1: Cronograma Projeto Final I.....	25
Tabela 2: Cronograma Projeto Final II.....	25
Tabela 3: Relação ataque e defesa elementar de cada inimigo.....	30

1. Introdução

Realidade Virtual, ferramenta idealizada por ser capaz de simular realidades que não a nossa. Com seu contínuo crescimento e potencial, tem se tornado diretriz popular e, em certos casos, essencial para áreas como construção, entretenimento, comunicação, saúde, entre outras. Nos dá a liberdade de gerarmos qualquer tipo de conteúdo, razão pela qual foi a escolha ideal para a produção desta proposta. Para seu desenvolvimento, foram utilizados o motor Unity com desenvolvimento em C#, principal motor gráfico atual compatível com a maioria dos hardwares existentes, e um servidor em NodeJS, este em Javascript, para comunicação com o Myo Armband, além do Leap Motion para a captura dos gestos na aplicação. Para seu manuseio, foram necessários estudos quanto a linguagens, reiteração de encapsulamentos aprendidos em programação modular, orientação a objetos, e reinventar métodos a fim de solucionar dificuldades encontradas para atender o projeto.

Vale pontuar que, dentro do ramo da realidade virtual, não são todos que conseguem aproveitar suas funcionalidades totais. Em casos de limitações cognitivas ou motoras, não são poucas as pessoas a não serem capazes de usufruir das totalidades deste ideal. Elas, por passarem por experiências traumáticas (fraturas, AVEs, torções, inflamações), tendem a estar em desgaste contínuo para execução de movimentos comuns ao dia a dia. Estes, dependendo do caso, passam por inúmeras sessões de fisioterapia para sua recuperação, e não só em consultórios, mas principalmente em casa. Retrato, então, a área da fisioterapia. Esta que não é atrativa para o paciente principalmente em situações como a de ter de realizar procedimentos e exercícios em casa. Seja por não ser prazeroso, por ser doloroso, monótono ou por ser subestimado quando comparado ao tratamento em consultórios. Para melhor contextualização quanto ao motivo da escolha desse projeto, devo realçar a importância em esclarecer a existência de duas vertentes para o tratamento e reabilitação de mãos, a fisioterapia e a terapia ocupacional: "A Fisioterapia se utiliza do movimento para recuperação, e a terapia ocupacional se utiliza da atividade da vida diária e prática como forma de reabilitação, por isso movimento e função se complementam" - Denise Flávio, presidente da

Associação de Fisioterapeutas do Brasil (AFB). E, em pacientes, é comum os a crença de que apenas nas sessões agendadas seriam dadas como necessárias para a devida recuperação. Contudo, este é um pensamento errôneo, os tratamentos nas clínicas são complementares para nossa mobilidade, claro que em determinados casos são estritamente necessários, por agirem como anestésicos, mas deveriam ser aproveitadas para fazer os exercícios e tratamentos que os pacientes seriam incapazes de reproduzir em casa. Sejam eles por conta do uso de equipamentos ou técnicas que requerem a presença do profissional (por exemplo: acupuntura, aparelhos de ondas curtas e ultrassom usados na eletroterapia).

Graças a traumas passados, eu, pessoalmente, fiz diversas sessões de fisioterapia para mãos, pés e joelhos. Fui ao encontro com diversos profissionais da área, então tive a melhor oportunidade para dar continuidade ao projeto implementado pelo Pedro Sousa Meireles (Meireles, P. S., Silva-Calpa, G. F. M., & Raposo, A. B. Exploring Direct User-Interaction Techniques in a Virtual Reality Game for People with Hand Impairments). E com a experiência que tive, percebi que poucos são aqueles, em seu pós-trauma, que exercem a terapia caseira, focada em tarefas diárias. Por isso, nesse projeto, focaremos em recriar esses exercícios direcionados aos membros superiores, já que são os principais responsáveis pela nossa capacidade de realizar atividades cotidianas ao longo da vida. Durante o projeto, ocorrem inúmeras conversas com profissionais da área. Através delas, ficou clara a possibilidade de refino da solução com as ênfáticas retratadas pelos profissionais quanto à forma a serem executados, pelos pacientes, os gestos implementados.

Na realidade, já existem programas que satisfazem as premissas do projeto, pelo menos quanto ao direcionar terapêutico. Inclusive, estes usam as mesmas ferramentas utilizadas neste projeto, no entanto possuem o público alvo direcionado às pessoas que precisam realizar a reabilitação de mãos. Neste projeto, entretanto, o público alvo é o comercial e o terapêutico, por assim dizer. O que não satisfaz as outras soluções como substitutas ou superiores a esse projeto, se trata de não ter potencial popularizante. Portanto, não seriam capazes de atender o público direcionado por não se tornar popular, por ser monótono. Ou seja, não agregam superioridade à terapia tradicional em casa.

Por meio desta aplicação, somos capazes de atingir toda forma de usuário. Seja um paciente que possui a necessidade de tratamento terapêutico, ou um usuário comum em busca de entretenimento, pois além de todos os

tópicos mencionados, o objetivo não é unicamente um software para tratamento, mas um jogo. E como sua própria definição nos diz: O conceito de jogo consiste numa atividade física ou intelectual formada por um conjunto de regras e define um indivíduo (ou um grupo) como vencedor e outro como perdedor.

2. Situação Atual

Nesta seção contextualiza-se o uso de realidade virtual para reabilitação física, apresentando as soluções já existentes nesse domínio e evidenciando as limitações observadas.

Desde a sua criação, em 2006 com o Nintendo Wii, a realidade virtual vem sendo amplamente estudada como alternativa para o auxílio aos métodos terapêuticos e a reabilitação pós-traumática. Esta, principalmente, direcionado a pacientes que sofreram acidente vascular encefálico (AVE). Eles, como consequência do AVE, são impactados com fraqueza muscular, perda sensitiva, espasticidade, confusão, alteração das funções cognitivas, déficit funcional, disartria, surdez unilateral e perda da visão. Para sua recuperação, segundo a Dra. Isabela Alves Marques, se fazem necessários diversos procedimentos:

1. avaliação inicial do comprometimento, para identificar as necessidades do paciente;
2. o esclarecimento de metas, ou seja, a definição de objetivos realistas e atingíveis de melhoria;
3. tipo intervenção, para auxiliar no alcance das metas;
4. reavaliação, para avaliar o progresso em relação aos objetivos acordados.

No ambiente reabilitacional virtual, foram desenvolvidos, principalmente, métodos terapêuticos direcionados a função cognitiva, confusão, espasticidade, déficit funcional e fraqueza muscular. Um exemplo Em 2010, Gustavo Saposnik foi responsável por uma equipe de pesquisa onde foi comparada a efetividade da reabilitação utilizando jogos, em pessoas com a faixa etária entre 41 e 83 anos, fazendo a separação dos pacientes em grupos: o primeiro faria terapia recreacional, ou seja, jogaria cartas, bingo e jogos similares; o segundo utilizaria um sistema de realidade virtual para o Nintendo Wii, console que tem como base de funcionamento o seu controle com detecção de movimento. Os resultados indicaram que, mesmo com menos sessões, o segundo grupo apresentou a maior taxa de recuperação.

Pesquisa feita por Flávia Gonçalves Fernandes, em 2017, foi implementado um jogo direcionado a crianças e adolescentes com deficiência

física nos membros superiores usando o Myo Armband como controlador de um jogo de quebra cabeça. Os resultados mostraram que conseguiram jogar utilizando o membro com deficiência e sentiram-se mais motivados a jogar. Visto que, por sua deficiência, normalmente não são capazes de ter uma experiência agradável em jogos comuns. O resultado foi muito positivo, pois, no primeiro momento, um número de pacientes a entender a execução dos movimentos imediatamente foi de 40%, mas todas as crianças conseguiram executar todos os movimentos.

Sobre o uso de jogos comerciais em reabilitação, foi conduzido outro estudo por Hung et al. (2016) sobre o uso de jogos em terapias, onde a opinião de pacientes e terapeutas foram ouvidas. Foram analisados 24 jogos. Os resultados mostraram que a terapia tradicional não é tão efetiva, os chamados jogos sérios, voltados à reabilitação de pacientes, não chamaram muita atenção, mas foi concordado que jogos comerciais seriam ideais por gerar entretenimento. Desse modo, os autores sugerem três recomendações para a criação de jogos voltados à reabilitação:

1. A existência de um desafio lógico a fim de manter o interesse do usuário;
2. Dar liberdade de escolha de ações pelo usuário, para evitar situações de anisedade;
3. O jogo deve ter uma interface simples e intuitiva.

Através de estudos sobre os jogos comerciais na área da reabilitação, eles não atendem os requisitos mínimos. Isso ocorre porque pacientes em necessidade de reabilitação possuem limitações que jogos comerciais não têm como público alvo. Desta forma, se faz necessário o desenvolvimento de jogos com objetivo de serem utilizados como exercícios terapêuticos.

“Um artigo desenvolvido por Karashanov, Manolova e Neshov (2016) fez o uso do reconhecimento de gestos a partir do sensor Leap Motion para criar um jogo que ajudasse na reabilitação. Gestos proporcionam uma forma intuitiva de comunicação não só entre pessoas, mas também entre humano e computador. A vantagem do uso de gestos é que não é necessária a manipulação de um controle, pois as interações são feitas diretamente com as mãos do usuário.

Nesse trabalho foi desenvolvida uma aplicação utilizando Unity3D para trabalhar movimentos que pudessem colaborar para a recuperação de pacientes com problemas nos membros superiores. Foram criadas quatro tarefas para trabalhar movimentos diferentes. A primeira consistia em usar os gestos de agarrar e soltar, fazendo o usuário coletar objetos e jogá-los em uma lata de lixo.

Na segunda o jogador deve pegar objetos pequenos e realizar movimentos precisos para encaixar formas em moldes e construir pirâmides. A terceira tarefa é a mais complexa, logo exige coordenação e movimentos precisos para guiar uma bola em um labirinto. Por fim, a última tarefa era uma forma de desenho em realidade virtual, em que o paciente deveria usar o dedo ou algo similar a uma caneta para seguir um padrão pré-desenhado.” - MEIRELES, PEDRO SOUSA, 2019.

Mesmo através de inúmeras formas de desenvolvimento, abordagens e ferramentas, poucos foram os casos encontrados de jogos desenvolvidos para reabilitação capazes de interessar o usuário da mesma forma que jogos comerciais. O que se destaca pelo ideal utilizado é o implementado por Pedro Sousa Meireles, o Sleght of Hands. Onde neste trabalho, busca-se fazer a avaliação de confiabilidade por profissionais de fisioterapia e terapia ocupacional, fazer as devidas alterações e testes em pacientes em processo de terapia reabilitacional e pessoas sem problemas cognitivos ou motores de membros superiores.

3. Objetivos do Trabalho

Este trabalho buscou dar continuidade e manutenção ao jogo implementado pelo ex-aluno Pedro Meireles, realizando refinamentos quanto aos métodos terapêuticos usados. A aplicação desenvolvida faz uso dos métodos utilizados para imergir o usuário em uma experiência cognitiva interessante, mantendo-o entretido. O jogo possui 4 técnicas terapêuticas para induzir o usuário a executá-las, mas através de opiniões profissionais da área, foi concluída a necessidade de mais um tipo de interação quando comparada com a versão original. O programa foi desenvolvido em Unity para Oculus Rift, conta com o Leap Motion para o rastreamento das mãos do usuário e com o Myo Armband, hardware adicionado na versão atual, para a captação do esforço. E para isso, a existência do Myo. Que há de corroborar com o desenvolvimento cognitivo e muscular do usuário. Com sua adição, é buscado gerar um jogo capaz de gerar resultados mais promissores do que os já implementados no mercado, de maneira a atrair público de todas as formas, com interesse terapêutico ou comercial. Isso ocorre, pois os movimentos requeridos pelo jogo não fazem parecer ter objetivo terapêutico, mas puramente recreativo.

4. Atividades Realizadas

4.1. Definição das ferramentas e plataformas a serem utilizadas

Durante a execução desse projeto, é dada continuidade ao uso do Unity, e Leap Motion, mas é acrescentado um novo acessório com objetivo de captar o esforço gerado pelo usuário ao executar um movimento. Este acessório é o Myo Armband. Através dele, temos a solução para o problema de fazer a leitura dos gestos através do movimento dos músculos, utilizando eletromiografia. Princípio de funcionamento para reconhecimento de gestos utilizados pelo Myo (GANIEV; SHIN; LEE, 2016). Em seu lançamento em 2014, seu preço de mercado era de US\$149, até sair de linha em 2018, com a compra, pela Google, da Thalmic Labs (hoje, North).



Figura 1: Myo Armband, por Thalmics Labs. Imagem retirada de <https://cosmos.bluesoft.com.br/produtos/628451040019-myo-gesture-control-armband-black>

O bracelete da Thalmic chamado Myo é um dispositivo que utiliza 8 sensores para medir sinais eletromiográficos produzidos pelos músculos ao realizar um movimento. Esses sinais são processados e enviados via Bluetooth para outros dispositivos. Além de outras funcionalidades do acessório, o Myo também possui sensores acelerômetro, giroscópio e magnetômetro. Acoplado no braço ou antebraço, diretamente sobre a pele. De fácil utilização, com uso

imediatamente após acoplado, possui funções de calibramento para cada usuário antes da utilização. O Myo é capaz de fornecer informações como: qual braço ele está colocado; sua orientação; se está direcionado ao punho ou ao cotovelo. Possui uma bateria recarregável de Lítio, 8 segmentos conectados por material expansível, permitindo que se adapte ao tamanho do braço do usuário, uma porta micro-USB, a logo, em LED, que mostra o estado de sincronismo atual, e um LED indicador que de acordo com a cor, representa um estado diferente (figura 3) (MURILLO; MORENO, 2016).

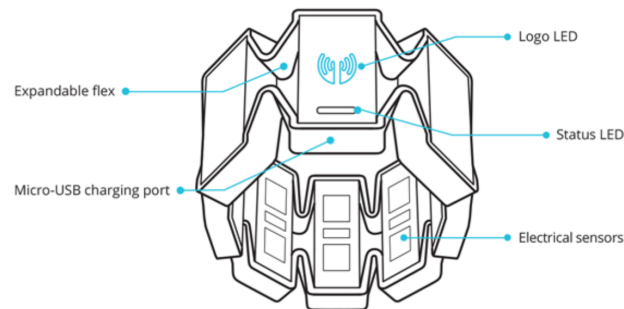


Figura 2: Características do Myo, por Thalmic Labs - Fonte:

https://www.researchgate.net/figure/The-Myo-gesture-control-armband-from-Thalmic-Labs-Figure-adapted-from_fig4_324040549

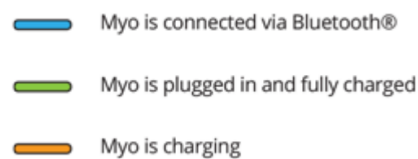


Figura 3: LEDs característicos do Myo e suas definições, por Thalmic Labs - Fonte:

https://www.researchgate.net/figure/The-status-LED-and-the-colors-meaning-adapted-from-11_fig3_334794917

O Myo possui 8 sensores de eletromiografia (EMG) e uma Unidade de Medição Inercial de 9 eixos, com acelerômetro de três eixos, giroscópio de três eixos e magnetômetro de três eixos (NYMOEN; HAUGEN; JENSENIUS, 2015). Os eletrodos do Myo são dispostos de maneira circular e os principais músculos captados são o músculo Extensor Digitorum e o músculo Flexor Digitorum (ARIEF; SULISTIJONO; ARDIANSYAH, 2015). Por meio dos impulsos elétricos captados, o Myo reconhece os gestos a serem executados. Ele possui uma configuração default, mas fazer a calibração é ideal, pois cada usuário tem peles diferentes e produzem reações diferentes em certos casos, possuem tamanho e forma muscular diferentes, de forma que facilite a leitura dos sensores (SATHIYANARAYANAN; RAJAN, 2016). Os gestos reconhecidos pelo myo são mostrados na figura 4.

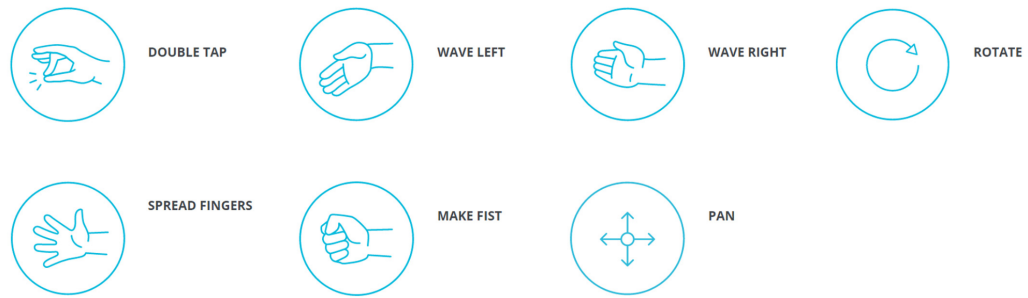


Figura 4: Gestos reconhecidos pelo Myo, imagem por Myo Market: Diagnostics Page -

Fonte: <https://drjpau.wordpress.com/tag/interface/>

A interface que age como intermédio entre o Myo e o computador é o Myo Connect. Responsável pela comunicação entre o myo e outros softwares, dando alguns controles básicos. Através dele, é possível fazer o upload de *scripts* próprios a fim de controlar o Myo e manipular suas informações recebidas. Além disso, oferece uma interface, Myo Armband Manager, onde é possível verificar o número de Myos conectados, o nível de bateria e a opção de conectar o bracelete (SANTOS, 2016).

O Myo possui uma página na internet chamada Diagnostics Page. Nela é possível visualizar as respostas dos sensores em tempo real. Pode-se ser feita a análise da resposta dos sensores IMU de acordo com cada movimento no espaço e a resposta dos sensores EMG para cada gesto executado. Além disso, ela exibe níveis de sinal de intensidade bluetooth, nível atual de bateria do dispositivo, data e hora da execução de movimentos, informações de posição, giroscópio e acelerômetro atuais.(SANTOS, 2016).

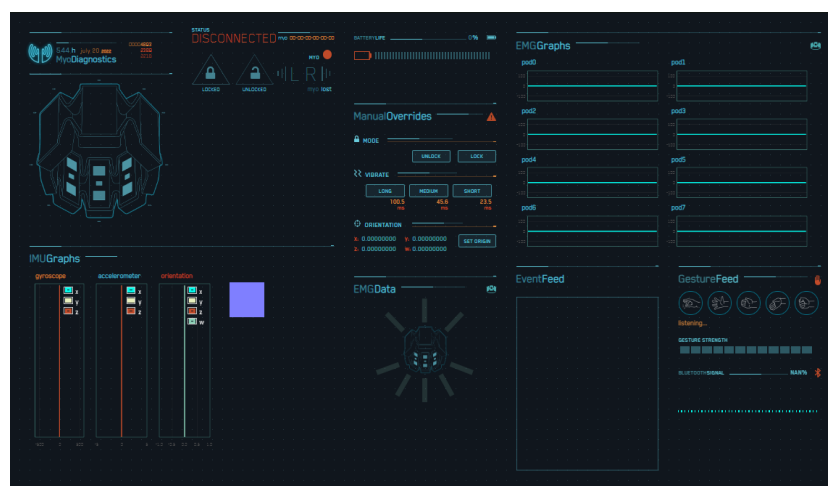


Figura 5: Myo Diagnostics Page - <http://diagnostics.myo.com>

Ao ser feita a execução de um movimento, o comando nervoso vindo do cérebro é transportado até o músculo em forma de sinal elétrico. A

eletromiografia descreve o monitoramento dessa atividade elétrica dos músculos, representando os potenciais de ação do sarcolema como efeito da voltagem em função do tempo. O sinal eletromiográfico pode ser afetado por propriedades musculares, anatômicas e fisiológicas, pelo controle do sistema nervoso periférico e a instrumentação utilizada para medição. Esse sinal consiste na soma algébrica de todos os sinais adquiridos em certa área (ENOKA, 2000). A eletromiografia é essencial ao ser feita a análise do movimento, pois fornece informações em relação ao tempo de início e fim da ativação da musculatura envolvida no movimento, a intensidade dessa ativação e a duração da atividade. O sinal EMG pode ser composto por uma combinação de sinais, sendo classificado em monopolar, bipolar e sistema multipolar (MARCHETTI; DUARTE, 2006). A captação do sinal de eletromiografia pode ser feita com eletrodos sobre a pele, chamada EMG de superfície, debaixo da pele mas acima do músculo, chamada EMG subcutânea, e dentro das fibras do músculo, chamada de EMG intramuscular (ENOKA, 2008).

4.2. Estudos preliminares

O aluno já possuía experiência com a implementação de servidores em NodeJS, porém não possuía conhecimentos da linguagem utilizada no projeto original, ou familiaridade com o Unity. Por isso, foi necessário tempo dedicado exclusivamente no aprendizado, através do mesmo curso realizado pelo próprio Pedro.

O aluno, por conta dos próprios traumas nas mãos, esteve em contato com diversos fisioterapeutas e terapeutas ocupacionais, e em especial, a Dra. Ana Carla Nogueira Santos, atual vice-presidente do Comitê Federal de Fisioterapia e Terapia Ocupacional, crucial para a elaboração desse projeto. Foi feito um estudo quanto aos movimentos adotados no código, apresentado aos profissionais para estabelecer as alterações a serem implementadas.

4.3. Estudos conceituais e de tecnologia

Com a elaboração do projeto em Unity, foi necessário o aprendizado da linguagem e do manuseio do motor Unity. Ao longo do aprendizado, foi feita a verificação do encapsulamento executado e boas práticas no código.

Felizmente, não houve a necessidade de fazer alterações quanto a essa questão, apenas a adição de novas features. Para o desenvolvimento deste projeto, foi feito o curso The Ultimate Guide to Game Development with Unity 2019 , criado por Jonathan Weinberger.

Após a conclusão do estudo do jogo, foram realizadas apresentações aos profissionais de saúde em busca de obter possíveis correções ou adições à mecânica do jogo. E com elas, concluída a necessidade de influenciar o usuário a exercer força para a execução de determinado movimento, e para atingir essa diretriz, foi usado o Myo Armband, que possui alguns dos movimentos já mapeados, capaz de captar o esforço do usuário ao executá-los. Este requereu profundo estudo sobre suas funcionalidades, funcionamento e referências quanto à sua utilização na área de reabilitação. Assimilá-lo ao código foi o maior desafio encontrado. Demandou a criação de uma API, implementada em Javascript com NodeJS, para comunicar-se com a aplicação. Por ter sido comprado pela Google, todas as SDKs relacionadas a Unity foram apagadas, assim como sua documentação. Porém, com a existência do Diagnostics Myo, aplicação em NodeJS e ReactJS que consomem uma biblioteca em Javascript, foi possível a realização de um servidor próprio que realize os procedimentos necessários a fim de gerar as respostas necessárias.

4.4. Testes e protótipos para aprendizado e demonstração

Após obter o hardware que seria adicionado à aplicação, foram feitos inúmeros testes para verificar se seria cabível ao projeto. Inicialmente feitos testes de funcionalidade através da API dos desenvolvedores, Myo Connect e Diagnostics Myo:

- **Funções básicas:**

1. Se a captação de movimentos funciona como esperado em ambos os braços;
2. Se todas as placas magnéticas estão funcionando corretamente ;
3. O funcionamento do acelerômetro, giroscópio e orientação do Myo no espaço;
4. Se a escolha do braço utilizado importa.

- **Testes para casos de borda e limitações:**

1. **Qual o requerimento de força, em uma das posições já mapeadas, para que seja reconhecida pelo hardware como “forte e fraco”?**

O funcionamento do Myo não é direcionado a força, mas a resposta aos impulsos nervosos, ou seja, ele interpreta o esforço para o gesto como força de execução através da leitura dos picos e vales exibidos no Diagnostics Myo, para cada pod.

2. **É possível calibrar o funcionamento dele para meus impulsos nervosos pessoais?**

Sim, visto que foi implementado de forma que possa salvar no Myo Connect o comportamento dos seus impulsos nervosos para qual o esforço que julgar necessário.

3. **A forma que é colocada no braço impacta o comportamento de como o software trata a informação?**

Foi encontrada sim, diferença no comportamento default ao alterar a forma de usar o Myo. O comportamento ideal foi percebido quando o mesmo possuía a porta USB do dispositivo voltada para o punho. Quando voltada para o cotovelo, ocorria a troca de interpretação de gestos x e y.

4. **Qualquer pessoa seria capaz de usar?**

Adultos e adolescentes, sim, porém crianças muito novas, com braços menores, teriam dificuldade de manter o bracelete estático na posição recomendada.

Apresentados resultados positivos, sem falhas quanto às funções básicas, e rasas as limitações encontradas, foi concluído que seria capaz de atender as demandas. Este possui instruções básicas desde o primeiro uso:

- Sobre como deve ser colocado;
- Suas funcionalidades extras, como mapeamento de teclado e libs internas com softwares de música;
- Tutorial sobre os gestos e como executá-los;
- Possibilidade de manter o funcionamento default, ou ser calibrado para seu braço;

- Inclui casos de borda para pessoas com braços delgados e carnudos por possuir travas que impedem a borracha de deformar para os braços mais esbeltos.

4.5. Método

O desenvolvimento do projeto foi dividido nas seguintes etapas:

- Definição da feature, seu comportamento e sua estrutura;
- Protótipo Inicial;
- Teste Piloto;
- Correções e calibrações;
- Teste Final.

4.5.1. Definição da feature, seu comportamento e sua estrutura

O primeiro passo do projeto foi o estudo aprofundado sobre o código deixado por Pedro, em paralelo com o aprendizado da linguagem e motor, através do curso que o mesmo utilizou para a realização do jogo. Após o entendimento do código e encapsulamento, foi feita uma pesquisa sobre seus movimentos implementados, e a forma como foram implementados em busca de gerar uma apresentação para os profissionais de saúde voltados para a fisioterapia e terapia ocupacional.

As opiniões foram todas muito similares quanto às alterações necessárias, sejam ideais do jogo ou com qual frequência cada gesto seria exigido. Entretanto, o destaque a ser pontuado é um gesto em questão que será chave para a continuação deste projeto: o exercício de fechar o punho. Fechar o punho, rotação de pulso, flexão de pulso e flexão de dedos foram os gestos implementados a serem interpretados pelo Leap Motion, escolhidos de forma excelente, elogiados pelos profissionais. Porém, um deles funciona de forma ludibriosa, pois o exercício tem como questão objetiva o aumento da força de pegada, de compressão pelos punhos, no entanto essa percepção não é possível através das câmeras infravermelhas, estas usadas pelo Leap Motion para captação do movimento das mãos, surgindo a motivação para o projeto.

Essa questão foi levada aos orientadores: “De que forma seríamos capazes de captar a força do usuário, como um dinamômetro?”. Assim, foram criadas diversas hipóteses, por mim, antes da conversa com os professores, sendo elas: integração de dinamômetros digitais, bolas de borracha

normalmente voltadas para ansiedade, handgrips (funcionam como dinamômetros analógicos), todos integrados com um microcontrolador a fim de estabelecer medidas, limites e comunicação com a aplicação. Até a apresentação do Myo Armband pelo professor Alberto. O Myo apresenta resultados e demonstrações de usos terapêuticos há anos, e graças às suas especificações encontradas na seção 4.1, foi concluído como a melhor opção, no momento, para o desenvolvimento da feature. O problema é que por não ser comercializado oficialmente desde 2018, esta não seria a escolha ideal para as massas, mas é a mais prática e funcional para a solução, visto que o laboratório tem posse do Myo.

Depois de estabelecidas as ferramentas, foram definidas duas formas de implementação:

1. Utilizar bibliotecas direcionadas para a comunicação entre o Myo e o Unity: Mostrada inicialmente como a opção ideal, por ser implementada diretamente no motor Unity, teria a menor latência de resposta ao usuário. Porém, quando foi comprado pela Google, toda a documentação e SDKs referentes ao uso do Myo com Unity foram retiradas do ar;
2. Criar um servidor em NodeJS para consumir a mesma biblioteca utilizada pelo Diagnostics Myo: Viabilidade retratada pela existência do Diagnostics Myo, com a visualização de respostas, conexão automática com o dispositivo Myo conectado e gráficos em tempo real da resposta nervosa das contrações musculares. Foi planejada uma solução pequena para interpretar os sinais recebidos, e por meio de chamadas HTTP (GET), a comunicação com o jogo.

A escolha, por boa parte do projeto, foi a primeira. Diminuta latência, menos etapas a serem concluídas no Set Up pelo usuário para iniciar o jogo, menos dependências, todas elas levadas em consideração e determinadas como essenciais para a produção da solução, contudo, infelizmente, não foi possível. Já que, como abordado acima, as dependências necessárias para o acoplamento do Unity com Myo, foram retiradas 4 anos atrás, tornando inviável a solução escolhida. Com isso, a opção de criar um servidor para a comunicação foi a usada no projeto.

4.5.2. Protótipo inicial

Nesta etapa, o objetivo foi criar um protótipo com o mínimo possível para ter um jogo funcional com o Myo. Primeiramente, estabelecida a comunicação

entre o Myo e o servidor. Segundamente, a adição de um módulo para encapsular o tratamento do esforço do usuário e suas chamadas para sobrescrever os valores default, sem instauração de novas telas ou instruções, a parecer que seu esforço físico seria refletido no jogo. O servidor interpreta um vetor de EMG (vetor com valores de -100 a 100, retratando o tamanho de onda gerado pelo esforço do usuário), em um espaço de tempo de 100 milissegundos, utilizando os valores absolutos mais altos de cada período de coleta de informação para gerar uma média a ser enviada pelo servidor para a aplicação.

4.5.3. Teste Piloto

Realizado com 5 pessoas com idades de 20 a 60 anos, sem deficiências motoras, sem conhecimento do objetivo terapêutico do jogo ou experiências com a realidade virtual.

É importante destacar que não foi aplicado nenhum questionário formal, mas foi coletada informação via observação direta e pela técnica “Think aloud”, onde os usuários falaram enquanto testaram a aplicação e deram feedback após o jogo. Apesar de terem achado a experiência divertida, diversos pontos negativos foram apontados.

Inicialmente, a reclamação a respeito do que foi implementado se destacou por conta da magia referente ao raio, proveniente do gesto de fechar o punho, ser muito fraca. Após explicar a funcionalidade do armband, os resultados foram melhores, mas ainda destacavam que, como magia, era muito fraca. Ela gerava muito esforço para chegar a um dano satisfatório nos inimigos, mesmo aos que sofrem o dobro do dano comum, sentindo dificuldade de passar por sua fase tutorial. A resposta do servidor à aplicação se mostrou imperceptível e não teve reclamações.

4.5.4. Correções e calibrações

Após o teste piloto, o foco foi aumentar o valor multiplicador da força recebida pelo servidor e gerar recompensas aos usuários capazes de alcançar uma faixa de valor de esforço. Através de testes locais, foi definido como valor aritmético ideal para o multiplicador, de $1,4 * \text{o valor médio absoluto calculado}$, este usando os resultados de saída dos sensores EMG dos últimos 0.1 segundos de esforço. Para referência, sem o Myo, o valor padrão de dano

gerado aos inimigos era de 25. Com a adição do Myo, inicialmente, o valor absoluto calculado sem o multiplicador, para uma faixa de esforço média-forte era de 17,5. Quando adicionado o multiplicador, ao gerar essa mesma faixa de esforço, o dano causado ao inimigo seria de 24,5, onde a mecânica de dano em dobro a determinados inimigos, com 50 pontos de vida, ficariam com 1 ponto de vida. O motivo por trás disso é para fazer com que a habilidade não se torne poderosa demais. O raio é uma magia de acerto em cadeia, então inimigos próximos ao principal recebem o mesmo ataque. Se este fosse muito forte, nenhum outro poder seria utilizado, pois é o mais prático, mais fácil de executar e acerta o maior número de inimigos. Porém, dado que o objetivo da feature é estimular o uso do poder e do esforço para ativá-lo, também foi implementado um Buff (aumento na capacidade destrutiva). Se o usuário fizer esforço de valor absoluto médio recebido pelo Myo acima de 71,4, o valor do esforço a ser enviado pelo servidor é dobrado. Ou seja, está sendo gerado ao usuário um sistema de recompensa por seu esforço em busca de influenciar o uso deste.

4.5.5. Teste final

Realizados com as mesmas pessoas em Testes Iniciais.

Resultados positivos quanto às alterações implementadas, o Raio se tornou o favorito dos usuários. Este, até a versão original, já era o favorito, mas de acordo com os usuários: “Ser capaz de estar cercado com pouca vida e sobreviver por um ataque que acerta todos e ainda mais forte se eu fizer força? Muito legal”. Inclusive, através dos testes, foi possível perceber a limitação motora de um dos usuários, a maioria não teve problemas ao executar os poderes depois de praticados, com exceção de um, o Kamehameha, poder gerado pela flexão dos dedos ao encontro do polegar. Para este, um dos usuários não conseguiu executar com a frequência desejada, e, ao comparar com os outros, percebeu que seus outros dedos, idealmente eretos quando não ao encontro do polegar, se encontravam dobrados.

4.6. Plano de Ação

Apresentam-se a seguir o plano das atividades feitas para atingir o objetivo proposto. Inicialmente, descrevem-se as atividades e posteriormente apresentam-se os cronogramas para o Trabalho Final I e II.

Atividades Projeto Final I	Set.	Outubro	Nov.	Dez.
Estudo				
- Escrita e entrega proposta				
- Estudo da linguagem C# e do Motor Gráfico Unity				
- Estudo do Software implementado pelo Pedro Sousa Meireles				
Apresentações para profissionais				
- Profissionais de Fisioterapia				
- Profissionais de Terapia Ocupacional				
Pesquisa				
Pesquisa sobre formas de acessar a força gerada pelo usuário				
- Escrita do relatório Projeto Final 1				
- Entrega do relatório Projeto Final 1				

Tabela 1: Cronograma do Projeto Final I

Atividades Projeto Final II	Mar.	Abril	Maio	Jun.	Jul.	Ago.
Estudo						
- Aprendizado de novas práticas e técnicas usadas no Unity						
- Estudo direcionado para avaliar precisão necessária para validação de movimentos						
- Estudo sobre o Myo Armband e suas bibliotecas para com o Unity						
Projeto						
- Coleta de Feedbacks dos usuários e profissionais de saúde						
Desenvolvimento						
- Implementação das melhorias no código quanto a eficiência e tempo de resposta						
- Resolução de possíveis Bugs e limpeza de código						
- Escrita do relatório Final						
- Entrega do relatório						
- Defesa do Projeto Final						

Tabela 2: Cronograma do Projeto Final II

5. Projeto e especificação do sistema

5.1. Versionamento e planejamento

Durante todo o desenvolvimento utilizou-se o Github para versionamento do projeto, garantindo que alterações não fossem perdidas e pudessem ser documentadas.

Também utilizou-se o Trello como ferramenta de organização das tarefas que deveriam ser realizadas, o que já estava em desenvolvimento e o que havia sido concluído.

5.2. Setup do projeto

O primeiro passo foi abrir o projeto em Unity, utilizando a versão 2019.2.8f1, a mesma versão na qual foi implementada. Fazer a instalação do software da Ultra Leap para o uso do Leap Motion, este não possui versão especificada, foi utilizada a versão Gemini 5.6.1. Fazer a instalação do Myo Connect, software responsável pela interpretação de gestos e API necessária para comunicação do Myo com o PC e Servidor implementado.

Para o desenvolvimento do Servidor, foi criado um projeto em Node na versão 12.16.1 através do NPM, mas gerado de forma que qualquer versão possa ser usada. Para a comunicação do Servidor com a aplicação Unity, foi instalada a biblioteca Express, ferramenta que usa o protocolo HTTP: “HTTP é um protocolo de transferência que possibilita que as pessoas que inserem a URL do seu site na Web possam ver os conteúdos e dados que nele existem. A sigla vem do inglês Hypertext Transfer Protocol.” E para transferências de informações entre o Servidor e o Myo, foi utilizada a biblioteca de integração, o myo.js, instalada por meio do NPM.

5.3. Desenvolvimento

5.3.1. Unity

Como a aplicação foi feita em Unity, será útil entender um pouco sobre como esse game engine funciona para embasar as implementações expostas a seguir. O desenvolvimento em Unity é baseado em cenas, *GameObjects* e componentes:

- Cenas são compostas de um cenário e objetos, seja uma tela de menu ou *gameplay* do jogo;
- *GameObjects* são os objetos responsáveis por representar qualquer objetivo no jogo. Podem representar inimigos e poderes, por exemplo, assim como um gerenciador de fases do jogo. Esses objetos são contêineres para os componentes;
- Componentes implementam as funções do jogo. Na Unity, existem inúmeros componentes disponíveis, responsáveis por atrelar atributos aos objetos. Por exemplo: colisão, iluminação, até comportamentos gravitacionais. Mas esses comportamentos também podem ser desenvolvidos via código para alcançar casos específicos referentes aos objetivos daquele objeto.

5.3.2. Exercícios

O jogo foi programado de forma que a cada quadro é verificada a existência de um exercício em execução. Caso não haja, temos um loop na tentativa de identificar o início da execução de um gesto. A partir do momento em que é identificado até a conclusão de seu uso atual, apenas ele será executado, impossibilitando o uso de 2 magias ao mesmo tempo. A estrutura descrita é demonstrada na Figura 6.

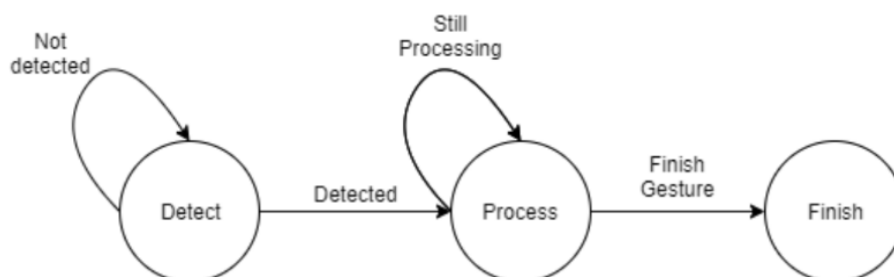


Figura 6: Autômato que controla a detecção dos exercícios - Fonte:
<https://www.sbgames.org/proceedings2020/JogosSaudeFull/209323.pdf>

“O primeiro estado (Detect) verifica se o jogador está fazendo a postura correta para iniciar o exercício determinado. Caso o exercício seja detectado, é iniciado o seu processamento (Process). Essa etapa normalmente consiste de um indicador que mostra ao jogador que o exercício foi detectado e ele deve realizá-lo por determinada quantidade de tempo para que a magia possa ser conjurada. Esse indicador varia com cada magia e será melhor explicado mais à frente. Por fim, após o processamento ter terminado, o jogador deve realizar um gesto final para que o feitiço tenha efeito (Finish). Só então o processo termina e o ciclo recomeça até que um exercício seja detectado. A qualquer momento durante esse processo é possível cancelar a execução do exercício fechando o punho da mão direita” - MEIRELES, PEDRO SOUSA, 2019.

5.3.2.1. Punho

O exercício de menor complexidade a ser executado pelo jogador. Em primeiro momento, tratava-se do simples movimento de fechar a mão em um punho, posteriormente abrindo-a novamente, como mostra a Figura 7. Porém, com a adição do Myo Armband, temos a captação do esforço utilizado para execução do gesto. De acordo com Pedro Sousa Meireles, em seu Relatório do Projeto Final: “Um detalhe importante da implementação da detecção deste exercício é que a mão deve estar apontando para frente no momento que o punho é fechado, uma vez que outros exercícios também utilizam a mão fechada em um punho. Isso foi feito para que não houvesse interseção entre os movimentos de cada exercício, evitando que um fosse detectado ao se executar o outro.”

Com o fechar o punho, ocorre a passagem do estado *Detect* para o *Process* (Figura 7). A partir de então, 2 processos são executados em paralelo:

1. Uma esfera é materializada na mão do jogador. Em busca de simular a necessidade de um tempo de espera para sua ativação, seu diâmetro cresce até atingir um máximo;
2. A aplicação Unity faz uma requisição para o servidor, em busca da força utilizada pelo usuário quando executou o gesto do punho.

Até finalmente, depois de a esfera ter atingido seu máximo, o jogador pode abrir a mão. Este é o momento em que é executada a magia.

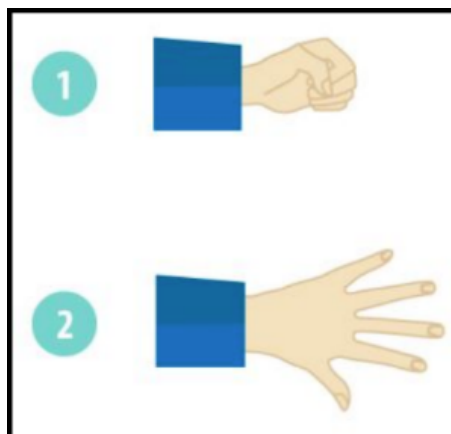


Figura 7: Movimento do exercício do punho -

<https://www.saebo.com/stroke-exercises-for-your-body/>

Para esse exercício, originalmente, foi atribuída a magia ofensiva mais fraca, uma vez que é o exercício de menor complexidade. Porém, com a adição do Myo, o potencial ofensivo aumenta significativamente, com destaque nos níveis de dificuldade mais baixa. Ao abrir a mão, um raio é disparado na direção que a palma do jogador estiver apontada (Figura 8). Esse raio tem como característica a capacidade de formar cadeias, ou seja, afeta inimigos próximos ao que o ataque foi direcionado.

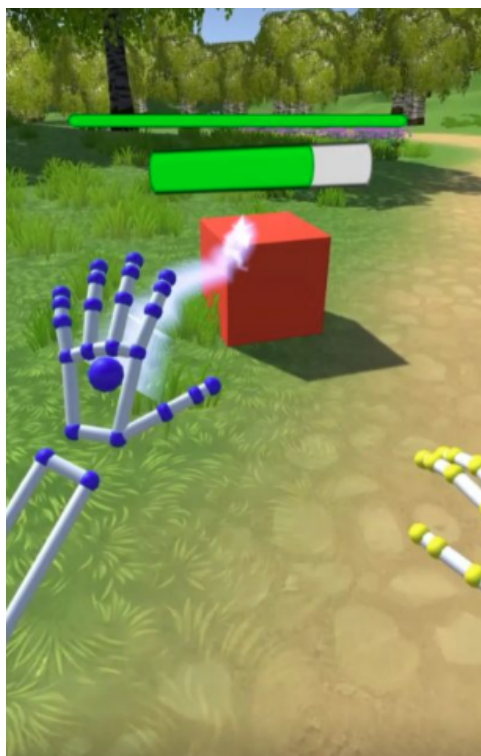


Figura 8: Raio realista utilizado no jogo, obtido da Unity Asset Store - Fonte: Sleght of Hands

5.3.3. Inimigos

Por meio do estudo de seu comportamento, percebe-se que cada inimigo tem uma propriedade defensiva diferente, relativa a cada magia existente. Estes são capazes de possuir os elementos fogo, terra, gelo, raio e normal. As magias e suas características funcionam da seguinte forma:

Ataque\Defesa	Normal	Fogo	Terra	Gelo	Raio
Normal	1x	1x	1x	1x	1x
Fogo	1x	1x	2x	1/2x	1x
Terra	1x	1/2x	1x	1x	2x
Gelo	1x	2x	1x	1x	1/2x
Raio	1x	1x	1/2x	2x	1x

Tabela 3: Relação ataque e defesa elementar de cada inimigo.

Podemos perceber que o usuário possui a propriedade normal defensiva. O motivo por trás disso é para que todo dano recebido seja o mesmo, independente do inimigo a atacá-lo. A importância dessa informação sobre a mecânica do jogo para o projeto será retratada na seção 5.5.3.

5.4. Servidor

5.4.1. Estrutura e desenvolvimento

Solução implementada em Javascript, utilizando o NodeJS com Express para o seu consumo. A aplicação em Node será a ponte de comunicação entre o Myo e o jogo, através de chamadas HTTP GET, a API verifica sua conexão com o hardware e o software, processa os valores absolutos providos pelo Myo e envia para o jogo. O valor calculado pelo projeto fica armazenado em uma classe do jogo: Strength. Por possuir características relacionais apenas com um dos exercícios, apenas o ChainLightning, magia gerada pelo gesto do punho, explicado na seção 5.4.4.1, é componente filho da nova classe criada.

A aplicação implementada é dividida em 3 módulos:

- O servidor, onde é inicializado e estabelece a conexão com o Myo e Myo Connect;
- Uma pasta de utilitários para o cálculo dos valores médios recebidos;

- Uma pasta de controllers, visto que trabalham no tratamento de chamadas do servidor.

5.4.2. Estabelecer conexão

Ao inicializar o programa temos o estabelecimento da conexão entre o Myo e o servidor através da biblioteca Myo.js, criada por Scott Tolksdorf. Esta faz a conexão via bluetooth e websocket, para ser mantida a conexão. Para contextualização, WebSockets é uma tecnologia avançada que torna possível abrir uma sessão de comunicação interativa entre o navegador do usuário e um servidor. Com esta API, você pode enviar mensagens para um servidor e receber respostas orientadas a eventos sem ter que consultar o servidor para obter uma resposta. Ainda utilizando por baixo dos panos o protocolo HTTP, se torna atual forma mais rápida de comunicação entre *client* e *server*.

5.4.3. Utilitários ou serviços

Neste módulo é feito o cálculo dos valores a serem enviados para a aplicação em Unity. Este possui 3 fases:

1. Faz a leitura do vetor EMG, e pega o maior valor absoluto;
2. Recebe os valores absolutos mais altos e faz uma média;
3. Recebe a média de valores absolutos mais altos, executa um multiplicador de 1,4x ao valor recebido e faz uma verificação para ver se o valor depois de multiplicado pelo coeficiente de ajuste é maior do que 1. Caso seja, o valor final será o dobro do modificado.

Os valores utilizados como coeficiente de ajuste e *buff* final o valor encontrado são resultados de testes locais, sobre quanta força é necessária para alcançar o valor de dano original. Antes, o valor para esta magia era de 25 dado ao inimigo. Após o ajuste, para atingir o valor inicial de 25, ou próximo, se faz necessário um nível de esforço médio-forte. Isso ocorre por conta do resultado gerado pela API, em sua maioria, por estar em uma faixa de valores de 0 a 100, o valor mais repetido foi o 70. Através do coeficiente de ajuste, o objetivo é aumentar a frequência de valores próximos ao original. E recompensar os usuários que ultrapassarem o nível de esforço de 70, dando-lhes o dobro do potencial destrutivo, com a faixa de valores entre 0 e 2.8 a ser multiplicados pelos 25 originais.

5.4.4. Controllers

Uma das camadas da arquitetura, responsável por fazer a recepção das requisições e passar adiante apenas as informações relevantes. Como dito em sua definição, a camada de Controllers faz o "*primeiro contato*" com as requisições, enviando a camada de Services apenas as informações relevantes para completar a requisição. Além disso, essa é a camada que irá enviar a resposta ao cliente, seja ela positiva ou negativa.

Na aplicação, temos 2 controllers: um responsável pela comunicação de requisição e resposta para o jogo e um responsável pela comunicação entre o Myo e o servidor. O primeiro, faz a verificação de existência da rota requerida, da conectividade entre o Myo e o servidor, e faz uso das funções utilitárias para gerar a resposta necessária, caso não esteja conectado, a resposta é negativa (400), com a explicação do motivo da falha, e caso contrário, devolve a resposta esperada com sucesso (200) e a informação necessária para o funcionamento do software. Atualmente, só se faz necessária a existência de uma rota para esse projeto, "/strength", que devolve um objeto com o valor a ser utilizado e se a requisição foi um sucesso ou não. O segundo trabalha mais de forma a interpretar os POSTs para o servidor, salvando os valores recentes em um vetor, ele trata os eventos gerados pelo Myo.

6. Implementação e Avaliação

Após a escolha da primeira opção mencionada na seção 4.5, foram passados 2 meses em busca de maneiras de integrar o Myo e o Unity através da biblioteca, descontinuada, da Thalmic Labs, mas nenhum resultado se mostrou promissor. Então, foi concluído que se faz necessária a mudança de abordagem. No início, pesquisas sobre visualização e testes funcionais do Myo, foi encontrado o Diagnostics Myo, API em NodeJS consumida por um client em ReactJS, ambos que eu já possuía conhecimento profundo sobre, concluí ser ideal o consumo da API para estabelecer a conexão com o Myo. Infelizmente, não foi encontrada a URL do servidor para que fôssemos capazes de consumir diretamente, por isso fiz um estudo sobre a biblioteca que este consome e implementei um servidor local a ser consumido pela aplicação através do uso do Express - framework para aplicativo da web do Node.js mínimo e flexível que fornece um conjunto robusto de recursos para aplicativos web e móvel - e Myo.js - faz a comunicação entre o servidor e o Myo através de websockets, capaz de

obter os eventos dos gestos, o tornando customizável, e valores do EMG que utilizamos para interpretar o esforço do usuário na aplicação.

Para avaliar a feature desenvolvida, foram executados testes em pessoas que não precisam de tratamento. Infelizmente, por motivos de saúde, não fui capaz de testar com pessoas em necessidade do tratamento direcionado pela aplicação. Os resultados dos testes estão na seção 4.4.

7. Considerações Finais

A escolha desse tema foi por estar familiarizado com o ramo terapêutico, por ter passado por todo o processo de reabilitação de mãos. Eu tive a experiência de ter passado por um caso mais leve do que os retratados, ao deslocar alguns de meus dedos, não busquei o tratamento terapêutico no primeiro momento, e mesmo depois de encontrá-lo, não o executei em casa, proporcionando uma deformidade nos dedos. Felizmente, minhas contribuições e experiências afetam diretamente os usuários, já que não houveram contra indicações ou reclamações quanto a adição da feature, esta tem potencial para ajudar inúmeras pessoas que passam por este problema. Graças a esse trabalho, fui instigado a aprender novas linguagens e seus fundamentos, fazer a manipulação de eventos em hardwares fora de linha e buscar as informações necessárias para sua manipulação.

Se tivesse a oportunidade de começá-lo agora, teria feito escolhas mais diretas quanto às ferramentas utilizadas, adição de outras features estéticas, adicionado o servidor a um domínio para acesso comum, com um *client* em ReactJS para melhor experiência ao usuário, sem necessidade da dependência de obter o código do servidor e ter o Node instalado para acessar essa funcionalidade. Faria a refatoração do tutorial, com adição de instruções mais completas, deixaria clara a existência do cancelamento de magia, por ter se mostrado muito útil aos usuários após ter conhecimento de sua existência e traria uma opção no menu que pudesse retirar o tutorial como obrigatório para a execução do jogo. São diversas oportunidades para o próximo aluno poder escolher, mas recomendo priorizar as questões mais comuns em jogos: a adição do cancelamento de magia no tutorial e a opção de pular o tutorial.

8. Referências

WORLD HEALTH ORGANISATION & THE WORLD BANK. The world report on disability. Disability and Society, v. 26, n. 5, p. 655–658, 2011.

PEDRO SOUSA MEIRELES; INF, R. Projeto Final II :Ambiente Virtual para Apoio à Reabilitação de Pessoas com Dificuldades Físicas nos Membros Superiores, 2019.

MIGUEL HENRIQUE JUVÊNCIO FREITAS; INF, Projeto Final: Utilização de dispositivos tipo myo para manutenção de equipamentos de distribuição de energia elétrica em ambiente virtual, p.27-30, 2017. Uberlândia - http://www.antigo.feelt.ufu.br/sites/feelt.ufu.br/files/TCC%2011311ECP007%20Miguel%20Henrique%20Juvêncio%20Freitas_FINAL.pdf

GABRIEL SILVESTRE; Controller e Service - Uma breve introdução, 2022 - <https://dev.to/gabrielhsilvestre/controller-e-service-uma-breve-introducao-24hk>

CARCI BLOG - OFICIAL; Conheça os principais equipamentos de fisioterapia para uma clínica de sucesso; <https://blog.carcioficial.com.br/equipamentos-de-fisioterapia-para-uma-clinica-de-sucesso/>

TIME; This Futuristic Armband Lets You Control Your Computer Like Magic, 2016; <https://time.com/4173507/myo-armband-review/>

THALMIC LABS; Plugin for Myo integration on Unity, 2015; <https://github.com/thalmiclabs/myo-unity>

GAME VÍCIO; Projeto MYO foi finalizado, sistema de controle por gestos chega ao mercado neste ano, 2014; <https://www.gamevicio.com/noticias/2014/06/projeto-myo-foi-finalizado-sistema-de-controle-por-gestos-chega-ao-mercado-neste-ano/>

MOZILLA WEB DOCS; WebSockets, 2021 - https://developer.mozilla.org/pt-BR/docs/Web/API/WebSockets_API

PCWORLD; This is not a toy: Oculus Rift's virtual talents could transform real lives, 2013 - <https://www.pcworld.com/article/452812/this-is-not-a-toy-oculus-rifts-virtual-talents-could-transform-real-lives.html>

TODAY ONLINE; Poly students create low-cost, gesture-controlled rehabilitative glove for stroke patients, 2019 - <https://www.todayonline.com/singapore/poly-students-create-low-cost-gesture-controlled-rehabilitative-glove-stroke-patients>

MEDGADGET; MYO Armband Uses Electromyography to Control External Devices (w/video), 2013 - <https://www.medgadget.com/2013/02/myo-armband-uses-electromyography-to-control-external-devices-wvideo.html>

HUSSEI HASAN; University of Thi-Qar; A Wearable Rehabilitation System to Assist Partially Hand Paralyzed Patients in Repetitive Exercises, 2019 https://www.researchgate.net/publication/334794917_A_Wearable_Rehabilitation_System_to_Assist_Partially_Hand_Paralyzed_Patients_in_Repetitive_Exercises

THALMIC LABS; Diagnostics Myo; <http://diagnostics.myo.com>

THALMIC LABS; Myo.js, 2014; <https://www.npmjs.com/package/myo>

TAYFUN UZUN; MyoSharp, 2013 - <https://github.com/tayfuzun/MyoSharp>

MARCHETTI, P. H.; DUARTE, M. Instrumentação em eletromiografia. Laboratório de Biofísica, Escola de Educação Física e Esporte. São Paulo: Universidade de São Paulo, 2006.

ENOKA, R. M. Neuromechanics of human movement. [S.l.]: Human kinetics, 2008.

SANTOS, S. G. Estudo sobre o myo: Determinação de uma função de transferência. 2016.

SATHIYANARAYANAN, M.; RAJAN, S. Myo armband for physiotherapy healthcare: A case study using gesture recognition application. In: IEEE. Communication Systems and Networks (COMSNETS), 2016 8th International Conference on. [S.l.], 2016. p. 1–6.

ARIEF, Z.; SULISTIJO, I. A.; ARDIANSYAH, R. A. Comparison of five time series emg features extractions using myo armband. In: IEEE. Electronics Symposium (IES), 2015 International. [S.l.], 2015. p. 11–14.

NYMOEN, K.; HAUGEN, M. R.; JENSENIUS, A. R. Mumyo—evaluating and exploring the myo armband for musical interaction. Louisiana State University, 2015.

MURILLO, P. U.; MORENO, R. J. Multi user myographic characterization for robotic arm manipulation. International Journal of Applied Engineering Research, v. 11, n. 23, p. 11299–11304, 2016.

GANIEV, A.; SHIN, H.-S.; LEE, K.-H. Study on virtual control of a robotic arm via a myo armband for the selfmanipulation of a hand amputee. Int. J. Appl. Eng. Res, v. 11, n. 2, p. 775–782, 2016.

DE SOUZA, IVAN; - Rock Content, 2019
-<https://rockcontent.com/br/blog/http/#:~:text=HTTP%20%C3%A9%20um%20protocolo%20de%20ingl%C3%AAs%20Hypertext%20Transfer%20Protocol.&text=Um%20dos%20caminhos%20para%20aumentar,neg%C3%B3cio%20%C3%A9%20o%20Marketing%20Digital>

FERNANDES, FLÁVIA GONÇALVES - O USO DO DISPOSITIVO VESTÍVEL MYO EM JOGOS PARA REABILITAÇÃO DE CRIANÇAS COM DEFICIÊNCIA FÍSICA NOS MEMBROS SUPERIORES, Revista Brasileira de Inovação Tecnológica em Saúde, 2017.

FERNANDES, FLÁVIA GONÇALVES - Serious Game para Auxílio na Realização de Exercícios Fisioterapêuticos utilizando Kinect e Realidade Virtual, SBGAMES, 2015 -
<http://www.sbgames.org/sbgames2015/anaispdf/computacao-short/147794.pdf>

MARQUES, ISABELA ALVES - Jogo sério e realidade virtual na reabilitação do AVC crônico: protocolo individualizado, UNIVERSIDADE FEDERAL DE UBERLÂNDIA FACULDADE DE ENGENHARIA ELÉTRICA, 2019
<https://repositorio.ufu.br/bitstream/123456789/27400/1/Jogo%20s%C3%A9rio%20e%20realidade%20virtual.pdf>

MEIRELES, PEDRO SOUSA - Exploring Direct User-Interaction Techniques in a Virtual Reality Game for People with Hand Impairments, SBGAMES, 2020 -
<https://www.sbgames.org/proceedings2020/JogosSaudeFull/209323.pdf>

Relatório semanal de desenvolvimento

Este documento contém o progresso do Projeto Final 2 do aluno Antônio Vasconcellos Chaves semana a semana. Onde tudo o que for implementado semana a semana vai ser documentado aqui, bem como o planejamento para a semana seguinte à atual.

Links importantes:

[Github do repositório Unity](#)

[Github do repositório Myo](#)

01/04 - 17/05

Pesquisa direcionada às bibliotecas para comunicação entre o Myo e o Unity. Próximos passos, primeira experiência com Realidade Virtual e Set Up da aplicação.

18/05 - 25/05

Estudo direcionado às diretrizes da Realidade Virtual com o professor Renato Cherullo. Primeira experiência com RV. Testes funcionais do projeto deixado por Pedro. Próximos passos, adição do Myo ao Unity por meio das bibliotecas encontradas: myo-unity (Thalmic Labs) e MyoSharp (Tayfun Uzun).

26/05 - 07/06

Tentativas de instalação das bibliotecas Unity com o Myo. Não solucionado. Próximos passos, pesquisa mais profunda sobre a aplicação em NodeJS, suas vantagens e desvantagens.

08/06 - 25/06

Afastado por conta da saúde.

26/06 - 15/07

Instauração do servidor em NodeJS e comunicação entre aplicação e server. Testes funcionais ligados ao funcionamento do Myo, que geraram a calibragem via código dos valores recebidos para obter uma melhor experiência.