

Universidade Federal Da Paraíba
Centro De Informática
Programa De Pós-Graduação Em Computação, Comunicação E Artes

FRANCISCO BRENO MOURA ALVES

**LUMU: DESENVOLVIMENTO DE UM INSTRUMENTO MUSICAL
CONTROLADOR SEM FIOS COMO RECURSO AUXILIAR EM SALA DE
AULA PARA PROFESSORES DE MÚSICA**

João Pessoa

2022

Francisco Breno Moura Alves

LuMU: Desenvolvimento de um instrumento musical controlador sem fios como recurso auxiliar em sala de aula para professores de música

Dissertação apresentada ao Programa de Pós-Graduação em Computação, Comunicação e Artes (PPGCCA) da Universidade Federal da Paraíba, como requisito parcial para a obtenção do título de Mestre em Computação, Comunicação e Artes, na linha de pesquisa Mídias em Ambientes Digitais.

Orientador: Prof. Dr. Carlos Eduardo Coelho Freire Batista

João Pessoa

2022

Catálogo na publicação
Seção de Catalogação e Classificação

A4741 Alves, Francisco Breno Moura.

LuMU : desenvolvimento de um instrumento musical controlador sem fios como recurso auxiliar em sala de aula para professores de música / Francisco Breno Moura Alves. - João Pessoa, 2022.

98 f. : il.

Orientação: Carlos Eduardo Coelho Freire Batista.
Dissertação (Mestrado) - UFPB/CI.

1. Luvas digitais. 2. Instrumento controlador. 3. Educação musical. 4. Gadget vestível. 5. Internet das coisas. I. Batista, Carlos Eduardo Coelho Freire. II. Título.

UFPB/BC

CDU 004.358(043)



Aos vinte e cinco dias do mês de agosto do ano de dois mil e vinte e dois, às 09h00min, por meio de videoconferência, instalou-se a banca examinadora de dissertação de Mestrado do aluno FRANCISCO BRENO MOURA ALVES. A banca examinadora foi composta pelos professores Dr. IÁLIS CAVALCANTE DE PAULA JÚNIOR, UFC, examinador externo à instituição, Dra. THAIS GAUDENCIO DO REGO, UFPB, examinadora externa ao programa, Dr. ALEXANDRE MAGNO E SILVA FERREIRA, UFPB, examinador interno, Dr. CARLOS EDUARDO COELHO FREIRE BATISTA, UFPB, presidente. Deu-se início a abertura dos trabalhos, por parte do professor Dr. CARLOS EDUARDO COELHO FREIRE BATISTA, coordenador do Programa, que, após apresentar os membros da banca examinadora e esclarecer a tramitação da defesa, iniciou a presidência dos trabalhos, de imediato solicitando ao candidato que iniciasse a apresentação da dissertação, intitulada LUMU: DESENVOLVIMENTO DE UM INSTRUMENTO MUSICAL CONTROLADOR SEM FIOS COMO RECURSO AUXILIAR EM SALA DE AULA PARA PROFESSORES DE MÚSICA. Concluída a exposição, o professor Dr. CARLOS EDUARDO COELHO FREIRE BATISTA, presidente, passou a palavra ao professor Dr. IÁLIS CAVALCANTE DE PAULA JÚNIOR, para arguir o candidato, e, em seguida, à professora Dra. THAIS GAUDENCIO DO REGO, e, em seguida, ao professor Dr. ALEXANDRE MAGNO E SILVA FERREIRA, para que fizessem o mesmo; após o que fez suas considerações sobre o trabalho em julgamento; tendo sido aprovado o candidato, conforme as normas vigentes na Universidade Federal da Paraíba.

Documento assinado digitalmente



IÁLIS CAVALCANTE DE PAULA JUNIOR
Data: 24/10/2022 14:37:42-0300
Verifique em <https://verificador.iti.br>

Dr. IÁLIS CAVALCANTE DE PAULA JÚNIOR, UFC

Examinador Externo à Instituição

Thais Gaudencio do Rego

Dra. THAIS GAUDENCIO DO REGO, UFPB

Examinador Externo ao Programa

Documento assinado digitalmente



ALEXANDRE MAGNO E SILVA FERREIRA
Data: 20/10/2022 15:44:58-0300
Verifique em <https://verificador.iti.br>

Dr. ALEXANDRE MAGNO E SILVA FERREIRA, UFPB

Examinador Interno



Dr. CARLOS EDUARDO COELHO FREIRE BATISTA, UFPB

FRANCISCO BRENO MOURA ALVES

FRANCISCO BRENO MOURA ALVES

Mestrando

Dedico este trabalho, a Francisco e Maria, meus pais, não apenas por serem os meus maiores apoiadores, mas por serem a minha motivação de seguir em frente.

AGRADECIMENTOS

Agradeço em primeiro lugar a Deus e aos espíritos de luz que me acompanham, por me manterem em pé, mesmo quando desacreditei de minhas forças, e por Ele ser a base de todas as minhas conquistas.

Aos meus pais Francisco José de Moura e Maria do Socorro Moura Alves e meu irmão João Phelipe Moura Alves por acreditarem nas minhas escolhas, mesmo quando nem mesmo eu acreditei.

A minha noiva Laídia Evangelista por ser o meu pilar de sustentação, nos bons e maus tempos.

Ao Professor Dr. Carlos Eduardo Coelho Freire Batista por me transferir conhecimentos essenciais ao desenvolvimento deste projeto.

Ao meu tio “Louro”, a minha tia Gorete e minha avó Teresinha, por incontáveis momentos de ajuda, desde a época da graduação. Sem eles, eu sequer teria obtido o título de bacharel primeiramente.

Ao meu primo Mirtiel Frankson por suas correções e apontamentos muito valiosos neste trabalho.

Aos mestres Eládio Junior e Ronnielly Diniz e a equipe de Taekwondo Parahyba Fighters por serem a minha segunda família em João Pessoa, sem vocês essa conquista também não seria possível.

RESUMO

O LuMU é um dispositivo controlador sem fios de baixo custo, vestível, no formato de luvas, com desenvolvimento orientado a intercomunicação entre dispositivos, utilizando conceitos de internet das coisas. Mais que um equipamento isolado, é uma suíte multimedia, composta pela integração de *hardware* projetado e *softwares* de código aberto. O LuMU busca simular a atuação de um teclado controlador, com mais mobilidade e liberdade de movimentos. Seu funcionamento se dá de maneira intuitiva ao tocar o polegar em zonas específicas em outros dedos da mão, acionando gatilhos que permitam controlar notas musicais específicas, acionar *samples* (áudio e vídeo) e/ou modular sonoridades. Ele foi projetado ser mais uma ferramenta auxiliar para professores de música em ambiente de sala de aula em suas mais diversas subáreas de conhecimento. Neste trabalho realizamos o desenvolvimento de um protótipo funcional, capaz de intercomunicação entre o dispositivo e PC hospedeiro, utilizando os protocolos de comunicação OSC-over-UDP, documentação de material multimídia a ser utilizado com o projeto e a execução de testes com usuários.

Palavras-chave: Instrumento controlador. Música. Gadget vestível. Educação musical.

ABSTRACT

The LuMU is a low-cost, wearable wireless controller device, in the form of gloves, with development oriented to intercommunication between devices, using Internet of Things concepts. More than an isolated device, it is a multimedia suite, composed by the integration of designed hardware and open source software. LuMU seeks to simulate the performance of a keyboard controller, with more mobility and freedom of movement. It works intuitively by touching the thumb to specific zones in other fingers of the hand, activating triggers that allow you to control specific musical notes, trigger samples (audio and video) and/or modulate sounds. It was designed to be another auxiliary tool for music teachers in the classroom environment in its various sub-areas of knowledge. In this work we developed a functional prototype, capable of intercommunication between the device and the host PC, using the OSC-over-UDP communication protocols, documentation of multimedia material to be used with the project, and the execution of tests with users.

Keywords: Music controller. Wearable gadget. Music education.

LISTA DE FIGURAS

Figura 1 - Porcentagem de professores do ensino médio que participaram de atividades de desenvolvimento profissional	14
Figura 2 - Conceitualização de funcionamento do projeto	15
Figura 3 - Cenário de utilização de IoT	26
Figura 4 - WEMOS LOLIN D32 (Vista frontal e posterior)	28
Figura 5 - Interface Pure Data	31
Figura 6 - Diagrama para mão direita	35
Figura 7 - Diagrama para mão esquerda	35
Figura 8 - Montagem do protótipo	36
Figura 9 - Costura da linha condutiva	37
Figura 10 - Conexão das portas do dispositivo	37
Figura 11 - Zona simulada por <i>Patch</i> Pure Data	39
Figura 12 - <i>Patch sender</i>	39
Figura 13 - <i>Patch receiver</i>	41
Figura 14 - Diagrama de funcionamento do LuMU e <i>Applets</i>	42
Figura 15 - Primeiro encontro avaliativo	44
Figura 16 - Respostas da pergunta 7.....	50

LISTA DE TABELAS

Tabela 1 - Comparativo entre trabalhos	22
Tabela 2 - Listagem e custo de insumos	35
Tabela 3 - <i>Softwares</i> testados com o LuMU	42
Tabela 4 - <i>Samples</i> para utilização com o LuMU	43

SIGLAS E ABREVIATURAS

DAW	<i>Digital Audio Workstation</i> (Estação de Trabalho de Áudio Digital)
IEEE	<i>Institute of Electrical and Electronic Engineers</i> (Instituto de Engenheiros Eletricistas e Eletrônicos)
IHC	Interação humano-computador
IoT	<i>Internet Of Things</i> (Internet das Coisas)
LAN	<i>Local Area Network</i> (Rede Local)
MIDI	<i>Musical Instrument Digital Interface</i> (Interface Digital de Instrumento Musical)
OSC	<i>Open Sound Control</i>
PD	Pure Data
SoC	<i>System-on-a-chip</i> (Sistema em um chip)
TCP	<i>Transmission Control Protocol</i> (Protocolo de Controle de Transmissão)
UDP	<i>User Datagram Protocol</i> (Protocolo de Datagrama do Usuário)
VST	<i>Virtual Studio Technology</i> (Tecnologia de Studio Virtual)

SUMÁRIO

1. INTRODUÇÃO	10
1.1. Justificativa e problemática inicial	13
1.2. Objetivos gerais e específicos	15
1.2.1. Objetivos de desenvolvimento físico	16
1.2.2. Objetivos de desenvolvimento virtual	16
2. ESTADO DA ARTE	18
2.1. Referências performáticas	18
2.2. Referências educacionais	20
2.3. Referências audiovisuais	22
3. REVISÃO TEÓRICA	24
3.1. Internet das Coisas	24
3.2. ESP32 E SoC	26
3.3. Arduino IDE	27
3.4. Baterias de LiFePo4	28
3.5. Linha de costura condutora	28
3.6. Módulo GY-91	29
3.7. Pure Data	29
3.8. <i>Open Sound Control</i>	30
3.9. <i>Musical Instrument Digital Interface</i>	31
3.10. <i>Digital Audio Workstation</i>	31
3.11. Processing	32
4. METODOLOGIA	33
4.1. Primeira fase - ideia inicial, modelagem do protótipo e aquisição dos insumos	34
4.2. Segunda fase - montagem do protótipo	36
4.2.1. Fixação dos componentes	36
4.2.2. Conexão dos componentes	36
4.3. Terceira fase - desenvolvimento dos <i>softwares</i> e configuração no DAW	38
4.3.1. Desenvolvimento do <i>sketch</i> - Arduino IDE (código embarcado)	38
4.3.2. Desenvolvimento dos <i>patches</i> (Pure Data)	38
4.3.2.1. <i>Sender</i>	38
4.3.2.2. <i>Receiver</i>	40
4.3.3. Desenvolvimento do <i>sketch</i> – Processing	41

4.3.4. Compêndio de <i>software</i> e <i>samples</i>	42
5. TESTES COM USUÁRIOS	44
5.1. Sessões de teste	44
5.1.1. Configuração do dispositivo	45
5.1.2. Demonstração prática de utilização do dispositivo	45
5.1.3. Aplicação de questionário	46
6. RESULTADOS E ANÁLISES	47
6.1. Resultados relacionados ao <i>software</i>	47
6.2. Resultados relacionados ao <i>hardware</i>	47
6.3. Resultado do questionário	48
7. CONCLUSÃO	52
8. REFERÊNCIAS	55
APÊNDICES	60

1 - INTRODUÇÃO

A segunda revolução industrial (1850 - 1939), com o advento dos dispositivos elétricos e eletrônicos, mudou e moldou o mundo como o conhecemos, assim, o desenvolvimento de materiais semicondutores permitiu um grande salto no aprimoramento dos recursos computacionais, esse avanço que claramente influenciou a forma como produzimos e ouvimos música.

Historicamente, o surgimento dos primeiros dispositivos eletrônicos ocorreu simultaneamente com avanços na tecnologia musical, esse avanço aos dispositivos controladores que temos atualmente no mercado iniciou-se em 1897 com a criação do espaçoso *Telharmonium*, “medindo 18 metros de largura e pesando 200 toneladas”. (FRITSCH, 2008, p. 25), considerado o primeiro instrumento eletrônico.

A criação do transistor nos anos 40 do século 20, viabilizou a redução de tamanho e consumo de potência dos dispositivos eletrônicos, possibilitando também o barateamento no desenvolvimento e comercialização de equipamentos musicais. A tecnologia digital e o uso de *samples* de sons, abriram um leque de novas sonoridades nos instrumentos eletrônicos, havendo uma brusca expansão na produção de instrumentos sintetizadores.

Sobre o conceito de sintetizadores:

Equipamento capaz de gerar sons musicais por processos eletrônicos, com grande controle sobre os parâmetros do som (altura, timbre e intensidade). A maioria dos sintetizadores atuais utiliza o teclado como meio de controle, mas existem muitos módulos sintetizadores sem teclado, controlados via MIDI. (RATTON, 2009, p. 164).

Com crescente número de sintetizadores no mercado, músicos, artistas, produtores musicais, compositores, arranjadores e demais profissionais artístico-musicais passaram a ter uma maior possibilidade de “construir seus sons”, modificando o mercado fonográfico para o que conhecemos hoje. Contudo, nem todos esses profissionais possuem propensão à execução de instrumentos de teclas, alguns performam instrumentos de cordas, instrumentos percussivos, entre outros tipos de instrumentos, ou sequer possuem prática alguma em um instrumento musical.

Esse é um dos intuitos dos dispositivos controladores, poder manipular, por meio de um protocolo de comunicação definido, sintetizadores, *samplers* e outros dispositivos

ou instrumentos eletrônicos musicais, cênicos, ou não, habilitados com padrão MIDI (*Musical Instrument Digital Interface*), OSC (*Open Sound Control*) ou semelhante.

Em anuência com o que expõe DE ALMEIDA (2018, p. 21) “Com a crescente expansão dos dispositivos computacionais na composição musical, muitas aplicações foram desenvolvidas”, esse texto aborda o uso de tecnologias já amplamente difundidas em associação de abordagens modernas, disruptivas, na construção de um aparato auxiliar como ferramenta didática-musical.

Sobre tecnologias disruptivas CHRISTENSEN *et al.* (2012, *apud* NOBREGA e MAIA, 2020) afirmam que “A palavra disruptiva se refere a um processo que engloba uma interrupção. Disruptivo, então, é tudo o que muda o curso das coisas, tudo o que tem a qualidade de ser um “divisor de águas””. DEMO (2012) completa: “A inovação é bem-vinda, mas serve para ‘sustentar’ o que já funciona.”

A maioria dos sistemas desenvolvidos a fim de facilitar o ensino de música se baseia em aparatos orientados a *software*. Tal aspecto, por natureza, por estar contido apenas no “espaço virtual”, ausentando-se da camada física da realidade, ignora um aspecto intrínseco do fazer musical, a corporalidade. Sobre educação e corporalidade podemos afirmar que:

A presença do homem no mundo se dá através do corpo. É a partir de sua existência corpórea que o ser humano percebe e é percebido, que age e interage com a realidade existencial. Quando entendemos essa condição de corpo no mundo podemos perceber que somos sujeitos encarnados e, portanto, incorporamos aquilo que vivemos. O ato de educar considerado aqui, refere-se dessa feita, ao espaço da convivência, nas relações estabelecidas entre seres humanos. (SILVA, 2008, p. 19, *apud* ALVES, 2006)

Ao fazer uso de ferramentas contidas apenas no espaço virtual, perde-se a capacidade de aproveitamento desse aspecto fundamental a educação musical. No sentido dessas e outras limitações impostas por essas ferramentas, o trabalho aqui apresentado visa descrever o desenvolvimento de um sistema capaz de facilitar a execução de sonoridades alinhadas ao aspecto corporal do educador em sintonia com seus alunos, consolidando uma verdadeira experiência formadora. JOSSO (2004) relata o processo de uma experiência formadora:

Começamos a perceber que o que faz a experiência formadora é uma aprendizagem que articula, hierarquicamente: saber-fazer e conhecimentos, funcionalidade e significação, técnicas e valores num espaço-tempo que oferece a cada um a oportunidade de uma presença para si e para a situação, por meio da mobilização de uma pluralidade de registros (JOSSO, 2004, p. 39).

Em maior detalhe apresentamos a concepção de um sistema criado por dois aspectos, o físico, através de aparato vestível, em formato de luvas, que simula a execução de um instrumento de teclas, e o virtual, através da criação de aplicativos, configuração de *software* livre e coleta de *samples* específicos. Tal conjunto de soluções aqui preconizado e que foi nomeado de LuMU (acônimo para Luvas Musicais), propõe a possibilidade de criação/produção musical baseada no conceito da performatividade,

Sobre performatividade musical é exposto:

A performance, portanto, presentifica (contém) uma – não a – interpretação, caracterizando-se, dessa forma, pela materialização de uma dentre várias boas possibilidades interpretativas que se criam a partir do estudo do repertório. Soma-se ainda, como material a ser incorporado nessa presentificação, os imprevistos aos quais qualquer performance está sujeita. Nesse sentido, a performatividade, conceito definido por autores desde J. Austin, a partir dos Atos da Fala, a Hans-Thies Lehmann no teatro contemporâneo, poderia ampliar o olhar que temos sobre a performance musical, compreendendo-a como “uma experiência do real que visa ser imediata”, como afirmará Lehmann; ou ainda, como dirá Cohen, um “instante onde o como interessa mais do que o que”. (MENEZES e KAYAMA, 2019)

O LuMU tem por pretensão oferecer ao professor de música mais liberdade e flexibilidade na execução de sons, sejam eles sintetizados ou sampleados, melódicos, percussivos ou harmônicos e/o imagens. Assim como ocasionar uma experiência educacional mais completa.

Este trabalho tem como eixos principais o processo de prototipação do *hardware* utilizado, o desenvolvimento de *software* pelo autor, através da codificação de *applets* de código aberto, a realização um compêndio de ferramentas de áudio de livre uso, como

softwares DAW¹, *plug-ins* VST² e VSTi² e uso de *samples*, em conjunto com os dois pontos descritos anteriormente, o processo de testes com usuários, a fim de documentação de processo de *feedback* para o desenvolvimento futuro do projeto.

Os processos de desenvolvimentos aqui preconizados nesta dissertação, encontram-se essencialmente firmados sobre as prerrogativas do *Design Science Research*, O *Design Science Research* (DSR) é um método orientado pela solução de problemas que se utiliza da Design Science como base epistemológica. Esta visa construir e avaliar de forma fundamentada e operacionalizada, a partir do entendimento do problema, artefatos que permitam mudar situações alterando-as para estados melhores ou desejáveis. Pode-se afirmar que a DSR tem como característica o fato de aproximar o conhecimento da prática em projetos que se caracterizam por produzir conhecimento científico enquanto produzem artefatos úteis (DRESCH *et al*, 2015; WIERINGA, 2009, apud BRANDÃO, 2020).

Este trabalho está dividido em seções de natureza descritiva, cada seção conta com a elaboração de conceitos teóricos ou práticos utilizados neste estudo. Na seção 2 Estado da Arte, será descrito uma série de trabalhos que se assemelham e/ou exerceram alguma influência nesta pesquisa. Na seção 3 Revisão Teórica é minuciado os conceitos técnicos pertinentes e necessários à produção deste trabalho. Na seção 4 Metodologia, ocorre a descrição dos métodos e processos utilizados na concepção do artefato. Na seção 5 Testes com Usuários há a descrição do processo de aquisição de *feedback* por parte de usuários. Na seção 6 Resultados e Análise ocorrem considerações ao processo aqui descrito como um todo.

1.1 - Justificativa e problemática inicial

No âmbito da realização da dissertação, a escolha desse tema foi advinda de uma interseção entre vida pessoal e vida acadêmica, sendo um caminho óbvio devido a minha paixão à musicalidade em suas diversas formas e expressões e ao meu percurso profissional como engenheiro de computação, horto com minha predisposição à solução de problemas e processos criativos, provavelmente proveniente dos aspectos da cultura *maker* a qual fui inserido em boa parte em minha graduação.

¹*Digital Audio Workstation* são suítes de edição de áudio multipista, também utilizados para performance ao vivo. Ver seção 3.10.

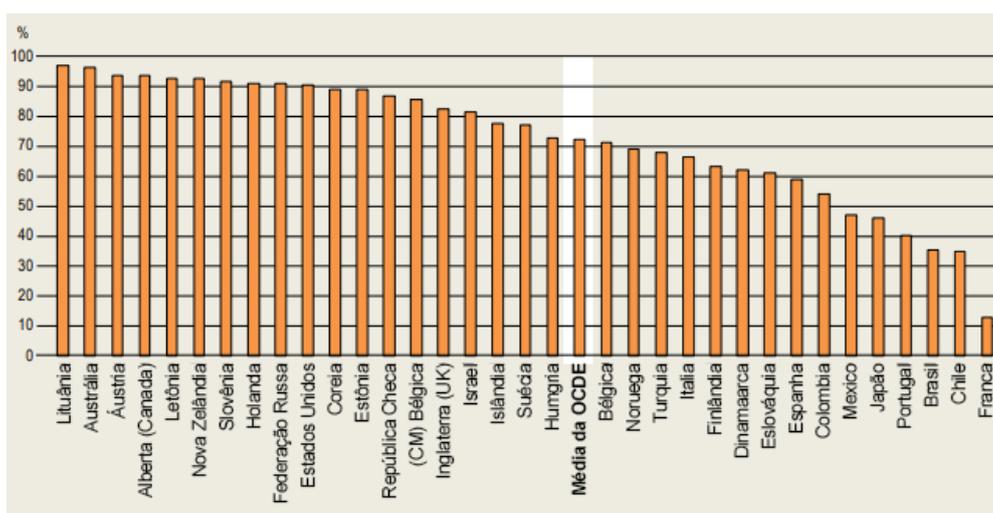
²*Plug-ins VST (Virtual Studio Technology)* e *VSTi (Virtual Studio Technology Instruments)* são interfaces em *software* desenvolvidas para originalmente para emulação de sistemas reais.

Abordar o desenvolvimento de um dispositivo (*hardware + software*) que incorpore distintas áreas tecnológicas, e tenha como alicerce a interdisciplinaridade, mostra-se como uma ferramenta adequada na consumação prática, não apenas dos inúmeros conhecimentos abordados no curso de mestrado do Programa de Pós-Graduação em Computação, Comunicação e artes da Universidade Federal da Paraíba, mas também como um objeto adequado às salas de aula do Brasil com seus inúmeros aspectos limitativos, financeiros inclusos.

Assim, esse tema manifesta-se também com o propósito primário de inserir mais uma possibilidade aos professores de música, alinhado com o propósito decorrente do fomento músico-cultural nacional.

A Lei nº 9.394/2008 tornou obrigatório o ensino de música no ensino fundamental e médio (BRASIL, 2008), contudo, segundo o estudo *Education at a Glance* (OCDE, 2020), o Brasil ainda investe menos por aluno que a média dos países integrantes da Organização para Cooperação e Desenvolvimento Econômico (OCDE). Outro dado relevante a ser apresentado e que incentiva a produção deste trabalho é apresentado na Figura 1, que mostra a porcentagem de professores do ensino médio que participaram de atividades de desenvolvimento profissional, com o Brasil ficando colocado nas 3 últimas posições, espelho do insuficiente investimento na área, como resultado a deterioração da condição dos profissionais de educação.

Figura 1 – Porcentagem de professores do ensino médio que participaram de atividades de desenvolvimento profissional



Fonte: *Education at a Glance* – OCDE, 2020

Esses dados apenas destacam que soluções de baixo custo, que possibilitem a adição de novas ferramentas didáticas aos profissionais de educação, se fazem necessárias

no atual cenário pedagógico brasileiro. Ainda nesse contexto é possível, antes de quaisquer observações, inferir uma carência em termos de infraestrutura disponível e diretamente aplicável a educação musical nas escolas brasileiras.

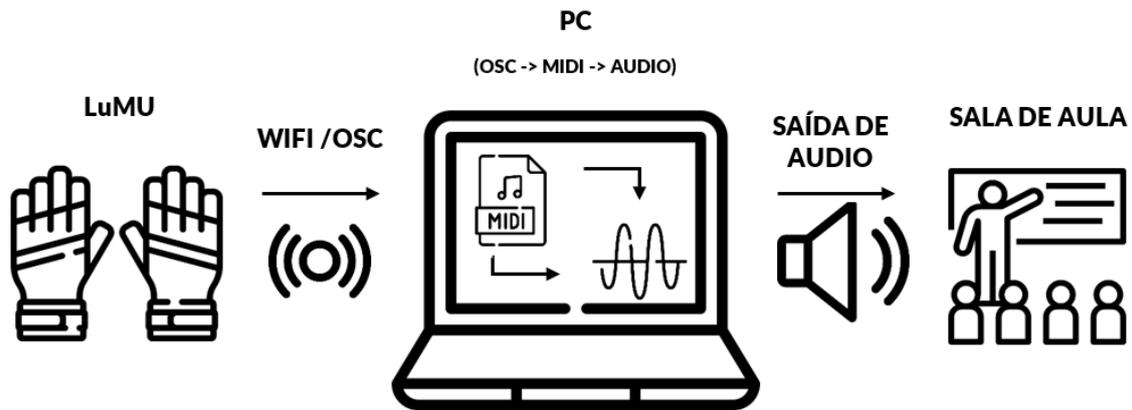
Essa ausência de recursos aos profissionais da educação teve forte influência como papel motivador desse trabalho. Outro forte motivador é a possibilidade de inserção de um processo educacional mais dinâmico, aliados aos recursos multimídia a serem descritos.

1.2 - Objetivos gerais e específicos

Este trabalho tem como objetivo geral o desenvolvimento de um dispositivo auxiliar em sala da aula para professores de música, conceitualizado como um instrumento musical controlador, e sua suíte de *softwares* auxiliares, que funcione sem a necessidade de ligação cabeada com o dispositivo controlado (*wireless*), de baixo custo viabilizado pelo emprego de tecnologias IoT, alinhando tendências *maker* e que utilize conceitos de tecnologias vestíveis, mais especificamente no formato de luvas.

Os objetivos específicos são dispostos em duas categorias, objetivos de desenvolvimento físico e objetivos de desenvolvimento de *software*. Os objetivos de desenvolvimento físico discorrem da conceitualização, prototipagem e execução tangível de *hardware* a serem demonstrados e são detalhados na seção 1.2.1. Os objetivos de desenvolvimento de *software* decorrem da conceitualização de produção e codificação de *software* autoral, do processo de investigação de uso de *software* já existente, que esteja alinhado com o propósito educacional do projeto, assim como na coleta e/ou produção de amostras de áudio, *samples* e são detalhados na seção 1.2.2. Esse alinhamento entre *hardware* e *software* pode ser observado na Figura 2, *hardware* representado pelo dispositivo LuMU e *software* nas ferramentas auxiliares executadas no PC.

Figura 2 – Conceitualização de funcionamento do projeto



Fonte: O autor³.

1.2.1 - Objetivos de desenvolvimento físico

- Testar a manipulação fios de liga aço-algodão⁴ em material têxtil pré-fabricado;
- Implementar circuitos integrados em material têxtil pré-fabricado;
- Desenvolver uma interface HCI⁵ para controle de parâmetros de som, como intensidade, duração, altura⁶ e timbre, e manipulação de efeitos no dispositivo;
- Testar a utilização de placas de desenvolvimento ESP32⁷ em protótipos com finalidades educacionais;

1.2.2 - Objetivos de desenvolvimento de *software*

- Desenvolver interface com o protocolo de comunicação em rede OSC⁸ para transmissão via IEEE 802.11 (Wi-Fi) usando protocolo UDP⁹;

³ Ícones produzidos por Freepik, Vitaly Gorbachevk, Pixel perfect e geotatah. Disponível em: www.flaticon.com.

⁴ Fio condutível de baixa resistência. Ver seção 3.5

⁵ *Human-computer interaction*, interação Humano-Computador, área de estudos da interação, suas interfaces e efeitos entre pessoas e computadores.

⁶ “Altura ou *pitch* é a sensação subjetiva que cada indivíduo possui em relação à frequência de um determinado tom. E é o que determina se um som é alto ou baixo, ou seja, um som alto seria um som agudo e um som baixo um som grave.” (CARNEIRO, 2003)

⁷ É o nome dado a uma família de placas de desenvolvimento de protótipos baseadas no módulo ESP32-S3-WROOM-1 da Expressif Systems. Ver seção 3.2.

⁸ *Open Sound Control* é um protocolo de comunicação comumente utilizado entre dispositivos multimídia em rede. Ver seção 3.8.

⁹ *User Datagram Protocol* é um protocolo de comunicação simplificado da camada de transporte (Modelo OSI), que diferentemente do protocolo TCP, não emprega controle de integridade na recepção dos pacotes.

- Desenvolver interface com o protocolo de comunicação OSC para interconexão com o protocolo MIDI¹⁰;
- Desenvolver aplicação em Processing¹¹ para controle de imagem e vídeo através do LuMU;
- Desenvolver *patch* Pure Data¹² para envio de mensagens OSC em LAN;
- Testar a recepção de mensagens OSC enviadas por plataforma microcontrolada para PC;
- Avaliar o grau de usabilidade do LuMU e seus *softwares* auxiliares através do processo de testes com usuários.

¹⁰ *MUSICAL INSTRUMENT DIGITAL INTERFACE* é uma interface protocolar de intercomunicação física Ver seção 3.9.

¹¹ Processing é uma linguagem de programação desenvolvida por Casey Reas e Bem Fry e tem como um dos objetivos atuar como ferramenta de produção visual. Ver seção 3.11.

¹² Pure Data (PD) é uma VPL (*Visual Programming Language*) desenvolvida para aplicações multimídia e criando interações entre homem, computador e música. Ver seção 3.7.

2 - ESTADO DA ARTE

A construção de dispositivos voltados à área de *interactive music*¹³ utilizando plataformas de fácil prototipagem (Arduino, NodeMCU/ESP, Teensy) é um assunto frequentemente abordado em trabalhos que visam conectar o movimento *maker* às práticas musicológicas, através da integração de *hardware*, *software* e música. O movimento *maker* refere-se amplamente ao crescente número de pessoas envolvidas na produção criativa de dispositivos no seu dia a dia e que se encontram em fóruns físicos e digitais para compartilhar seus processos e produtos com outras pessoas. (HALVERSON e SHERIDAN, 2014, p. 495, *tradução nossa*).

2.1 - Referências performáticas

Nesse sentido, nesta seção apresentaremos trabalhos e dispositivos que utilizem os conceitos *maker*, e/ou IoT e/ou *Wearable Technology* na modelagem de instrumentos controladores que de alguma forma se alinham ou influenciam ao projeto LuMU.

TSUNG-CHING, SHU-HUI e CHE-YI (2011) demonstraram a modificação de 4 Teremins (*Ætherophone*) para criação de um novo dispositivo, utilizado no padrão de execução não-tátil do Teremim. Um novo instrumento controlador foi desenvolvido, modelado com a plataforma Arduino, o dispositivo se conecta ao computador através do protocolo MIDI, permitindo ao *performer* controlar não apenas alterações rítmicas ou de altura do som, como também outras características do som, como o *envelope*¹⁴, ou efeitos secundários como eco e reverberação. Em nosso trabalho a “manipulação” de som se dá pela alteração na diferença de potencial, quando há contato dos polegares a zonas nos outros dedos da mão, enquanto no trabalho de TSUNG-CHING, SHU-HUI e CHE-YI (2011) se dá pela alteração da distância entre as antenas captadores do Teremim, utilizando a capacitância intrínseca do corpo humano.

A artista e engenheira de áudio, Dra. Imogen Jennifer Heap lançou uma campanha de financiamento coletivo na plataforma *Kickstarter*¹⁵ para um projeto de luvas para controle de sons. A campanha arrecadou apenas U\$ 182.547 ao invés dos U\$ 274.970 pleiteados, contudo, após 5 anos, em 2019, Heap lançou comercialmente o projeto,

¹³ *Interactive Music* também chamada de *Adaptative Music* ou *Dynamic Music* é uma área da musicologia que aborda a alteração de parâmetros musicais como volume, ritmo ou pitch por interações como movimentação corpórea e/ou por instrumentos e dispositivos não convencionais.

¹⁴ Correspondente a característica de como a amplitude do som se comporta em relação ao tempo percorrido.

¹⁵ Disponível em: <<https://www.kickstarter.com/projects/mimu/mimu-glove-for-music>>

nomeado de Mi.Mu. O dispositivo de Heap não consta na literatura, e é apenas encontrado disponível para aquisição comercial, contudo devido a relevância para o nosso projeto ele será descrito a seguir.

O projeto de Heap, assim como nosso dispositivo, possui a capacidade de comunicação sem fio OSC-*over*-WiFi e controle de gatilhos MIDI, contudo, o Mi.Mu é focado na execução de gestos combinados na posição dos dedos com a movimentação completa da mão e braço captados por um sensor 9-Eixos presente nas duas mãos. Em nosso projeto, a execução se dá pelo contato do polegar a zonas localizadas nos dedos, e o sensor 9-Eixos está presente apenas na mão esquerda para modulação de características do som e não para execução de notas ou *samples*. Também há diferenças no *design* das luvas utilizadas, as luvas de Heap possuem corte aberto nos dedos, enquanto as LuMU possuem padrão fechado.

Apesar de serem dispositivos voltados à área de atuação semelhantes, o dispositivo de Heap se distingue do nosso devido ao viés artístico e gestual em sua execução, enquanto nosso trabalho é voltado para uma performance instrumental mais precisa, se aproximando da execução de um instrumento de teclas.

SARANG *et al.* (2015) criou um dispositivo de execução musical de caráter percussivo. Ele utiliza o SDK do Xbox Kinect fornecido pela Microsoft em conjunto com a plataforma Arduino para realizar leitura de movimentos do corpo humano, associando esses movimentos a *samples* de bateria e percussão em geral. O resultado é uma performatividade percussiva corporal, como os autores denominam, de *air drums*.

Esse trabalho conta com um propósito geral mais simplificado, contudo, o seu desenvolvimento aborda conceitos que nosso projeto também explora, principalmente a capacidade da atribuição de *samples* a ações específicas através de ações corporais.

Algumas distinções entre nosso trabalho e de SARANG *et al.* (2015) são notáveis, uma vez que no dele a execução musical se limita a elementos de bateria e percussão. Essa percussividade pode ser atribuída a forma de execução musical corporal do dispositivo, ao contrário do nosso dispositivo que pode executar, além de elementos melódicos, elementos harmônicos. Por fim, existe a possibilidade de controlar características dos sons executados em tempo real.

Outro trabalho disponível na literatura é o de KLIPFEL (2017), *MIDI Motion Gloves*, em seu artigo que trata não da criação de um dispositivo focado na execução musical em si, mas de um aparato controlador MIDI para o controle de interfaces DAW, para facilitar e dinamizar o processo de composição e produção musical. KLIPFEL (2017)

focou em facilitar o acesso a ferramentas utilizadas comumente na manipulação de som, edição, *layering*¹⁶ de arranjos.

Assim como nosso trabalho, KLIPFEL (2017) optou por utilizar o *design* de luvas no desenvolvimento do dispositivo. Outra semelhança é no acionamento dos gatilhos MIDI, que se dão pelo pressionamento do polegar aos demais dedos, porém esse movimento força o acionamento um resistor sensível à força (FSR) localizado na ponta dos dedos. No nosso dispositivo, o acionamento é realizado pelo contato entre uma porta digital ao GND, através do acionamento de porta do tipo *INPUT_PULLUP* na função *pinMode()* no Arduino IDE. O dispositivo de KLIPFEL (2017) também conta com um display LCD para exibição de informações relacionadas ao uso com o DAW. Embora o dispositivo de KLIPFEL (2017) também consiga executar notas musicais (em diferentes alturas diferentes), as *MIDI Motion Gloves* possuem uma limitação de execução de escalas diatônicas¹⁷, enquanto as LuMU são cromáticas.

2.2 - Referências educacionais

Ao consultar o indexador de repositórios acadêmicos Google Scholar, é possível encontrar diversos trabalhos que engajem a utilização de Pure Data e práticas pedagógicas musicais. Contudo, os resultados que utilizem OSC e Pure Data alinhadas a um projeto controlador físico como recurso pedagógico sonoro para sala de aula são escassos. Os artigos alinhados com o projeto aqui descrito estão listados nos próximos parágrafos desta seção.

LUNHANI e MANNIS (2010), no artigo Dispositivos Físicos Interativos Aplicados À Projeção Sonora Na Música Eletroacústica, apresentam o desenvolvimento do *Reactable*, um dispositivo audiovisual interativo desenvolvido com a linguagem de programação Processing, um *patch* auxiliar Pure Data e comunicação através do protocolo OSC. LUNHANI e MANIS (2010, p. 1) descrevem que manipulando objetos sobre uma superfície sensível o usuário pode controlar a posição e o deslocamento de fontes sonoras durante a performance. Com a presença da superfície sensível que está sendo simulada através do aplicativo Processing, empregando o protocolo OSC.

¹⁶ *Layering* é quando se combinam múltiplos sons individuais para criar um som único e mais completo (HYPERBITS, 2022, tradução nossa).

¹⁷ A escala cromática possui 12 sonoridades, enquanto a escala diatônica tradicional possui apenas 7 sonoridades.

COTRIM (2020) descreve, na tese de doutorado *Práticas Pedagógicas Criativas Musicais em Ambiente de Estúdio Eletroacústico: Experiência e Polissonia em Sala de Aula*, a empregabilidade de programas de áudio gratuitos, como Pure Data, em conjunção com o sistema operacional, também gratuito, GNU/Linux, como parte integrante de um ambiente de estúdio eletroacústico. COTRIM (2020, p. 14) ainda descreve como a presença de um ambiente de estúdio eletroacústico em sala de aula introduz a reflexão do “...pensar a educação como prática da liberdade, voltada para a valorização da individualidade de cada sujeito, e onde os estudantes possam ter experiências representativas que ampliem suas perspectivas de pertencimento no campo musical.”

SILVEIRA (2012), em *Orquestra de Papel: Uma Ação Pedagógica Envolvendo Música e Tecnologia*, relata a experiência da empregabilidade do uso de novas abordagens de aprendizagem musical, através do uso de tecnologias como Pure Data em uma série de ações pedagógicas com alunos do Ensino Fundamental I na disciplina de educação musical. Como descrito por SILVEIRA (2012, p. 8-9) o “...processo de ação pedagógica voltada a sensibilização artística dos alunos participantes através da criação artística de paisagens sonoras mediadas por aparato tecnológico...” contribui com o “...desenvolvimento das habilidades necessárias para a organização de uma técnica instrumental baseada nos conceitos estéticos construídos a partir das experiências que ocorreram dentro da sala de aula, durante os encontros das oficinas musicais”.

DE ALMEIDA (2018), na dissertação *Aprendizagem Musical Através de Composição Assistida por Computador*, descreve a utilização de estratégias computacionais nos vários níveis de ensino de música, com práticas destinadas a auxiliar o aprendizado formal da música para indivíduos sem treino musical, em especial para a infância através do uso de tecnologia. Sobre a empregabilidade de interfaces humano-computador e a utilização de mapeamentos (OSC-MIDI) podemos afirmar que:

A criação de instrumentos electrónicos e interfaces humano-computador são exemplos típicos de tecnologia que usa mapeamentos entre diversos componentes, como sensores, botões, LEDs, e as características de sintetizadores, algoritmos ou efeitos sonoros. (DE ALMEIDA, 2018, p. 39).

DE ALMEIDA (2018, p. 39) Ainda destaca a utilização de Pure Data como programa para criação musical em tempo real “que possuem mecanismos que permitem a prática de mapeamentos”.

2.3 - Referências audiovisuais

Ao consultar o indexador de repositórios acadêmicos Google Scholar é possível encontrar diversos trabalhos que utilizam individualmente os termos OSC, Pure Data e Processing em projetos audiovisuais. Contudo, os resultados que utilizem as três terminologias em projetos audiovisuais são escassos. Os artigos encontrados estão listados nos próximos parágrafos.

Outro ponto está associado ao nome da linguagem utilizada, devido ao caráter genérico da palavra, ao incluir o termo “Processing” na busca, poucos são os resultados que realmente se referem a linguagem de programação desenvolvida por FRY e REAS (2001), sendo exibidos quaisquer resultados com referência a palavra processamento em inglês.

MARQUES (2018) apresenta a utilização de Pure Data para a produção de mídia gráfica através de processos de erros computacionais produzidos deliberadamente. Na seção “V Desenvolvimento com Pure Data, *Live Glitch & Paisagem*”, MARQUES (2018) apresenta diversos *patches* Pure Data desenvolvidos para interações com áudio em tempo real, assim como no *patch* produzido nesse trabalho.

PORCARO e LEVY (2009) detalham a criação do Blendnik, uma instalação virtual interativa, controlada via MIDI, para geração de sons e imagens em cenas performáticas. Através da interação de *patches* Pure Data, criam animações com estéticas randomizadas através da plataforma de criação de manipulação gráfica de código aberto Blender.

Na Tabela 1 o comparativo entre os trabalhos listados anteriormente:

Tabela 1 – Comparativo entre trabalhos.

Trabalho	Atuação	Comunicação	Plataforma	Valor
LuMU	Gestos e contato	WiFi – OSC	ESP32	U\$ 52,95 ¹⁸
Quad-Theremin - TSUNG-CHING, SHU-HUI e CHE-YI (2011)	Gestos manuais	Serial – MIDI	Arduino	U\$ 800 ¹⁹
Mi.Mu - Heap	Gestos manuais	WiFi – OSC	ESP32	U\$ 3233 ²⁰

¹⁸ Custo de aquisição dos insumos para produção do dispositivo, detalhamento disponível na Tabela 2.

¹⁹ Os artigos de TSUNG-CHING, SHU-HUI E CHE-YI, SARANG *ET AL.* e KLIPFEL não disponibilizam a informação, o valor apresentado é uma estimativa e não inclui custos de frete ou logística.

²⁰ Valor de comercialização do par de luvas em 28/08/2019 em mimugloves.com.

Air Drums - SARANG <i>et al.</i> (2015)	Gestos corporais	Serial - MIDI	Arduino e Kinect SDK	U\$ 250 ²⁰
MIDI Motion Gloves – KLIPFEL (2017)	Gestos e contato	Serial - MIDI	Teensy	U\$ 65 ²⁰
LUNHANI e MANNIS (2010)	Manipulação de objetos sobre plano	OSC	Reactable	U\$ 7400 ²¹
COTRIM (2020)	Não se aplica	Não se aplica	GNU/Linux	Não se aplica
SILVEIRA (2012)	Não se aplica	Não se aplica	Pure Data	Não se aplica
DE ALMEIDA (2018)	Não se aplica	Não se aplica	Não se aplica	Não se aplica
MARQUES (2018)	Reação a áudio <i>real-time</i>	Não se aplica	Pure Data	Não se aplica
Blendnik - PORCARO e LEVY (2009)	Reação a áudio <i>real-time</i>	MIDI	Pure Data / Processing	Não se aplica

Fonte: O autor.

²¹ Preço de varejo do modelo Reactable Live! S6 em 19/06/2022 divulgado em <https://newatlas.com/reactable-live-s6-table-synth/40073/#:~:text=The%20Reactable%20Live!,You%20can%20see%20the%20Live!>

3 - REVISÃO TEÓRICA

Nessa seção será apresentada uma discussão geral sobre os conceitos abordados neste trabalho, assim como a fundamentação teórica pertinente. Neste capítulo são apresentadas as fundamentações relacionadas à tecnologia, computação e ferramentas essenciais empregadas que auxiliaram no desenvolvimento, assim como os conceitos relacionados a tecnologia musical e musicologia que são utilizados no desenvolvimento do dispositivo.

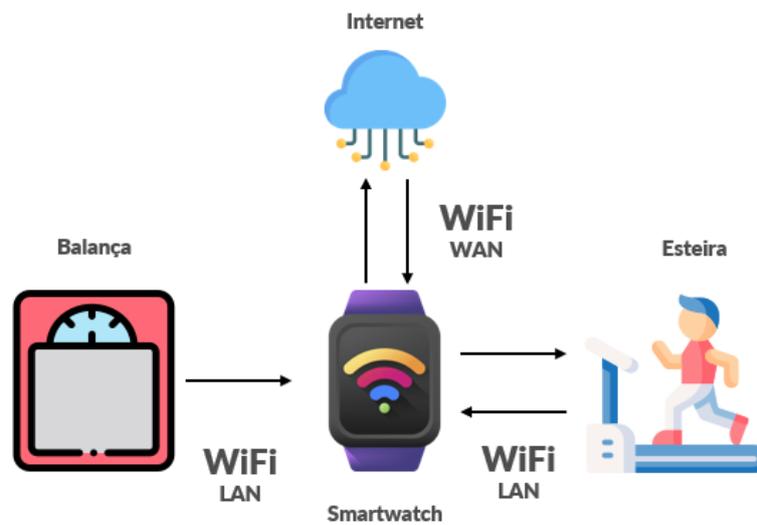
3.1 - Internet das Coisas

A expressão Internet das Coisas é designada para se referir ao novo paradigma de intercomunicação de objetos utilizados no dia a dia. Objetos que normalmente não possuem capacidade de trocar informações entre si.

A IoT caracteriza-se como um paradigma capaz de conectar objetos e fazê-los se comunicar, por meio de alguma tecnologia de comunicação, criando uma rede de interação entre objeto-a-objeto e podem coletar e trocar dados com pouca ou nenhuma interferência de seus usuários. Com mais de 20 bilhões de objetos que devem estar conectados à Internet, até 2020, essa tecnologia promete uma série de benefícios, (PEREIRA, et al. 2014 e CANHOTO, 2016, apud FRÓES e ESPUNY, 2020) e as possibilidades desse novo conceito são expandidas à medida que novos dispositivos, antes não conectados, são integrados a rede.

O seguinte cenário pode ser aventado para melhor entendimento das vantagens obtidas através da intercomunicação de objetos do dia a dia: O usuário se pesa em sua balança no banheiro de sua casa, a balança, estando conectada a rede WiFi da residência, envia informações de pesagem periodicamente ao *smartwatch* do usuário. Este último, conhecendo o histórico de variação de peso e outras informações relacionadas a saúde do usuário obtidas através de alguns sensores e também estando conectado a mesma rede WiFi, envia para a esteira ergométrica um programa de exercício personalizado a fim de estabilizar uma possível variação de peso do usuário. Ao final do processo a esteira envia ao *smartwatch* informações detalhadas sobre o exercício, a fim de fornecer ao usuário um *feedback* mais completo sobre o seu estado de saúde, através de um processo de comparação e parametrização do biotipo do usuário, graças ao acesso a uma base de dados online. Esse cenário aqui descrito é graficamente descrito na Figura 3.

Figura 3 – Cenário de utilização de IoT



Fonte: O autor²².

Para que os dispositivos e as coisas do dia a dia pudessem ter acesso a bases de dados e estar conectados também à Internet, algumas questões precisavam ser resolvidas: uma forma de identificação eficiente em custos. Apenas assim informações sobre objetos com identidade poderiam ser coletadas e processadas automaticamente. As *tags*, enquanto elementos que inseridos em objetos cotidianos corporificam uma comunicação ubíqua e estabelecem um senso de ordem, atribuem informações, sendo facilmente reconhecíveis e reproduzíveis, e dentro do caráter temporário que apresentam, podem ser reprogramadas a qualquer momento. (SANTAELLA, 2013) Esse processo de coleta dos dados, em outras palavras, o sensoriamento também precisou evoluir.

Um dos pontos fundamentais da IoT é a captação de dados ambientais, o processo de sensoriamento. Os sensores estão nas raízes do conceito de Internet das Coisas. São dispositivos importantes para coletar informações em campo, em ambientes inacessíveis, em sistemas produtivos, na área médica, na análise de distúrbios, na criação animal e outros inimagináveis processos. (FACCIONE FILHO, 2016) O avanço tecnológico possibilitou a miniaturização desses componentes de tal forma, que agora é comum a empregabilidade de sensores em peças de vestuário. Assim dispositivos chamados de vestíveis (*wearables*) se tornaram um ponto focal no cenário IoT moderno.

Wearables habilitados para IoT são dispositivos inteligentes que podem ser usados como acessórios externos, incorporados em roupas e vestuário, implantados diretamente

²² Ícones produzidos por Freepik. Disponível em: www.flaticon.com.

no corpo ou até mesmo aderidos ou tatuados na pele. Esses dispositivos são capazes de se conectar à Internet para coletar, enviar dados e receber as informações que podem ser usadas para a tomada de decisões inteligentes. (DIAN *et al.*, 2020) O dispositivo descrito neste trabalho, resume-se a uma peça de vestuário que coleta informações relacionadas a movimentação e conjunção das mãos, que por sua vez envia essas informações a uma LAN, caracterizando-o como um *wearable*.

3.2 - ESP32 E SoC

O termo ESP se refere a uma série de microcontroladores utilizados como plataformas de prototipagem de baixo custo e baixo consumo de energia, criadas pela Espressif Systems, as placas ESP são do tipo SoC. Sobre o conceito de SoC:

Em termos simples, definimos um SoC como um IC²³, projetado a partir da união de vários projetos VLSI independentes para fornecer funcionalidade completa para um aplicativo. Essa definição de SoC enfatiza claramente modelos pre-projetados de funções complexas conhecidas como núcleos (termos como bloco de propriedade intelectual, componentes virtuais e macros também são usados) que atendem a uma variedade de aplicativos. (RAJSUMAN, 2000, p. 3, tradução nossa).

Segundo JOHANN FILHO, PONTES e LEITHARDT ([20--?]) “Os projetos de SOCs estão se tornando cada vez mais complexos, e a necessidade de integração e comunicação entre diversos sistemas embarcados está se tornando característica chave dos sistemas modernos”. Essa possibilidade de comunicação multicamada disponível nos SoCs abre novos preceitos, incluindo dispositivos IoT²⁴, como dispositivos vestíveis conectados a uma rede de computadores.

Atualmente dois tipos de placa ESP são mais utilizadas para projetos *maker* e IoT, e utilizam o *devkit*²⁵ NodeMCU, a ESP8266 e sua sucessora, mais avançada, o ESP32-WROOM-32, abreviado para ESP32. Nosso projeto utiliza duas placas ESP32, uma variante de baixo custo, fabricada pela empresa WEMOS, o LOLIN D32 (Figura 4), que possui maior desempenho e pinos digitais disponíveis que sua versão antecessora. A placa D32, diferente da versão original fabricada pela ESPRESSIF, possui um conector de

²³ *Integrated Circuit*: Circuito integrado.

²⁴ *Internet of Things*; Internet das Coisas.

²⁵ *Devkit* ou placas de desenvolvimento, são plataformas de *hardware* desenvolvidas com o propósito de facilitar o desenvolvimento de dispositivos protótipos.

bateria padrão XH.

Figura 4 – WEMOS LOLIN D32 (Vista frontal e posterior)



Fonte: wemos.cc²⁶

As placas ESP são bastante utilizadas em projetos IoT e de automação residencial devido a conectividade WiFi+BL/BLE²⁷ integrada ao circuito e a interoperabilidade com a Arduino IDE. Em nosso projeto também utilizaremos essa conectividade para comunicação entre as luvas e um computador.

3.3 - Arduino IDE

A Arduino IDE é um ambiente multiplataforma desenvolvido inicialmente para programar placas da plataforma Arduino (Uno, Mega, Nano etc.); sua atuação é tão simples e prática, em comparação a programação de microcontroladores isoladamente, que outros fornecedores de placas de prototipagem possibilitaram a programação de seus produtos na Arduino IDE.

A Arduino IDE foi desenvolvida em JAVA, e é licenciada sobre GNU, versão 2. A Arduino IDE suporta linguagens C e C++, utilizando regras especiais na estrutura do código. As principais funções da Arduino IDE são: controlar as entradas e saídas dos microcontroladores, para uso de sensores e/ou atuadores, e programar rotinas de atuação para coleta de dados ou condições específicas.

²⁶ Disponível em: <<https://wiki.wemos.cc/products:d32:d32>>

²⁷ Bluetooth/Bluetooth Low Energy

3.4 - Baterias de LiFePo4

Células de LiFePO₄ são baterias que contêm fosfato de ferro e lítio como componente principal em seu catodo e um eletrodo com carbono grafítico em seu ânodo. Sua composição permite a fabricação de baterias de baixo custo, baixa toxicidade e longa duração.

Diferente das baterias de íons de lítio convencionais, baterias LiFePO₄, não oferecem riscos de incêndio ou explosão, mesmo se mantidas em funcionamento fora da faixa de operação especificada ou se houver perfuração das células. Essa segurança é requerimento básico de operação, já que tais baterias permanecem fixadas diretamente nas luvas, operando em proximidade ao corpo do usuário.

Células de LiFePO₄ não possuem a mesma capacidade de alto fornecimento de carga que uma bateria de íons de lítio convencional, porém, a taxa de fornecimento das células é suficiente para a operação da placa D32, assim apresentando um suficiente fornecimento de carga para o nosso projeto.

Em nosso projeto foi utilizada uma bateria padrão 18650 do fornecedor Soshine com as seguintes especificações:

- Capacidade / Tensão: 1600 mAh / 3.2 V;
- Dimensões / Peso: 65x18x18 mm / 60g.

Sobre a operabilidade do uso de baterias LiFePO₄ frente às convencionais baterias de íons de lítio é possível afirmar que o fosfato de lítio e ferro (LiFePO₄) é um candidato promissor para baterias de lítio de baixo custo porque tem uma alta capacidade teórica (170 mAh/g), excelente estabilidade durante o ciclo e perspectivas de uma célula mais segura em comparação com LiCoO₂. (TERADA *et al.*, 2001, *apud* STRIEBEL *et al.*, 2005).

3.5 - Linha de costura condutora

A linha de costura condutora é um dos materiais chave para o desenvolvimento desse projeto. Inicialmente fabricada para o uso com a placa LilyPad, uma variação da placa Arduino Nano desenvolvida para projetos que utilizem tecnologias vestíveis, e para utilização em luvas com capacidade *touch screen*. A linha contém aço inoxidável em sua fabricação, permitindo o fluxo de carga elétrica a uma baixa resistência (6 OHMs).

3.6 - Módulo GY-91

O módulo GY-91 é um sensor composto pela junção dos módulos MPU9250 e BPM280, resultando em um dispositivo com giroscópio triaxial, acelerômetro triaxial, leitor de campo magnético triaxial e barômetro; para esse projeto serão utilizadas a capacidade de detecção de rotação fornecida pelo giroscópio e movimentação pelo acelerômetro, parâmetros que serão mapeados para alterações em características do som.

O GY-91 possui as seguintes especificações segundo o fabricante²⁸:

- Tensão de operação: 3 – 5 V;
- Comunicação: I2C e SPI;
- Intervalo de leitura do giroscópio: ± 250 500 1000 2000 °/s;
- Faixa de leitura da aceleração: 4 2 $\pm \pm \pm \pm$ 8 16g;
- Faixa de leitura de campo magnético: $\pm 4800\mu\text{T}$;
- Faixa de leitura de pressão: 300-1100hPa.

3.7 - Pure Data

Pure Data também designado como PD, desenvolvido por Miller Puckette, é uma VPL (*Visual Programming Language*²⁹) com foco de utilização em aplicações multimídia. Sua utilização se dá através da manipulação de entidades visuais. A realização de interconexões e de outras manipulações tem como resultado a geração de informação, áudio, vídeo, texto e outros multimeios, como mostra a Figura 5.

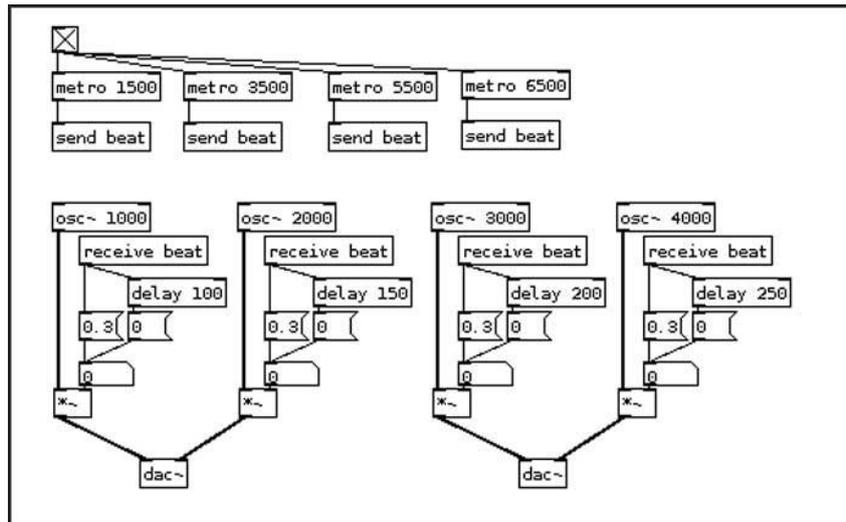
De código aberto, Pure Data atua com quatro tipos básicos de entidades: mensagens, objetos, átomos e comentários. Outras entidades derivativas são obtidas a partir dos quatros tipos originais. A atuação do PD é semelhante ao seu concorrente comercial MAX/MSP, porém com algumas disparidades.

²⁸ Disponível em:

<https://pt.aliexpress.com/item/1005002956541258.html?spm=a2g0o.productlist.0.0.45b64c48gcK9NG&algo_pvid=bf789e9a-daa0-43bb-996b-6d439fa9a036&algo_exp_id=bf789e9a-daa0-43bb-996b-6d439fa9a036-0>

²⁹ Linguagem de Programação Visual

Figura 5: Interface Pure Data



Fonte: Soundbytes Magazine³⁰.

3.8 - Open Sound Control (OSC)

O OSC é um protocolo de comunicação de instrumentos musicais e dispositivos multimídia. O OSC foi projetado para permitir o compartilhamento de informações musicais em tempo real sobre uma rede (WiFi, ethernet e outros). O OSC se assemelha ao MIDI, contudo, possui características e capacidades superiores. Segundo PHILLIPS (2008, tradução nossa) OSC tem como principais características:

- Esquema de nomenclatura simbólica em *open-ended*, dinâmico e URL orientada;
- Dados numéricos e simbólicos de alta resolução;
- *Pattern matching* para especificar vários destinatários de uma única mensagem;
- *Tags* de tempo de alta resolução;
- "Pacotes" de mensagens cujos efeitos devem ocorrer simultaneamente;
- Sistema de consulta para descobrir dinamicamente os recursos de um servidor OSC e obter documentação;
- O sistema baseado em rede utiliza mecanismos comuns de transporte UDP/TCP.

De forma simples, podemos tratar do OSC como o mesmo objetivo do protocolo MIDI, porém orientado para execução em redes. O protocolo OSC pode ser implementador também em computadores pessoais. Existem diversas ferramentas que auxiliam a implementação de mensagens OSC em PCs, entre elas podemos citar a linguagem de programação Pure Data.

³⁰ Disponível em: < <https://soundbytesmag.net/pd-pure-data-a-public-domain-patching-environment/> >.

3.9 - *Musical Instrument Digital Interface (MIDI)*

MIDI é o acrônimo em inglês para Interface Digital de Instrumentos Musicais. É um protocolo de comunicação para interconexão física, primariamente, entre instrumentos musicais. O padrão foi definido em 1983 pela MIDI Manufacturers Association. Uma conexão MIDI pode ter até 16 canais de comunicação, no qual cada um pode controlar parâmetros específicos de um equipamento, como altura do som, velocidade, *panning*, entre outros. Essa comunicação geralmente ocorre por um cabo padrão MIDI ou, de forma menos comum, por algum padrão de conexão de rede de computadores, como ethernet, WiFi ou Bluetooth (ou BLE).

O padrão MIDI engloba variados aspectos relacionados a tecnologia musical, porém para o desenvolvimento de nosso projeto as mensagens MIDI representam a estrutura base, juntamente com o padrão de mensagens OSC, na comunicação entre as luvas e um computador. Segundo BREWSTER (2003, Ed. 2, p. 227) “Uma mensagem MIDI é uma instrução que controla algum aspecto do dispositivo receptor. Uma mensagem MIDI consiste de um *byte* de estado, que indica o tipo da mensagem, seguido por até dois *bytes* de dados que contêm os parâmetros.”

3.10 - *Digital Audio Workstation (DAW)*

Digital Audio Workstation, é a denominação de dispositivos voltadas à manipulação de áudio. DAW são, resumidamente, sequenciadores que tem a finalidade de gravar, tocar e editar áudio digital. São Sistemas de gravação multipista implementado em um microcomputador convencional ou num equipamento autônomo. Usa disco rígido como meio de armazenamento. As DAW possuem dispositivos de digitalização de áudio (conversores A/D e D/A) e software específico. (RATTON, 2009, p. 49)

Atualmente DAWs evoluíram quase que exclusivamente para o formato de suítes de *softwares*, contendo todas as ferramentas necessárias para produção de áudio digital. Podendo ser projetados com qualquer interface, vantajosamente os DAWs acumulam funções de diversos dispositivos utilizados por trabalhadores da indústria fonográfica, como *mixer*, *samplers*, sintetizadores (digitais), unidades de efeito, compressores, sequenciadores, entre outros.

Existem vários DAW disponíveis no mercado, pode-se citar os mais utilizados, com o respectivo fabricante:

- Cubase (Steinberg)
- REAPER (Cockos)

- ACID Pro (Sony)
- Live (Ableton)
- FL Studio (Image-Line)
- Pro Tools (Avid)
- Logic Pro (Apple)

Por conta da saturação na disponibilidade de *softwares* no mercado, algumas empresas desenvolveram DAW voltados a aplicações específicas para atingir um nicho específico do mercado, como o Live, uma suíte de aplicações para áudio voltada para apresentações ao vivo desenvolvido pela empresa alemã Ableton. Alguns DAWs possuem a vantagem de serem de uso gratuito, como o REAPER e Ableton Live Lite.

3.11 - Processing

Processing é uma linguagem de programação criada por Casey Reas e Ben Fry em 2001, com código aberto. Processing é comumente utilizada no desenvolvimento de aplicações visuais e na criação de ferramentas para design de interação. Segundo REAS e FRY (2006, tradução nossa) “A linguagem Processing é uma linguagem de programação de texto projetada especificamente para gerar e modificar imagens”. Programas desenvolvidos em Processing, assim como na Arduino IDE, são chamados de rascunhos ou *sketchs*.

4 - METODOLOGIA

Os processos de desenvolvimentos aqui preconizados neste trabalho de dissertação encontram-se essencialmente firmados sobre as prerrogativas do *Design Science Research*. O DSR tem como presunção base a descrição do desenvolvimento de artefatos. As etapas desse método consistem essencialmente na identificação do problema, conscientização e revisão sistemática do problema na literatura, definição da classe de problemas, estruturação de artefato com o intuito de resolver o problema, avaliação do artefato, explicitações das aprendizagens e conclusões (DA NÓBREGA, 2019).

Esta dissertação tem por eixo a aplicação do DSR na descritividade dos mais diferentes processos de fabricação do dispositivo. Referente ao *hardware*, essa metodologia já possui sua empregabilidade universalizada. Referente ao desenvolvimento de *software*, o DSR oferece um meio eficaz de abordar a lacuna de relevância que tem atormentado a pesquisa acadêmica, particularmente nas disciplinas de gerenciamento e sistemas de informação. Os métodos de pesquisa em ciências naturais são apropriados para o estudo de fenômenos existentes e emergentes; no entanto, eles são insuficientes para o estudo de "problemas organizacionais perversos", o tipo de problema que exige soluções criativas, novas e inovadoras. Tais problemas são abordados de forma mais eficaz usando o tipo de mudança de paradigma oferecido pelo DSR (HEVNER, CHATTERJEE, 2010, *tradução nossa*).

Esses problemas podem ser traduzidos como impedimentos para o firmamento dos objetivos aqui propostos. Há alguns modelos que têm intenção de regular como se deve conduzir esse tipo de pesquisa, uma das opções se chama ciclo regulatório. Este atua como um organograma que de forma lógica orienta a resolução de problemas de ordem prática (DRESCH et al, 2015; WIERINGA, 2009, apud BRANDÃO, 2020).

O ciclo regulatório consiste na observação de um fluxo de ação predeterminado. Tal fluxo se inicia da avaliação da implementação, através da revisão da problemática inicial e das primeiras etapas de desenvolvimento, seja de *hardware* ou *software*. Em seguida é realizada uma análise do projeto de soluções. Essa fase consiste no desenvolvimento de um modelo de ações a serem estabelecidas a fim de conceber a produção do artefato. No próximo ponto, a validação de projeto (testes com usuários), consistindo da prévia utilização da solução desenvolvida por um nicho de usuários semelhantes aos futuros usuários reais, tal como produzido nessa dissertação. Em sua última fase há a implementação da solução, contudo tal dissertação se limita a um estudo

descritivo do desenvolvimento do dispositivo, por tanto tal fase não se aplica ao proposto.

Os resultados obtidos por esse trabalho serão utilizados como lições aprendidas, fomentando futuras alterações para o aprimoramento do dispositivo em seu desenvolvimento final a ser explorado em trabalhos futuros. Tal contexto de lições aprendidas é explorado comumente no DSR como fator de descrição dos resultados do ciclo regulatório. Ainda neste contexto, esse projeto apresenta o desenvolvimento de um dispositivo controlador sem fios com viés musical; com o propósito de se obter uma ferramenta móvel e de fácil utilização. Todas as ferramentas empregadas em seu desenvolvimento são de fácil aquisição e baixo custo.

Instrumentos controladores³¹ são capazes de controlar instrumentos virtuais, reproduzir *samples* em um computador, modular características do som e controlar outros instrumentos físicos. Assim como qualquer instrumento controlador, o LuMU consegue controlar aspectos relacionados a execução de sons e realizar processos de modulação de sons, incluindo, mas não limitado a adição de efeitos como reverberação e *delas*, porém o LuMU se diferencia da maioria dos instrumentos controladores por dois aspectos, primeiro pela capacidade de funcionamento sem fios, através do IEEE 802.11, e segundo por se tratar de um dispositivo *wearable* no formato de luvas, alimentado por baterias, alcançando assim um padrão de locomobilidade superior aos instrumentos controladores mais utilizados.

O desenvolvimento do dispositivo está descrito nas próximas seções em fases.

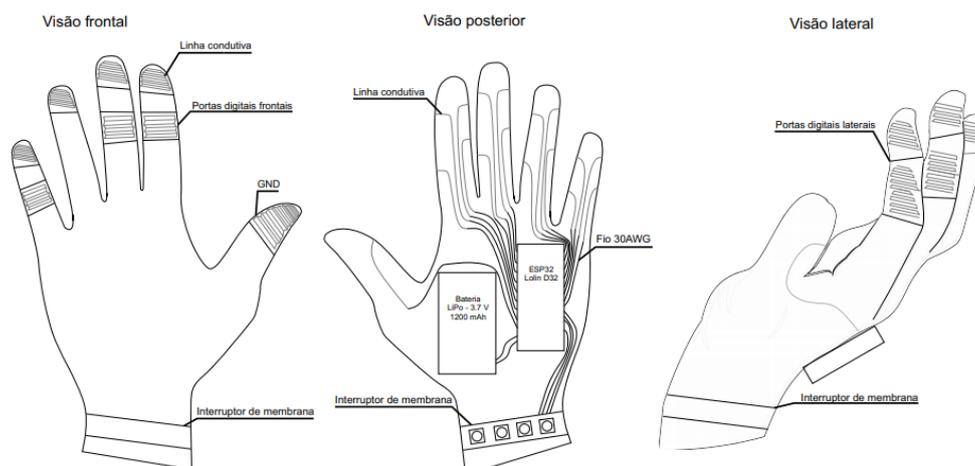
4.1 - Primeira fase - ideia inicial, modelagem do protótipo e aquisição dos insumos

A primeira fase do projeto foi orientada para modelagem do dispositivo, que inclui o desenvolvimento do esboço inicial do protótipo e escolha dos componentes mais adequados para as tarefas propostas. A Figura 6 mostra o diagrama inicial do protótipo para mão direita em visão frontal, posterior e lateral, e a Figura 7 para mão esquerda:

Ficando evidentes todos os aparatos necessários para produção do protótipo, realizou-se a aquisição através de importação de fornecedores asiáticos, evidenciando a não produção nacional da maioria dos itens necessários. O custo total de aquisição dos insumos está descrito na Tabela 2.

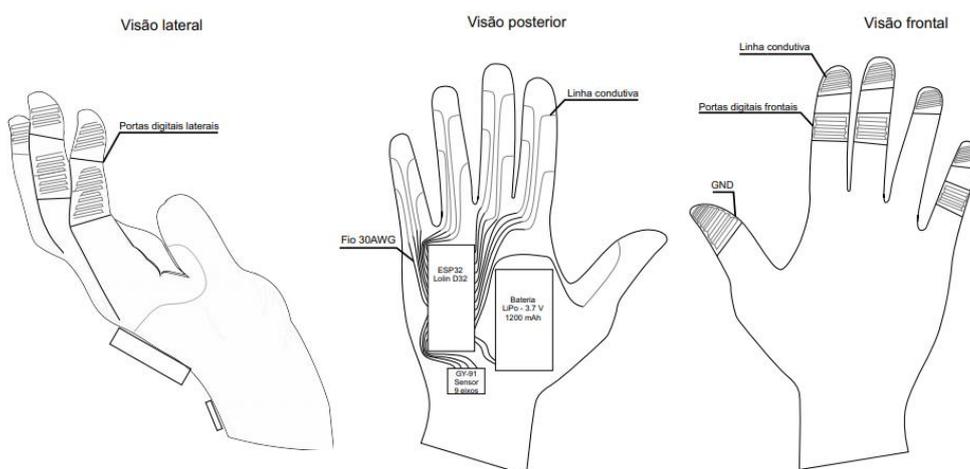
³¹ São equipamentos com capacidade de comunicação com outros dispositivos através de algum protocolo de comunicação auxiliar, como OSC ou MIDI. Instrumentos controladores não “criam sons” ao serem performados, eles criam informações que são enviadas para um dispositivo hospedeiro (como computadores, sintetizadores e *samplers*), o qual cria ou executa algum som previamente programado.

Figura 6 – Diagrama para mão direita



Fonte: O autor.

Figura 7 – Diagrama para mão esquerda



Fonte: O autor.

Tabela 2 – Listagem e custo de insumos³²

Equipamento	Preço (R\$)
1 Par de luvas	41,85
20 Metros de fio costurável condutível	43,58
2 Baterias LiFePO4 de 3.7 V e 1200 mAh	60,18
2 ESP32 (Wemos Lolin D32)	70,71
1 Modulo GY-91	24,75

³² Preços de acordo com listagem em < <https://pt.aliexpress.com> >

1 Interruptor de membrana de 4 botões	12,78
10 Metros de fio 30AWG	24,17
10 Bastões de cola preta	19,03
Total	297,05

Fonte: O autor.

4.2 - Segunda fase - montagem do protótipo

4.2.1 - Fixação dos componentes

Na segunda fase do projeto foi realizada a fixação dos componentes na luva, conforme Figura 8 A. Os componentes foram fixados utilizando um polímero termotivado, em uma base de espuma, para compor elevação suficiente para se trabalhar nos contatos das placas D32 e GY-91, como mostrada na Figura 8 B.

Figura 8 – Montagem do protótipo



Fonte: O autor.

Durante esta fase, objetivou-se construir a base do protótipo visando a eficiência no uso final do dispositivo. Para isso, vários testes de uso do polímero colante foram conduzidos antes da aplicação no material, em conjunto com a experimentação de variados tipos de agulhas em amostras de tecidos com textura semelhante. Algumas agulhas apresentaram deterioração, não sendo adequadas ao material. Optou-se por usar uma agulha do tipo Darning, com tamanho N1 (6 cm), tendo ela uma melhor maleabilidade no material da luva.

Os componentes apresentaram boa fixação, não aparentando sinais de folga ou possibilidade de deslocamento através de movimentação forçosa na luva; o tecido que

constitui a luva também não apresentou desgaste e degeneração aparente com o calor aplicado no uso do polímero.

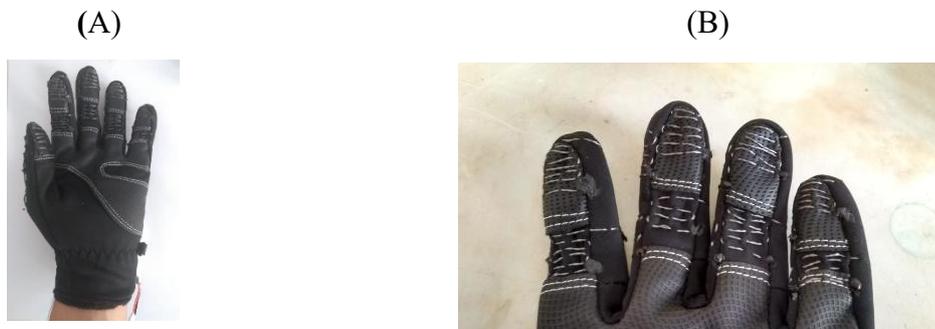
4.2.2 - Conexão dos componentes

Após a fixação dos componentes na luva, foi realizado o processo de costura dos fios condutores nas zonas específicas dos dedos, assim como mostrada na Figura 9 A. A costura foi realizada à mão.

Durante o processo optou-se pela inclusão de mais zonas de contato no dedo anelar do que projetado inicialmente, sendo conveniente as portas sobressalentes para a possibilidade de utilização de mais *samples* por mão.

Para ligação das seções presentes nos dedos às portas digitais da placa D32, foi utilizado fio de cobre esmaltado, padrão 30AWG. A ligação entre o fio de cobre e a linha de costura, Figura 10, foi feita de forma mecânica (nó meia volta) com aplicação de uma camada de polímero colante. Para ligação entre o fio de cobre e as portas do D32 foi utilizada solda de estanho-chumbo (solda branca). Mesmo com as alterações estruturais, a luva ainda apresenta mobilidade e conforto, é possível fechar a mão sem haver tensionamento considerável dos fios.

Figura 9 – Costura da linha condutiva



Fonte: O autor.

Figura 10 – Conexão das portas do dispositivo



Fonte: O autor.

4.3 - Terceira fase - desenvolvimento dos *softwares* e configuração no DAW

4.3.1 - Desenvolvimento do sketch - Arduino IDE (código embarcado)

Para o desenvolvimento do *software* embarcado no SOC, foi utilizada a Arduino IDE. O código-fonte se define em três seções, declarações de variáveis globais e distanciamento de bibliotecas de código, *Setup* e *Loop*.

Na primeira seção, há a declaração de variáveis globais e introdução das bibliotecas de código utilizadas pelas seções posteriores; na segunda seção há a inicialização de recursos necessários para o correto funcionamento do sistema, a configuração do roteamento WiFi, a definição de modo de trabalho para as portas digitais e inicialização da porta serial. No Arduino IDE a instância *Setup* () é executada apenas uma única vez. Na terceira seção, *Loop* (), está o corpo do código, a execução de todas as rotinas necessárias para traduzir o acionamento das portas digitais do ESP32 em mensagens OSC, através do protocolo UDP via WiFi (rede criada pelo próprio dispositivo). O código fonte completo está disponível no Apêndice A.

4.3.2 - Desenvolvimento dos *patches* (Pure Data)

Dois *patches* foram desenvolvidos em Purr Data³³, uma plataforma cruzada de desenvolvimento em Pure Data desenvolvida com JavaScript, criada por Jonathan Wilkes.

O Purr Data é uma variação da plataforma Pd-L2Ork para Linux, criada por Ico Bukvic, que por sua vez é uma variação do PD-Extended, um ambiente de desenvolvimento em Pure Data com vários elementos externos pré-embutidos, facilitando assim o desenvolvimento. Embora o PD-Extended tenha sido descontinuado pelos seus desenvolvedores em 2013, o Purr Data e o Pd-L2Ork têm suporte ativo continuado e constantemente recebem novas atualizações.

Para os *patches* atuarem como *applets* para o projeto LuMU, o *sender* e o *receiver* utilizam o *external* Mr.Peach, desenvolvida por Martin Peach, como principal elemento. O Mr.Peach é uma biblioteca desenvolvida para facilitar o processo de comunicação do Pure Data com a rede local e para controle de mensagens OSC.

4.3.2.1 - *Sender*

O primeiro *patch*, nomeado como *sender*, atua como uma simulação da atuação de duas

³³ Disponível em: < [https://puredata.info /downloads/purr-data](https://puredata.info/downloads/purr-data) >

zonas de contato de um dedo do LuMU, como mostra a Figura 11. Enviando mensagens OSC via rede LAN, através do protocolo UDP.

Figura 11: Zona simulada por *Patch Pure Data*



Fonte: O autor.

O *sender* foi desenvolvido primariamente para facilitar os processos de teste e desenvolvimento do *receiver*. Para cada teste do *receiver* era necessário iniciar o LuMU e realizar a conexão do computador a rede WIFI gerada pelo dispositivo, o que demandava um maior esforço e tempo para a realização de simples testes de recepção do protocolo UDP.

Assim, o *sender* foi construído para simular o mesmo processo de envio de mensagens OSC realizado pelo dispositivo, porém sem a contraparte de aguardar o *set-up* do mesmo, tornando o processo como um todo mais prático. O *patch* desenvolvido para o *applet sender* pode ser visto na Figura 12.

Figura 12: *Patch sender*



Fonte: O autor.

O fluxo de atuação do *sender* é descrito a seguir:

1. Uma conexão é estabelecida através da porta local 8333;
2. Uma mensagem OSC contendo a informação MIDI *note on* (nota pressionada) é enviada ao objeto seguinte;
3. A mensagem OSC é convertida em uma lista de números *float* (encapsulamento);
4. A lista de números é enviada via UDP;
5. Uma mensagem OSC contendo a informação MIDI *note off* (nota solta) é enviada ao objeto seguinte;
6. A mensagem OSC é convertida em uma lista de números *float*;
7. A lista de números é enviada via UDP.
8. Caso não seja mais necessário envio de mensagens OSC o *patch* é desconectado da porta 8333;

4.3.2.2 - *Receiver*

O segundo *patch*, nomeado como *receiver*, atua como um tradutor das mensagens OSC obtidas enviadas pelo *sender*, ou diretamente pelo LuMU, em mensagens MIDI a serem enviadas para um software DAW.

O *receiver* foi desenvolvido para que o LuMU tenha uma solução em software não-proprietário para conversão de suas mensagens OSC em rede local via WIFI para mensagens MIDI, assim controlando instrumentos virtuais ou reais de forma mais transparente. As mensagens MIDI são enviadas pelo objeto *noteout* através de um *driver* de MIDI virtual para o DAW. O *patch* desenvolvido para o *applet receiver* pode ser visto na Figura 13.

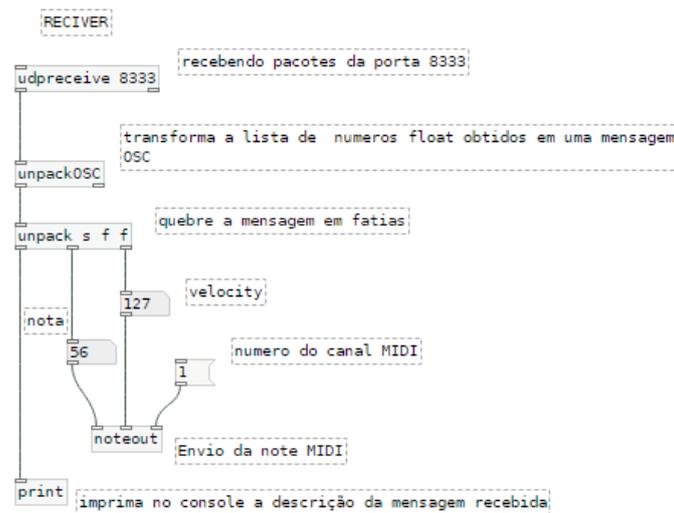
O fluxo de atuação do *receiver* é descrito a seguir:

1. O *patch* monitora a porta 8333 aguardando o recebimento de quaisquer pacotes OSC “encapsulados”;
2. Os pacotes recebidos são convertidos de uma lista de números *float* para uma mensagem OSC convencional;
3. A mensagem é destrinchada em seções, uma variável *symbol (string)* para descrição do dedo e da zona pressionada pelo LuMU ou simulada pelo *sender* e duas variáveis *floats* para descrição da nota a ser tocada e da informação de *velocity* da nota;
4. A variável *symbol* é impressa no console do Purr Data como referência da ação

executada;

5. As informações de *velocity* e número da nota são enviadas para o *noteout*, responsável por enviar informação MIDI para o *driver* de MIDI virtual;
6. A informação é recebida pelo DAW que executa a ação correspondente, sendo a execução de uma nota musical ou ação de algum comando especificado pelo usuário.

Figura 13: *Patch receiver*



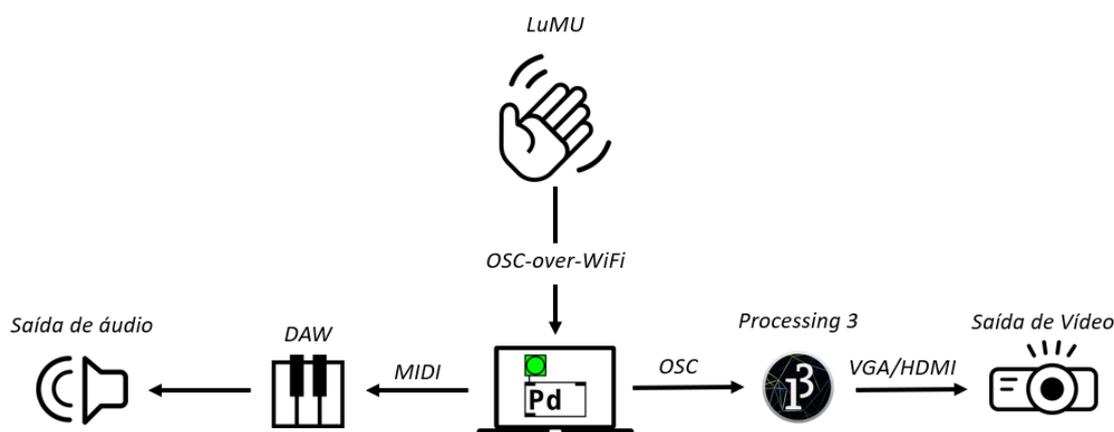
Fonte: O autor.

4.3.3 - Desenvolvimento do sketch - Processing

O Funcionamento do *applet* desenvolvido em Processing tem a finalidade de alternar a execução de arquivos de vídeo definidos, através da captação de mensagens “gatilho” OSC na rede local enviadas pelo projeto LuMU, ou por *applet* Pure Data, simulando o comportamento do LuMU. Desta forma é possível criar cenários visuais a serem exibido com projetores para performance com o LuMU, como pode ser visto na Figura 14. O controle de recepção de mensagens OSC no Processing é feito pela biblioteca *oscP5*, desenvolvida por SCHLEGEL (2011).

O *sketch* desenvolvido para o *applet* descrito nessa seção é constituído de 3 blocos principais: *pre-setup*, *setup* e *draw*. O *pre-setup* ilustra o bloco de declarações iniciais. Em seguida o *setup*, trecho designado para importação de bibliotecas de código necessárias e declarações de variáveis globais. Por último a execução do *draw*, função que é executada em *loop*. O código fonte completo está disponível no Apêndice B.

Figura 14: Diagrama de funcionamento do LuMU e *Applets*



Fonte: O autor.

4.3.4 - Compêndio de *software* e *samples*

Essa seção consiste de uma listagem exemplificativa (não taxativa) de *softwares* e instrumentos virtuais e *samples* para utilização em sala de aula, ou em atividades performáticas.

Referente aos *softwares* DAW e instrumentos virtuais, a Tabela 3 lista ferramentas que tiveram seu uso efetivamente testado com o LuMU. Contudo, essa listagem representa apenas uma parcela infinitesimal de possibilidades de uso, tendo em vista a abrangência da compatibilidade do protocolo MIDI.

Tabela 3: *Softwares* testados com o LuMU

Nome	Tipo	Sistema operacional	Licença
Ableton Live ³⁴	DAW	Windows, Mac OS	Pago (90 dias de <i>trial</i>)
Reaper ³⁵	DAW	Windows, Mac OS, Linux	Pago (60 dias de <i>trial</i>) ³⁶
Cubase LE ³⁷	DAW	Windows, Mac OS	Livre
Garageband ³⁸	DAW	Mac OS	Livre
Sound Magic Piano One ³⁹	VSTi - Piano	Windows, Mac OS	Livre
U-he Zebralette ⁴⁰	VSTi - Synth	Windows, Mac OS, Linux	Livre

³⁴ Disponível em: <https://www.ableton.com/en/trial>

³⁵ Disponível em: <https://www.reaper.fm/download.php>

³⁶ Após o encerramento do período de *trial*, diferentemente de outros DAW pagos, o Reaper continua livre para uso.

³⁷ Disponível em: <https://www.steinberg.net/cubase/le/>

³⁸ Disponível em: <https://apps.apple.com/br/app/garageband/id682658836>

³⁹ Disponível em: <https://neovst.com/piano-one/>

⁴⁰ Disponível em: <https://u-he.com/products/zebralette>

Dexed ⁴¹	VSTi - <i>Synth</i>	Windows, Mac OS	Livre
Vember Audio Surge ⁴²	VSTi - <i>Synth</i>	Windows, Mac OS	Livre
Dub Turbo DrumTroop ⁴³	VSTi - Bateria	Windows, Mac OS	Livre
Steven Slate Drums SSD5 Free ⁴⁴	VSTi - Bateria	Windows, Mac OS	Livre

Fonte: O autor.

A Tabela 4 lista locais, incluindo bibliotecas e sites da *internet*, onde é possível obter *samples* gratuitamente para utilização com o LuMU em conjunto com algum *sampler* nativo ou integrado a um *software* DAW.

Tabela 4: *Samples* para utilização com o LuMU

Nome	Tipo
Cymatics Free Vault ⁴⁵	Percussivos, harmônicos e melódicos
That Sound Gratis Pack ⁴⁶	Percussivos
Black Octopus - Ultimate Free Sample Collection ⁴⁷	Percussivos e efeitos sonoros
SampleFocus ⁴⁸	Percussivos, harmônicos, melódicos, efeitos sonoros e miscelânea

Fonte: O autor.

⁴¹ Disponível em: <https://asb2m10.github.io/dexed>

⁴² Disponível em: <https://surge-synthesizer.github.io>

⁴³ Disponível em: <https://www.kvraudio.com/product/drumtroop-by-dubturbo>

⁴⁴ Disponível em: <https://stevenslatedrums.com/ssd5/#SSD5FREE>

⁴⁵ Disponível em: <https://cymatics.fm/pages/free-download-vault>

⁴⁶ Disponível em: <https://store.iwantthatsound.com/products/gratis>

⁴⁷ Disponível em: <https://blackoctopus-sound.com/free-downloads>

⁴⁸ Disponível em: <https://samplefocus.com>

5 - TESTES COM USUÁRIOS

O processo de testes com usuários foi conduzido com a presença de indivíduos com histórico de atuação no campo da educação musical. Esse processo foi conduzido por 3 etapas, uma etapa de configuração do dispositivo, uma etapa de demonstração prática de utilização do dispositivo, em seguida uma etapa de aplicação de um questionário avaliativo.

O escopo de testes foi desenvolvido a fim de verificar o atendimento dos requisitos inicialmente propostos, entre eles estão a verificação de aplicabilidade da interface HCI para controle de sons, incluindo disparo de sons, altura e timbre em sala de aula, no contexto de aula de música. verificação da viabilidade da utilização do protótipo desenvolvido com placas de desenvolvimento ESP32 com finalidades educacionais e avaliar a complexidade de uso e facilidade de configuração do dispositivo em seu atual estágio de desenvolvimento.

É significativo ressaltar que o escopo do processo de testes foi direcionado as capacidades de execução e manipulação de áudio do dispositivo, não contemplando a funcionalidade descrita na seção 4.4.3. Como detalhado anteriormente, o *applet* realiza manipulações de arquivos de vídeos, funcionalidade que, inicialmente, foge do escopo de atuação dos usuários participantes. Contudo, tal dedução não altera a validade do processo testes, uma vez que o ponto focal do dispositivo, como instrumento musical auxiliar em sala de aula, é a reprodução e manipulação de sons.

5.1 - Sessões de teste

Esse processo foi conduzido em duas oportunidades distintas. A primeira foi sessão foi conduzida com oito discentes do curso de Música Licenciatura. A segunda sessão foi conduzida com duas licenciandas em Música. Todos os usuários participantes, possuem experiência prática em ensino de Música.

Figura 15: Sessão de testes



Fonte: O autor.

5.1.1 - Configuração do dispositivo

Os participantes eram instruídos a ler o Manual de Configuração e Utilização do LuMU (APÊNDICE C) e executar o passo a passo de forma autônoma, isto é, sem o auxílio do facilitador ou dos demais participantes. Cada participante executou o processo individualmente uma vez. O propósito da execução autônoma dessa etapa é devido a posterior avaliação do grau de dificuldade do processo de configuração do dispositivo.

O processo de configuração se resume a preparação da suíte de aplicativos a serem instalados no ambiente do computador com o sistema operacional Microsoft Windows (funcionais em quaisquer versões acima da versão Windows Vista) e abertura dos programas e arquivos necessários para deixar o usuário pronto para a execução de sons no dispositivo.

O primeiro software a ser instalado é a suíte PurrData, em seguida uma versão de avaliação da suíte Ableton Live 11. Após a instalação dos softwares o usuário é orientado a abrir o arquivo “Receiver.pd” (*Patch* responsável pela captação de mensagens OSC-over-UDP), localizado na pasta “LuMU Experimento” localizada na área de trabalho do sistema. Se a instalação do *software* ocorreu com sucesso, o arquivo é automaticamente aberto pelo PurrData.

Em seguida o usuário deve abrir o arquivo “Lumu Experimento.als” localizado dentro da pasta “Lumu Experimento Project”. Se a instalação do *software* também ocorreu com sucesso, o arquivo é automaticamente aberto pelo Ableton Live 11.

5.1.2 - Demonstração prática de utilização do dispositivo

Essa etapa é caracterizada pela demonstração da utilização do dispositivo. Diferentemente da primeira etapa, nesta etapa os usuários foram instruídos em dois níveis distintos: sobre a prática de como performar com o dispositivo. Acerca da distribuição de suas notas musicais, respectivos intervalos e execução tátil do instrumento; E uma breve explicação do funcionamento da suíte Ableton Live, principalmente no que é tangível a troca de canais, para um entendimento de como realizar a troca entre instrumentos virtuais e samples percussivos.

Após o processo instrutivo inicial, os usuários eram convidados, um a um, para utilizarem o dispositivo livremente por cerca de dez minutos. De tal forma que fosse possível ao usuário minimamente compreender o viés de execução das notas, reprodução de *samples* e entender as capacidades e indisponibilidades do dispositivo.

5.1.3 - Aplicação de questionário

Após a livre utilização do dispositivo, nessa etapa os participantes eram convidados a responder um questionário com 9 itens (APÊNDICE D). Divididos em 7 perguntas em escala *likert*, de caráter avaliativo (discordo totalmente – concordo totalmente), a respeito da utilização do dispositivo em sala de aula, 1 pergunta de múltipla escolha, sobre em quais tipos de aulas o dispositivo poderia ser utilizado e 1 pergunta aberta para a inserção de sugestões no desenvolvimento futuro do dispositivo. Anexo ao questionário há presente um termo de consentimento livre e esclarecido (APÊNDICE E). Os resultados do questionário são apresentados e discutidos na seção 6.3 – Resultado do questionário

6 - RESULTADOS E ANÁLISES

Nesta seção serão apresentados e discutidos os resultados obtidos referentes ao *software* e *hardware*, assim como o resultado do questionário realizado no processo de testes com usuários.

6.1 - Resultados relacionados ao *software*

Os *applets* desenvolvidos executam as funções designadas. Os objetivos propostos foram contemplados e executados. O *applet sender* atua corretamente como um simulador para desenvolvimento de futuras implementações em *software* para o projeto LuMU. Não foi constatado em período de desenvolvimento e utilização nenhum erro ou conduta não desejada na lógica da aplicação. O *applet receiver* atua de forma precisa como tradutor e *handler* das mensagens enviadas. Não há atraso humanamente perceptível no processo de conversão, que possa comprometer na execução do LuMU como instrumento musical controlador e sua utilização em sala de aula.

O *applet* desenvolvido com Processing 3 para criação de cenário visual apresenta execução satisfatória, executando a variação três cenários visuais pré-estabelecidos. Contudo, em trabalhos futuros será estudada a aplicabilidade na utilização de um maior número de cenários visuais e sua real aplicabilidade em sala de aula.

6.2 - Resultados relacionados ao *hardware*

No dispositivo não há latência⁴⁹ aparente, supõe-se que o atraso que pode ser considerado, contudo não humanamente perceptível, está relacionado à transmissão do protocolo UDP sobre WiFi e ao *buffer size*⁵⁰ do *drive* da interface de áudio⁵¹ utilizada.

Algumas ressalvas são necessárias. Se o considerarmos o LuMU como um teclado controlador padrão, evidencia-se a atual incapacidade da execução de acordes. Devido a forma como o código-fonte foi desenvolvido, ao haver um acionamento de porta digital, uma estrutura de seleção é ativada, onde o fluxo do código permanece estagnado até a finalização do envio da mensagem OSC. Não sendo possível a reprodução de duas ou

⁴⁹ “Período de Latência é a diferença de tempo entre o início de um evento e o momento em que os seus efeitos se tornam perceptíveis [...]”. (KELLER, 2014)

⁵⁰ “Termo utilizado de forma genérica para se referir a uma parte da memória de um equipamento digital que armazena os dados temporariamente. [...] Em *softwares* de áudio digital, os *buffers* são usados pelo processador para colocar os dados de áudio que devem ser tocados e gravados; [...]”. (RATTON, 2009)

⁵¹ Placa de áudio que realiza a conversão analógico/digital e digital/analógico.

mais notas ao mesmo tempo, havendo um atraso de milissegundos entre elas, e tal atraso é humanamente perceptível.

Algumas possibilidades para contornar esse entrave serão abordadas em trabalhos futuros, como a emulação do uso de acordes via *software*, através do mapeamento de canais MIDI ou a utilização de paralelismo⁵², permitindo execução de comandos simultaneamente.

Outro ponto a ser analisado é que por utilizar o protocolo UDP, que não possui uma estrutura de controle na entrega das mensagens, pode haver perda de informação no envio das mensagens OSC, porém, é um evento de baixa ocorrência, após cerca de 2 horas de utilização contínua do dispositivo, apenas 1 falha no envio de notas foi percebida.

Após a realização do processo de testes com usuários, descrito na seção 5, o regulador de tensão das placas ESP32 apresentaram mal funcionamento, não sendo mais possível a utilização de baterias com as placas fixadas na luva. Esse tipo de cenário de falha de hardware já era esperado. Devido ao caráter de protótipo do dispositivo e a sua utilização de placas de desenvolvimento pré-fabricadas, como a Wemos D32. A versão final do dispositivo, a ser explorada em trabalhos futuros, deverá optar pela utilização direta do chip (sem uma placa de desenvolvimento), assim economizando espaço e diminuindo a chance de falha de *hardware*.

No *hyperlink* <https://bit.ly/3aHoA1v> está disponível um breve vídeo com funcionamento do dispositivo.

6.3 - Resultados do questionário

Essa sessão trata de realizar uma síntese dos resultados obtidos através da aplicação do questionário. Todas as respostas, com seus respectivos gráficos, estão integralmente presentes no Apêndice F.

A primeira questão busca avaliar a primeira etapa para utilização do dispositivo, a respeito do processo de configuração original, através do questionamento: “Eu acredito que o dispositivo é de fácil configuração”. A convergência das respostas aponta para uma resposta mediana sobre a facilidade de configuração do dispositivo, com nenhuma resposta aos extremos (concordo ou discordo totalmente) (APÊNDICE F, F-1). Embora esse processo seja realizado apenas uma única vez, podemos inferir que o atual processo

⁵² É uma forma de computação em que vários cálculos são realizados ao mesmo tempo (ALMASI e GOTTLIEB, 1989)

de configuração pode vir a ser uma barreira para a utilização do dispositivo. E que é imperativo desenvolver uma nova forma mais simplificada para a realização da configuração inicial.

Em seguida a próxima questão aborda sobre a utilização do dispositivo com o questionamento: “Eu acredito que o dispositivo é de fácil utilização”. Em contraste a primeira pergunta, a disposição das respostas desmonta para um possível atributo positivo do dispositivo. Com 50% das respostas no extremo 5 (concordo totalmente) e o restante no ponto 4 (APÊNDICE F, F-2). Podemos inferir, que para os participantes, o dispositivo apresenta um uso fácil e intuitivo.

A terceira pergunta visa esclarecer a cerca da curva de aprendizagem para uma efetiva utilização do dispositivo, através o questionamento: “Na sua opinião o aprendizado para a utilização do dispositivo foi intuitivo?”. Assim como na segunda pergunta, o *feedback* obtido foi positivo, com 62,5% (5) das respostas como concordo totalmente, 25% (2) em concordo em partes e 12,5% (1) em não concordo nem discordo. Apontando que mais 87,5% dos participantes acha o dispositivo de fácil uso em algum nível (APÊNDICE F, F-3). Aqui é possível inferir que o dispositivo apresenta, em algum nível, uma intuitividade associada, o que o torna o processo de aprendizado de sua manipulação, trivial.

A quarta pergunta visa compreender se o contexto de utilização do dispositivo, como meio auxiliar na sala de aula, pode ter um impacto positivo do processo de aprendizagem, através do questionamento: “Eu acredito que a utilização de recursos auxiliares na sala de aula pode desempenhar um fator de aumento do interesse e participação dos alunos?”. As respostas apontam unanimidade para concordância em algum nível com a afirmação. Com concordo totalmente, com 87,5 %, 7 de 8, dos participantes e o remanente concordando em partes (APÊNDICE F, F-4). Há uma concordância entre os usuários acerca da utilização de recursos auxiliares em sala de aula, assim podemos inferir que para os participantes o uso do dispositivo pode vir a ser algo benéfico ao processo de aprendizagem.

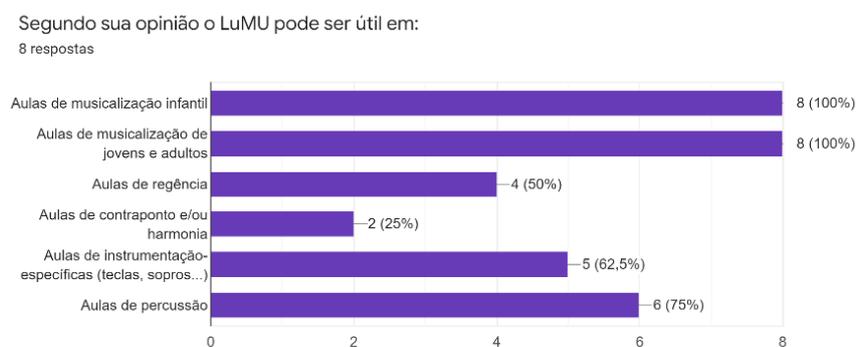
Ainda no contexto do impacto em sala de aula, a quinta pergunta visa compreender se a possibilidade de integração de sons melódicos e percussivos em sala de aula, poderia ser um fator positivo no aprendizado, através do seguinte questionamento: “A possibilidade de execução de sons percussivos e melódicos em um mesmo dispositivo é

um fator positivo para o aprendizado musical?”. Semelhantemente a quarta pergunta as respostas apontam para uma concordância positiva na afirmação. Com concordo totalmente, com 87,5 % (APÊNDICE F, F-5).

A sexta pergunta aborda a mobilidade e praticidade de traslado do equipamento e material auxiliar para a sala de aula, através do questionamento: “Levar o dispositivo (LuMU + Computador + Caixa de som) para sala de aula é algo impeditivo para a utilização do dispositivo?”. As respostas tenderam para um declínio da afirmação, com 75% (6) discordando totalmente da afirmação e 25% (2) discordando em partes (APÊNDICE F, F-6).

Divergindo das questões anteriores, a sétima pergunta, de múltipla escolha, visa compreender em que cenários, no contexto de sala de aula, os participantes acreditam que o dispositivo teria uma boa empregabilidade. Através do seguinte questionamento: “Segundo sua opinião o LuMU pode ser útil em:”. Os resultados apontam uma aplicabilidade diversificada em sala de aula do dispositivo, com unanimidade de aplicação para aulas de musicalização infantil e de jovens e adultos. O resultado deste item está apresentado na Figura 16 e no Apêndice F, F-7.

Figura 16: Respostas da pergunta 7



Fonte: O autor.

A oitava pergunta, de caráter aberto e de resposta não obrigatória, permitiu aos participantes a possibilidade de realizar sugestões acerca do desenvolvimento futuro do dispositivo. Dentre os 8 participantes apenas 2 realizaram sugestões:

Criar um aplicativo para celular ao invés do uso com o computador. (APÊNDICE F, F-8-A);

Que a luva possa ser vestida e retirada mais facilmente.

(APÊNDICE F, F-8-B);

O item F-8-A é de relevância real a este projeto. Ao introduzir a utilização de um *smartphone* (celular) como dispositivo de controle, traremos o dispositivo a um patamar mais relevante no que tange aos conceitos de mobilidade. Essa possibilidade será explorada em trabalhos futuros. Referente ao item F-8-B, a sugestão representa um cenário limitante do protótipo. Por ter o desenvolvimento em estado de protótipo, a luva escolhida para fixação dos componentes do projeto, foi adquirida em tamanho único. Contudo a sugestão traz um fator a ser considerado, como a utilização de um aparato têxtil mais flexível para o dispositivo final.

7 - CONCLUSÃO

O processo de popularização de utilização de recursos tecnológicos em sala de aula vem coexistindo com o progresso metodológico vivenciado no contexto pedagógico da música e artes, motivado pela crescente escalada da integralidade de dispositivos comuns as redes de computadores. Quais quer dispositivos que, neste contexto, desponhem como ferramentas auxiliares em sala de aula, podem vir a ser mais um acessório a serem utilizados por educadores.

É de suma importância destacar o papel desses dispositivos, não como pontos focais no processo de aprendizado, mas como acessórios a, primeiramente, facilitarem ao educador ao alcance ao interesse do aluno, e em segundo plano, como recurso propriamente ativo no processo de ensino. Paralelamente, a introdução de um recurso midiático, economicamente acessível, a esse leque de ferramentas é algo bem-acolhido pela necessidade do magistério em si, seja ele a um nível artístico, médio, técnico ou superior.

Este trabalho buscou apresentar, através de sua conceitualização, prototipação e testes com usuários, o desenvolvimento do LuMU, um dispositivo controlador MIDI como ferramenta auxiliar, para uso em sala de aula no ensino de música. Tal desenvolvimento abordou a concepção em dois níveis distintos, o físico, referente ao *hardware* e o lógico, referente ao *software*.

No que é tangível ao hardware, como resultado do uso plataformas de código aberto para sua prototipação, o custo de produção em demanda, de uma versão final do dispositivo, para distribuição entre docentes, seria barateado. Desta forma, também se evitaria processos licitatórios, necessários ao se lidar com *hardware* proprietário. As atuais especificações de *hardware* do chip Espressif ESP32 atendem aos requisitos necessários para uma implementação mais massificada do dispositivo, contudo a utilização deste chip vinculada a uma placa de desenvolvimento deve ser comutada, para a utilização em uma placa desenvolvida exclusivamente para o proposito deste projeto, afim de diminuir o espaço utilizado no material têxtil e, mais uma vez, baratear a produção.

Já a nível lógico, a também utilização de ferramentas de código aberto para o desenvolvimento do *software* embarcado é positivamente encorajado, devido a real possibilidade de contribuições para melhorias futuras na implementação, realizada por terceiros que venham a se engajar em algum nível no projeto. Já a utilização de *software* proprietário a nível de usuário, isto é, na utilização de DAWs não gratuitos, embora não

encorajada, é viável caso o usuário disponha de uma licença. Sobre o uso de DAWs gratuitos, não apenas é possível, como recomendado, tendo em vista que o processo de configuração e uso do dispositivo com essas ferramentas, é semelhante.

Ainda sobre as possibilidades relacionadas ao *software*, uma sugestão apresentada no processo de teste com usuários, trouxe à tona uma oportunidade fundamentada de ampliação da mobilidade do dispositivo para o usuário final. Através do desenvolvimento futuro de uma aplicação para *smartphones* que substitua a utilização de um notebook e *software* DAW. Tal cenário deverá ser adotado em trabalhos futuros, no desenvolvimento continuado deste projeto.

Esse projeto se iniciou com um levantamento teórico a cerca da utilização de dispositivos multimídia em sala de aula no contexto da docência de música, a seguir de uma revisão do estado da arte, a fim de compreender onde um dispositivo IoT vestível se encaixa atualmente nesse contexto pedagógico. Desta forma, este trabalho contribui para o incremento de novas estratégias de ensino. Outro tópico pertinente, é que por intermédio do detalhamento de produção do protótipo aqui descrito, quaisquer terceiros com acesso a este trabalho, podem vir a espelhar esse desenvolvimento e produzir artefatos semelhantes ou ainda melhores que os aqui descritos para empregabilidade em sala de aula.

Este trabalho realizou como parte da proposta de pesquisa a aplicação de teste de uso (teste de usuários), a fim de compreender se haviam fatores limitantes a empregabilidade do dispositivo em ações correlatas a pedagógica musical. Contudo, o artefato aqui produzido não se limita a utilização apenas em vias pedagógicas. O dispositivo é por si só um instrumento musical, e como tal, pode ser utilizado de maneira performática livre. Outro aspecto, agora pertinente a qualidade de dispositivo compatível com MIDI, é que o LuMU também poderá ser utilizado em contextos não musicais, podendo vir a ser ferramenta visual, como detalhado na produção do *patch* para Processing, ou elencado a quaisquer outros propósitos que utilizem a tecnologia MIDI e OSC, como na manipulação de interfaces 3D e controle de veículos não tripulados.

Dentro do escopo de futuras produções, há potencial para a descritividade de processos de refinamento do *hardware*, na produção de aparato (*hardware*) definitivo do projeto. Assim como no refinamento de *software*, também mencionado anteriormente. Futuramente, um estudo mais aprofundado, qualitativo e quantitativo, da real utilização em sala de aula do dispositivo e seus fatores ampliativos e limitantes, aventado pela parceria com instituições de ensino de música, em nível médio e/ou superior, se faz

necessário. Permitindo uma investigação mais robusta a respeito do emprego do artefato aqui descrito.

Finalmente, tais frentes de estudo, são idealizadas para um desenvolvimento continuado do DSR, sincronicamente, como frações de um projeto de doutorado. Esses processos descritivos tem como papel documentar e ampliar conhecimento do que pode ser desenvolvido, no sentido da concepção de novos aparatos em sala de aula, não apenas no contexto da música, arte e audiovisual, mas como ferramentas para uma contribuição social mais abrangente.

8 - REFERÊNCIAS

ALMASI, G.S. e A. GOTTLIEB (1989). **Highly Parallel Computing**. Benjamin-Cummings, Redwood City, CA.

BRASIL. **Lei nº 9.934, de 18 de agosto de 2008**. Altera a Lei no 9.394, de 20 de dezembro de 1996, Lei de Diretrizes e Bases da Educação, para dispor sobre a obrigatoriedade do ensino da música na educação básica. Disponível em: http://www.planalto.gov.br/ccivil_03/_ato2007-2010/2008/lei/111769.htm. Acesso em: 24 abr. 2021.

BRANDÃO, Lucas Lima (2020). **Suporte Ao Aprendizado De Música Baseado Em Gravações Usando Ferramentas De Código Aberto**.

BREWSTER, Stephen. "Nonspeech Auditory Output". **The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies, and Emerging Applications**. Ed. Julie A. Jacko; Andrew Sears. Mahwah: Lawrence Erlbaum Associates, 2003. p.227

CARNEIRO, Fabio Gomes. **PERCEPÇÃO DO PITCH FANTASMA UTILIZANDO A SIRENE DE SEEBECK**. Universidade Estadual de Campinas - UNICAMP

COTRIM, Ricardo Murtinho Braga. **Práticas Pedagógicas Criativas Musicais em Ambiente de Estúdio Eletroacústico: Experiência e Polissonia em sala de aula**. 2020. Disponível em: <<http://www.repositorio-bc.unirio.br:8080/xmlui/handle/unirio/13185>>. Acesso em: 18 de jul. 2021.

FACCIONI FILHO, Mauro. **Internet das coisas**. Unisul Virtual, 2016.

FRITSCH, Eloy. **Música Eletrônica: uma introdução ilustrada**. Porto Alegre: Editora da UFRGS, 2008.

FRÓES, Luiza Ten Cate; ESPUNY, Ana Lúcia Gil. **Comportamento E Percepção De Valor Dos Usuários De Dispositivos Vestíveis**: um estudo de caso de usuários de Xiaomi Mi Band.

DA NÓBREGA, Rafael Moura Toscano, 2019. **Framework Para Auxiliar Terapeutas No Design De Recursos Audiovisuais Para Ensinar Competências E Habilidades A Crianças Autistas**

DA SILVA NÓBREGA, José Cândido; MAIA, Torben Fernandes. **As tecnologias disruptivas na educação**. Informativo Técnico do Semiárido, v. 14, n. 2, p. 202-205, 2020.

DE ALMEIDA, Isabela Corintha. **Aprendizagem Musical Através de Composição Assistida por Computador**. 2018. Disponível em: < <https://repositorio-aberto.up.pt/handle/10216/114266>>. Acesso em: 18 de jul. 2021.

DEMO, Pedro. **Rupturas urgentes em educação**. Ensaio: avaliação e políticas públicas em educação, v. 18, p. 861-871, 2010.

DIAN, F. John; VAHIDNIA, Reza; RAHMATI, Alireza. **Wearables and the Internet of Things (IoT), applications, opportunities, and challenges: A Survey**. IEEE Access, v. 8, p. 69200-69211, 2020.

JOHANN FILHO, Sergio; PONTES, Julian; LEITHARDT, Valderi. **Multiprocessor System on a Chip**. PUCRS.

JOSSO, Marie Christine. **Experiências de vida e formação**. São Paulo: Cortez, 2004.

MENEZES, Angélica Andrade; KAYAMA, Adriana Giarola. Performance (e performatividade) musical: das experiências às reflexões. In: **XXIX Congresso da Anppom-Pelotas/RS**. 2019.

HALVERSON, Erica Rosenfeld; SHERIDAN, Kimberly. **The Maker Movement in Education**. Harvard Educational Review. Cambridge, p. 495-504. dez. 2014. Disponível

em: <<https://www.hepgjournals.org/doi/abs/10.17763/haer.84.4.34j1g68140382063>>.
Acesso em: 22 ago. 2019.

HEVNER, Alan; CHATTERJEE, Samir. Design science research in information systems. In: **Design research in information systems**. Springer, Boston, MA, 2010. p. 9-22.

HYPERBITS. **Layering Music**: 20 ways to layer sounds. 20 Ways to Layer Sounds. Disponível em: <https://hyperbits.com/layering-sounds/>. Acesso em: 19 jun. 2022.

KELLER, Alexandre. Novatec, ed. **Asterisk na prática**. agosto de 2014 2 ed. São Paulo: [s.n.] 9788575222867

KLIPFEL, Kristofer. **MIDI Motion**: Interactive Music Composition Gloves. In: INTERNATIONAL CONFERENCE ON TANGIBLE, EMBEDDED, AND EMBODIED INTERACTION, 11., 2017, Yokohama. Proceedings... . Colorado: Acm, 2017. p. 757 - 760. Disponível em: <<https://dl.acm.org/citation.cfm?id=3035535>>.
Acesso em: 20 set. 2019.

LUNHANI, g. m., & MANNIS, j. a. **dispositivos físicos interativos aplicados à projeção sonora na música eletroacústica**.

MAIER, A., SHARP, A., & VAGAPOV, Y. (2017). **Comparative Analysis and Practical Implementation of The ESP32 Microcontroller Module For The Internet of Things**. 2017 Internet Technologies and Applications (ITA). doi:10.1109/itecha.2017.8101926

MARQUES, Daniel Sorrentino. **Glitch Art E Pure Data**: Utilizando pure data para transformar o processo de criação de glitch. 2018. 125 f. Dissertação (Mestrado) - Programa de Pós-Graduação em Computação, Comunicação e Artes, Centro de Informática, Universidade Federal da Paraíba, João Pessoa, 2018. Cap. 5. Disponível em: <https://repositorio.ufpb.br/jspui/handle/123456789/15310>. Acesso em: 29 dez. 2020.

MARTIN, Grant; CHANG, Henry (Ed.). **Winning the SoC revolution**: experiences in real design. Springer Science & Business Media, 2012.

MULTIMEDIA TECHNOLOGY, 2011, Taipei. Proceedings... Hangzhou: Ieee, 2011. p. 6179 - 6182. Disponível em: <<https://ieeexplore.ieee.org/abstract/document/6002156/authors#authors>>. Acesso em: 22 ago. 2019.

OECD. **Education at a Glance 2019**: OECD Indicators. OECD, 2019. DOI.org (Crossref), doi:10.1787/f8d7880d-en.

PEACH, Martin. **Mr.Peach**. Disponível em: <https://puredata.info/downloads/mrpeach>. Acesso em: 03 jan. 2021.

PHILLIPS, Dave. **An Introduction To OSC**. 2008. Disponível em: <<https://www.linuxjournal.com/content/introduction-osc>>. Acesso em: 21 out. 2019.

PORCARO, N. and LEVY, E., 2009. **Blendnik: A Real-Time Performance System Using Blender and Pure Data**. In: 3rd PD International Convention in Sao Paolo 2009. [online] Sao Paolo. Available at: <<http://puredata.info/community/conventions/convention09/porcaro.pdf/view>> [Accessed 30 December 2020].

RAJSUMAN, Rochit. Introduction. In: RAJSUMAN, Rochit. **System-on-a-Chip: Design and Test**. Norwood: Artech House, 2000. Cap. 1, p. 3.

RATTON, Miguel. **Novas Tecnologias Aplicadas À Música**. 2006. Disponível em: <<http://tiny.cc/rattom>>. Acesso em: 19 ago. 2019.

RATTON, Miguel. **Dicionário de Áudio e Tecnologia Musical**. 2. ed. Rio de Janeiro: Música e Tecnologia, 2009.

SANTAELLA, Lucia et al. **Desvelando a Internet das coisas**. Revista GEMInIS, v. 4, n. 2, p. 19-32, 2013.

SARANG, Prathamesh et al. **Air Drum Using Kinect and Arduino**. International Journal Of Computer Science And Information Technologies. Mumbai, p. 1153-1155. 2015. Disponível em:

<<https://pdfs.semanticscholar.org/2c7f/fe97e7123f3278f383b8a798b6625dd5ac1b.pdf>>.

Acesso em: 17 set. 2019.

SCHLEGEL, Andreas. **oscP5-a implementation of the OSC protocol for Processing**. 2011.

SILVA, Valéria Carvalho da. **Corporeidade e educação: sinfonia de saberes na educação musical**. 2008.

SILVEIRA, Jairo Perin. **Orquestra de Papel: Uma Ação Pedagógica Envolvendo Música e Tecnologia**. NICS Reports, n. 6, p. 01-10, 2014. Disponível em: <<https://revistas.nics.unicamp.br/revistas/ojs/index.php/nr/article/view/90>>. Acesso em: 18 de jul. 2021.

STRIEBEL, K., SHIM, J., SRINIVASAN, V., & NEWMAN, J. (2005). **Comparison of LiFePO₄ from Different Sources**. Journal of The Electrochemical Society, 152(4), A664. doi:10.1149/1.1862477

Reas, C., & Fry, B. (2006). **Processing: programming for the media arts**. AI & SOCIETY, 20(4), 526–538. doi:10.1007/s00146-006-0050-9

TSUNG-CHING, Liu; SHU-HUI, Chang; CHE-YI, Hsiao. **A modified Quad-Theremin for interactive computer music control**. In: INTERNATIONAL CONFERENCE ON MULTIMEDIA TECHNOLOGY. IEEE, 2011. p. 6179-6182.

APÊNDICES

APÊNDICE A – CÓDIGO FONTE EM ANSI C

A-1 MÃO DIREITA

```
#include <WiFi.h>
#include <WiFiUdp.h>
#include <WiFiClient.h>
#include <OSCMessages.h>

char ssid[] = "LUMU";    // your network SSID (name)
char pass[] = "keypasslumu";    // your network password

WiFiUDP Udp;
const IPAddress outIp(192, 168, 4, 2);
const unsigned int outPort = 8333;
const unsigned int localPort = 8000;

//indicador
bool estadoiz1 = false;
bool estadoiz2 = false;
bool estadoiz3 = false;
bool estadoiz4 = false;

//medio
bool estadomz1 = false;
bool estadomz2 = false;
bool estadomz3 = false;
bool estadomz4 = false;

//anelar
bool estadoaz1 = false;
bool estadoaz2 = false;
bool estadoaz3 = false;
bool estadoaz4 = false;

//minimo
bool estadomiz1 = false;
```

```

bool estadomiz2 = false;
bool estadomiz3 = false;
bool estadomiz4 = false;
/*
//membrana
bool estadome1 = false;
bool estadome2 = false;
bool estadome3 = false;
*/

void setup() {

//indicador
pinMode(26, INPUT_PULLUP);
pinMode(14, INPUT_PULLUP);
pinMode(25, INPUT_PULLUP);
pinMode(33, INPUT_PULLUP);

//medio
pinMode(15, INPUT_PULLUP);
pinMode(2, INPUT_PULLUP);
pinMode(27, INPUT_PULLUP);
pinMode(13, INPUT_PULLUP);

//anelar
pinMode(32, INPUT_PULLUP);
pinMode(12, INPUT_PULLUP);
pinMode(4, INPUT_PULLUP);
pinMode(16, INPUT_PULLUP);

//minimo
pinMode(5, INPUT_PULLUP);
pinMode(18, INPUT_PULLUP);
pinMode(17, INPUT_PULLUP);
pinMode(21, INPUT_PULLUP);
/*
//membrana
pinMode(39, INPUT_PULLUP);
pinMode(36, INPUT_PULLUP);

```

```

pinMode(34, INPUT_PULLUP);
*/
Serial.begin(115200);

// Connect to WiFi network
Serial.println(ssid);
WiFi.begin(ssid, pass);

while (WiFi.status() != WL_CONNECTED) {
  delay(500);
  Serial.println("Connecting to WiFi..");
}

Serial.println("");

Serial.println("WiFi conectado");
IPAddress myIP = WiFi.localIP();
Serial.print("Endereço de IP: ");
Serial.println(myIP);

Serial.println("Iniciando UDP");
Udp.begin(localPort);
Serial.print("Porta local: ");
Serial.println(localPort);
}

void loop() {
  // Indicador
  OSCMessage msgiz1("/Direita/Indicador/z1");
  OSCMessage msgiz2("/Direita/Indicador/z2");
  OSCMessage msgiz3("/Direita/Indicador/z3");
  OSCMessage msgiz4("/Direita/Indicador/z4");

  //indicador Z1
  if (digitalRead(26) == 0 && estadoiz1 == false) {
    msgiz1.add(55);
    msgiz1.add(127);
    Udp.beginPacket(outIp, outPort);
    msgiz1.send(Udp);
  }
}

```

```

    Udp.endPacket();
    msgiz1.empty();
    estadoiz1 = true;
}
if (digitalRead(26) == 1 && estadoiz1 == true) {
    msgiz1.add(55);
    msgiz1.add(0);
    Udp.beginPacket(outIp, outPort);
    msgiz1.send(Udp);
    Udp.endPacket();
    estadoiz1 = false;
}

//indicador Z2
if (digitalRead(14) == 0 && estadoiz2 == false) {
    msgiz2.add(56);
    msgiz2.add(127);
    Udp.beginPacket(outIp, outPort);
    msgiz2.send(Udp);
    Udp.endPacket();
    msgiz2.empty();
    estadoiz2 = true;
}
if (digitalRead(14) == 1 && estadoiz2 == true) {
    msgiz2.add(56);
    msgiz2.add(0);
    Udp.beginPacket(outIp, outPort);
    msgiz2.send(Udp);
    Udp.endPacket();
    estadoiz2 = false;
}

//indicador Z3
if (digitalRead(25) == 0 && estadoiz3 == false) {
    msgiz3.add(67);
    msgiz3.add(127);
    Udp.beginPacket(outIp, outPort);

```

```

    msgiz3.send(Udp);
    Udp.endPacket();
    msgiz3.empty();
    estadoiz3 = true;
}
if (digitalRead(25) == 1 && estadoiz3 == true) {
    msgiz3.add(67);
    msgiz3.add(0);
    Udp.beginPacket(outIp, outPort);
    msgiz3.send(Udp);
    Udp.endPacket();
    estadoiz3 = false;
}

//indicador Z4
if (digitalRead(33) == 0 && estadoiz4 == false) {
    msgiz4.add(68);
    msgiz4.add(127);
    Udp.beginPacket(outIp, outPort);
    msgiz4.send(Udp);
    Udp.endPacket();
    msgiz4.empty();
    estadoiz4 = true;
}
if (digitalRead(33) == 1 && estadoiz4 == true) {
    msgiz4.add(68);
    msgiz4.add(0);
    Udp.beginPacket(outIp, outPort);
    msgiz4.send(Udp);
    Udp.endPacket();
    estadoiz4 = false;
}

//Medio
OSCMessage msgmz1("/Direita/Medio/z1");
OSCMessage msgmz2("/Direita/Medio/z2");
OSCMessage msgmz3("/Direita/Medio/z3");

```

```

OSCMMessage msgmz4("/Direita/Medio/z4");

//medio Z1
if (digitalRead(15) == 0 && estadomz1 == false) {
    msgmz1.add(57);
    msgmz1.add(127);
    Udp.beginPacket(outIp, outPort);
    msgmz1.send(Udp);
    Udp.endPacket();
    msgmz1.empty();
    estadomz1 = true;
}
if (digitalRead(15) == 1 && estadomz1 == true) {
    msgmz1.add(57);
    msgmz1.add(0);
    Udp.beginPacket(outIp, outPort);
    msgmz1.send(Udp);
    Udp.endPacket();
    estadomz1 = false;
}

//medio Z2
if (digitalRead(2) == 0 && estadomz2 == false) {
    msgmz2.add(58);
    msgmz2.add(127);
    Udp.beginPacket(outIp, outPort);
    msgmz2.send(Udp);
    Udp.endPacket();
    msgmz2.empty();
    estadomz2 = true;
}
if (digitalRead(2) == 1 && estadomz2 == true) {
    msgmz2.add(58);
    msgmz2.add(0);
    Udp.beginPacket(outIp, outPort);
    msgmz2.send(Udp);
    Udp.endPacket();
    estadomz2 = false;
}

```

```

}

//medio Z3
if (digitalRead(27) == 0 && estadamz3 == false) {
    msgmz3.add(69);
    msgmz3.add(127);
    Udp.beginPacket(outIp, outPort);
    msgmz3.send(Udp);
    Udp.endPacket();
    msgmz3.empty();
    estadamz3 = true;
}
if (digitalRead(27) == 1 && estadamz3 == true) {
    msgmz3.add(69);
    msgmz3.add(0);
    Udp.beginPacket(outIp, outPort);
    msgmz3.send(Udp);
    Udp.endPacket();
    estadamz3 = false;
}

//medio Z4
if (digitalRead(13) == 0 && estadamz4 == false) {
    msgmz4.add(70);
    msgmz4.add(127);
    Udp.beginPacket(outIp, outPort);
    msgmz4.send(Udp);
    Udp.endPacket();
    msgmz4.empty();
    estadamz4 = true;
}
if (digitalRead(13) == 1 && estadamz4 == true) {
    msgmz4.add(70);
    msgmz4.add(0);
    Udp.beginPacket(outIp, outPort);
    msgmz4.send(Udp);
    Udp.endPacket();
    estadamz4 = false;
}

```

```
}
```

```
//anelar
```

```
OSCMMessage msgaz1("/Direita/Anelar/z1");  
OSCMMessage msgaz2("/Direita/Anelar/z2");  
OSCMMessage msgaz3("/Direita/Anelar/z3");  
OSCMMessage msgaz4("/Direita/Anelar/z4");
```

```
//anelar Z1
```

```
if (digitalRead(32) == 0 && estadoaz1 == false) {  
    msgaz1.add(59);  
    msgaz1.add(127);  
    Udp.beginPacket(outIp, outPort);  
    msgaz1.send(Udp);  
    Udp.endPacket();  
    msgaz1.empty();  
    estadoaz1 = true;  
}
```

```
if (digitalRead(32) == 1 && estadoaz1 == true) {  
    msgaz1.add(59);  
    msgaz1.add(0);  
    Udp.beginPacket(outIp, outPort);  
    msgaz1.send(Udp);  
    Udp.endPacket();  
    estadoaz1 = false;  
}
```

```
//anelar Z2
```

```
if (digitalRead(12) == 0 && estadoaz2 == false) {  
    msgaz2.add(74);  
    msgaz2.add(127);  
    Udp.beginPacket(outIp, outPort);  
    msgaz2.send(Udp);  
    Udp.endPacket();  
    msgaz2.empty();
```

```

    estadoaz2 = true;
}
if (digitalRead(12) == 1 && estadoaz2 == true) {
    msgaz2.add(74);
    msgaz2.add(0);
    Udp.beginPacket(outIp, outPort);
    msgaz2.send(Udp);
    Udp.endPacket();
    estadoaz2 = false;
}

//anelar Z3
if (digitalRead(4) == 0 && estadoaz3 == false) {
    msgaz3.add(71);
    msgaz3.add(127);
    Udp.beginPacket(outIp, outPort);
    msgaz3.send(Udp);
    Udp.endPacket();
    msgaz3.empty();
    estadoaz3 = true;
}
if (digitalRead(4) == 1 && estadoaz3 == true) {
    msgaz3.add(71);
    msgaz3.add(0);
    Udp.beginPacket(outIp, outPort);
    msgaz3.send(Udp);
    Udp.endPacket();
    estadoaz3 = false;
}

//anelar Z4
if (digitalRead(16) == 0 && estadoaz4 == false) {
    msgaz4.add(75);
    msgaz4.add(127);
    Udp.beginPacket(outIp, outPort);
    msgaz4.send(Udp);
    Udp.endPacket();
    msgaz4.empty();
    estadoaz4 = true;
}

```

```

}
if (digitalRead(16) == 1 && estadoaz4 == true) {
    msgaz4.add(75);
    msgaz4.add(0);
    Udp.beginPacket(outIp, outPort);
    msgaz4.send(Udp);
    Udp.endPacket();
    estadoaz4 = false;
}

```

```

OSCMMessage msgmiz1("/Direita/Minimo/z1");
OSCMMessage msgmiz2("/Direita/Minimo/z2");
OSCMMessage msgmiz3("/Direita/Minimo/z3");
OSCMMessage msgmiz4("/Direita/Minimo/z4");

```

```

//Minino Z1

```

```

if (digitalRead(5) == 0 && estadomiz1 == false) {
    msgmiz1.add(60);
    msgmiz1.add(127);
    Udp.beginPacket(outIp, outPort);
    msgmiz1.send(Udp);
    Udp.endPacket();
    msgmiz1.empty();
    estadomiz1 = true;
}

```

```

if (digitalRead(5) == 1 && estadomiz1 == true) {
    msgmiz1.add(60);
    msgmiz1.add(0);
    Udp.beginPacket(outIp, outPort);
    msgmiz1.send(Udp);
    Udp.endPacket();
    estadomiz1 = false;
}

```

```

//Minino Z2

```

```

if (digitalRead(18) == 0 && estadomiz2 == false) {
    msgmiz2.add(61);
}

```

```

msgmiz2.add(127);
Udp.beginPacket(outIp, outPort);
msgmiz2.send(Udp);
Udp.endPacket();
msgmiz2.empty();
estadomiz2 = true;
}
if (digitalRead(18) == 1 && estadomiz2 == true) {
    msgmiz2.add(61);
    msgmiz2.add(0);
    Udp.beginPacket(outIp, outPort);
    msgmiz2.send(Udp);
    Udp.endPacket();
    estadomiz2 = false;
}

//Minino Z3
if (digitalRead(17) == 0 && estadomiz3 == false) {
    msgmiz3.add(76);
    msgmiz3.add(127);
    Udp.beginPacket(outIp, outPort);
    msgmiz3.send(Udp);
    Udp.endPacket();
    msgmiz3.empty();
    estadomiz3 = true;
}
if (digitalRead(17) == 1 && estadomiz3 == true) {
    msgmiz3.add(76);
    msgmiz3.add(0);
    Udp.beginPacket(outIp, outPort);
    msgmiz3.send(Udp);
    Udp.endPacket();
    estadomiz3 = false;
}

//Minino Z4
if (digitalRead(21) == 0 && estadomiz4 == false) {

```

```

msgmiz4.add(77);
msgmiz4.add(127);
Udp.beginPacket(outIp, outPort);
msgmiz4.send(Udp);
Udp.endPacket();
msgmiz4.empty();
estadomiz4 = true;
}
if (digitalRead(21) == 1 && estadomiz4 == true) {
    msgmiz4.add(77);
    msgmiz4.add(0);
    Udp.beginPacket(outIp, outPort);
    msgmiz4.send(Udp);
    Udp.endPacket();
    estadomiz4 = false;
}

```

```
/*
```

```

OSCMMessage membrana1("/Membrana/1");
OSCMMessage membrana2("/Membrana/2");
OSCMMessage membrana3("/Membrana/3");

```

```
//Membrana 1
```

```

if (digitalRead(39) == 0 && estadome1 == false) {
    membrana1.add(0);
    membrana1.add(0);
    membrana1.add(64);
    membrana1.add(127);
    Udp.beginPacket(outIp, outPort);
    membrana1.send(Udp);
    Udp.endPacket();
    membrana1.empty();
    estadome1 = true;
}
if (digitalRead(39) == 1 && estadome1 == true) {
    membrana1.add(0);
    membrana1.add(0);
    membrana1.add(64);

```

```

membrana1.add(0);
Udp.beginPacket(outIp, outPort);
membrana1.send(Udp);
Udp.endPacket();
estadome1 = false;
}

//Membrana 2
if (digitalRead(36) == 0 && estadome2 == false) {
  membrana2.add(0);
  membrana2.add(0);
  membrana2.add(65);
  membrana2.add(127);
  Udp.beginPacket(outIp, outPort);
  membrana2.send(Udp);
  Udp.endPacket();
  membrana2.empty();
  estadome2 = true;
}
if (digitalRead(36) == 1 && estadome2 == true) {
  membrana2.add(0);
  membrana2.add(0);
  membrana2.add(65);
  membrana2.add(0);
  Udp.beginPacket(outIp, outPort);
  membrana2.send(Udp);
  Udp.endPacket();
  estadome2 = false;
}

//membrana 3
if (digitalRead(34) == 0 && estadome3 == false) {
  membrana3.add(0);
  membrana3.add(0);
  membrana3.add(66);
  membrana3.add(127);
  Udp.beginPacket(outIp, outPort);
  membrana3.send(Udp);
  Udp.endPacket();
}

```

```
membrana3.empty();
estadome3 = true;
}
if (digitalRead(34) == 1 && estadome3 == true) {
  membrana3.add(0);
  membrana3.add(0);
  membrana3.add(66);
  membrana3.add(0);
  Udp.beginPacket(outIp, outPort);
  membrana3.send(Udp);
  Udp.endPacket();
  estadome3 = false;
}
*/
}
```

A-2 MÃO ESQUERDA

```
#include <WiFi.h>
#include <WiFiUdp.h>
#include <WiFiClient.h>
#include <OSCMessages.h>

char ssid[] = "LUMU";    // your network SSID (name)
char pass[] = "keypasslumu";    // your network password

WiFiUDP Udp;
const IPAddress outIp(192, 168, 4, 2);
const unsigned int outPort = 8333;
const unsigned int localPort = 8000;

//indicador
bool estadoiz1 = false;
bool estadoiz2 = false;
bool estadoiz3 = false;
bool estadoiz4 = false;

//medio
bool estadamz1 = false;
bool estadamz2 = false;
bool estadamz3 = false;
bool estadamz4 = false;

//anelar
bool estadoaz1 = false;
bool estadoaz2 = false;
bool estadoaz3 = false;
bool estadoaz4 = false;

//minimo
bool estadomiz1 = false;
bool estadomiz2 = false;
bool estadomiz3 = false;
bool estadomiz4 = false;
```

```

void setup() {

  //indicador
  pinMode(23, INPUT_PULLUP);
  pinMode(22, INPUT_PULLUP);
  pinMode(21, INPUT_PULLUP);
  pinMode(19, INPUT_PULLUP);

  //medio
  pinMode(18, INPUT_PULLUP);
  pinMode(5, INPUT_PULLUP);
  pinMode(17, INPUT_PULLUP);
  pinMode(16, INPUT_PULLUP);

  //anelar
  pinMode(32, INPUT_PULLUP);
  pinMode(14, INPUT_PULLUP);
  pinMode(15, INPUT_PULLUP);
  pinMode(27, INPUT_PULLUP);

  //minimo
  pinMode(12, INPUT_PULLUP);
  pinMode(13, INPUT_PULLUP);
  pinMode(4, INPUT_PULLUP);
  pinMode(2, INPUT_PULLUP);

  Serial.begin(115200);

  // Connect to WiFi network
  Serial.println();
  Serial.println();
  Serial.print("Conectando: ");
  Serial.println(ssid);
  WiFi.softAP(ssid, pass, 1, false, 2);

  Serial.println("");

  Serial.println("WiFi conectado");
}

```

```

IPAddress myIP = WiFi.softAPIP();
Serial.print("Endereço de IP: ");
Serial.println(myIP);

Serial.println("Iniciando UDP");
Udp.begin(localPort);
Serial.print("Porta local: ");
Serial.println(localPort);

}

void loop() {
  // Indicador
  OSCMessage msgiz1("/Esquerda/Indicador/z1");
  OSCMessage msgiz2("/Esquerda/Indicador/z2");
  OSCMessage msgiz3("/Esquerda/Indicador/z3");
  OSCMessage msgiz4("/Esquerda/Indicador/z4");

  //indicador Z1
  if (digitalRead(23) == 0 && estadoiz1 == false) {
    msgiz1.add(36);
    msgiz1.add(127);
    Udp.beginPacket(outIp, outPort);
    msgiz1.send(Udp);
    Udp.endPacket();
    msgiz1.empty();
    estadoiz1 = true;
  }
  if (digitalRead(23) == 1 && estadoiz1 == true) {
    msgiz1.add(36);
    msgiz1.add(0);
    Udp.beginPacket(outIp, outPort);
    msgiz1.send(Udp);
    Udp.endPacket();
    estadoiz1 = false;
  }
}

```

```
//indicador Z2
if (digitalRead(22) == 0 && estadoiz2 == false) {
    msgiz2.add(37);
    msgiz2.add(127);
    Udp.beginPacket(outIp, outPort);
    msgiz2.send(Udp);
    Udp.endPacket();
    msgiz2.empty();
    estadoiz2 = true;
}
if (digitalRead(22) == 1 && estadoiz2 == true) {
    msgiz2.add(37);
    msgiz2.add(0);
    Udp.beginPacket(outIp, outPort);
    msgiz2.send(Udp);
    Udp.endPacket();
    estadoiz2 = false;
}
```

```
//indicador Z3
if (digitalRead(19) == 0 && estadoiz3 == false) {
    msgiz3.add(48);
    msgiz3.add(127);
    Udp.beginPacket(outIp, outPort);
    msgiz3.send(Udp);
    Udp.endPacket();
    msgiz3.empty();
    estadoiz3 = true;
}
if (digitalRead(19) == 1 && estadoiz3 == true) {
    msgiz3.add(48);
    msgiz3.add(0);
    Udp.beginPacket(outIp, outPort);
    msgiz3.send(Udp);
    Udp.endPacket();
    estadoiz3 = false;
}
```

```

//indicador Z4
if (digitalRead(21) == 0 && estadoiz4 == false) {
    msgiz4.add(49);
    msgiz4.add(127);
    Udp.beginPacket(outIp, outPort);
    msgiz4.send(Udp);
    Udp.endPacket();
    msgiz4.empty();
    estadoiz4 = true;
}
if (digitalRead(21) == 1 && estadoiz4 == true) {
    msgiz4.add(49);
    msgiz4.add(0);
    Udp.beginPacket(outIp, outPort);
    msgiz4.send(Udp);
    Udp.endPacket();
    estadoiz4 = false;
}

```

```

//Medio

```

```

OSCMMessage msgmz1("/Esquerda/Medio/z1");
OSCMMessage msgmz2("/Esquerda/Medio/z2");
OSCMMessage msgmz3("/Esquerda/Medio/z3");
OSCMMessage msgmz4("/Esquerda/Medio/z4");

```

```

//medio Z1

```

```

if (digitalRead(18) == 0 && estadomz1 == false) {
    msgmz1.add(38);
    msgmz1.add(127);
    Udp.beginPacket(outIp, outPort);
    msgmz1.send(Udp);
    Udp.endPacket();
    msgmz1.empty();
    estadomz1 = true;
}

```

```
}  
if (digitalRead(18) == 1 && estادمz1 == true) {  
    msgmz1.add(38);  
    msgmz1.add(0);  
    Udp.beginPacket(outIp, outPort);  
    msgmz1.send(Udp);  
    Udp.endPacket();  
    estادمz1 = false;  
}
```

```
//medio Z2
```

```
if (digitalRead(17) == 0 && estادمz2 == false) {  
    msgmz2.add(39);  
    msgmz2.add(127);  
    Udp.beginPacket(outIp, outPort);  
    msgmz2.send(Udp);  
    Udp.endPacket();  
    msgmz2.empty();  
    estادمz2 = true;  
}
```

```
if (digitalRead(17) == 1 && estادمz2 == true) {  
    msgmz2.add(39);  
    msgmz2.add(0);  
    Udp.beginPacket(outIp, outPort);  
    msgmz2.send(Udp);  
    Udp.endPacket();  
    estادمz2 = false;  
}
```

```
//medio Z3
```

```
if (digitalRead(5) == 0 && estادمz3 == false) {  
    msgmz3.add(50);  
    msgmz3.add(127);  
    Udp.beginPacket(outIp, outPort);  
    msgmz3.send(Udp);  
    Udp.endPacket();  
    msgmz3.empty();  
    estادمz3 = true;
```

```

}
if (digitalRead(5) == 1 && estادمz3 == true) {
    msgmz3.add(50);
    msgmz3.add(0);
    Udp.beginPacket(outIp, outPort);
    msgmz3.send(Udp);
    Udp.endPacket();
    estادمz3 = false;
}

```

```

//medio Z4
if (digitalRead(16) == 0 && estادمz4 == false) {
    msgmz4.add(51);
    msgmz4.add(127);
    Udp.beginPacket(outIp, outPort);
    msgmz4.send(Udp);
    Udp.endPacket();
    msgmz4.empty();
    estادمz4 = true;
}
if (digitalRead(16) == 1 && estادمz4 == true) {
    msgmz4.add(51);
    msgmz4.add(0);
    Udp.beginPacket(outIp, outPort);
    msgmz4.send(Udp);
    Udp.endPacket();
    estادمz4 = false;
}

```

```

//Medio
OSCMessage msgaz1("/Esquerda/Anelar/z1");
OSCMessage msgaz2("/Esquerda/Anelar/z2");
OSCMessage msgaz3("/Esquerda/Anelar/z3");
OSCMessage msgaz4("/Esquerda/Anelar/z4");

```

```

//anelar Z1
if (digitalRead(15) == 0 && estadoaz1 == false) {
    msgaz1.add(40);
    msgaz1.add(127);
    Udp.beginPacket(outIp, outPort);
    msgaz1.send(Udp);
    Udp.endPacket();
    msgaz1.empty();
    estadoaz1 = true;
}
if (digitalRead(15) == 1 && estadoaz1 == true) {
    msgaz1.add(40);
    msgaz1.add(0);
    Udp.beginPacket(outIp, outPort);
    msgaz1.send(Udp);
    Udp.endPacket();
    estadoaz1 = false;
}

//anelar Z2
if (digitalRead(32) == 0 && estadoaz2 == false) {
    msgaz2.add(72);
    msgaz2.add(127);
    Udp.beginPacket(outIp, outPort);
    msgaz2.send(Udp);
    Udp.endPacket();
    msgaz2.empty();
    estadoaz2 = true;
}
if (digitalRead(32) == 1 && estadoaz2 == true) {
    msgaz2.add(72);
    msgaz2.add(0);
    Udp.beginPacket(outIp, outPort);
    msgaz2.send(Udp);
    Udp.endPacket();
    estadoaz2 = false;
}

```

```

//anelar Z3
if (digitalRead(27) == 0 && estadoaz3 == false) {
    msgaz3.add(52);
    msgaz3.add(127);
    Udp.beginPacket(outIp, outPort);
    msgaz3.send(Udp);
    Udp.endPacket();
    msgaz3.empty();
    estadoaz3 = true;
}
if (digitalRead(27) == 1 && estadoaz3 == true) {
    msgaz3.add(52);
    msgaz3.add(0);
    Udp.beginPacket(outIp, outPort);
    msgaz3.send(Udp);
    Udp.endPacket();
    estadoaz3 = false;
}

```

```

//anelar Z4
if (digitalRead(14) == 0 && estadoaz4 == false) {
    msgaz4.add(73);
    msgaz4.add(127);
    Udp.beginPacket(outIp, outPort);
    msgaz4.send(Udp);
    Udp.endPacket();
    msgaz4.empty();
    estadoaz4 = true;
}
if (digitalRead(14) == 1 && estadoaz4 == true) {
    msgaz4.add(73);
    msgaz4.add(0);
    Udp.beginPacket(outIp, outPort);
    msgaz4.send(Udp);
    Udp.endPacket();
    estadoaz4 = false;
}

```

```
OSCMMessage msgmiz1("/Esquerda/Minimo/z1");
OSCMMessage msgmiz2("/Esquerda/Minimo/z2");
OSCMMessage msgmiz3("/Esquerda/Minimo/z3");
OSCMMessage msgmiz4("/Esquerda/Minimo/z4");
```

```
//Minino Z1
```

```
if (digitalRead(4) == 0 && estandomiz1 == false) {
    msgmiz1.add(53);
    msgmiz1.add(127);
    Udp.beginPacket(outIp, outPort);
    msgmiz1.send(Udp);
    Udp.endPacket();
    msgmiz1.empty();
    estandomiz1 = true;
}
```

```
if (digitalRead(4) == 1 && estandomiz1 == true) {
    msgmiz1.add(53);
    msgmiz1.add(0);
    Udp.beginPacket(outIp, outPort);
    msgmiz1.send(Udp);
    Udp.endPacket();
    estandomiz1 = false;
}
```

```
//Minino Z2
```

```
if (digitalRead(13) == 0 && estandomiz2 == false) {
    msgmiz2.add(54);
    msgmiz2.add(127);
    Udp.beginPacket(outIp, outPort);
    msgmiz2.send(Udp);
    Udp.endPacket();
    msgmiz2.empty();
    estandomiz2 = true;
}
```

```
if (digitalRead(13) == 1 && estandomiz2 == true) {
    msgmiz2.add(54);
    msgmiz2.add(0);
    Udp.beginPacket(outIp, outPort);
```

```
msgmiz2.send(Udp);
Udp.endPacket();
estadomiz2 = false;
}
```

```
//Minino Z3
```

```
if (digitalRead(2) == 0 && estadomiz3 == false) {
    msgmiz3.add(65);
    msgmiz3.add(127);
    Udp.beginPacket(outIp, outPort);
    msgmiz3.send(Udp);
    Udp.endPacket();
    msgmiz3.empty();
    estadomiz3 = true;
}
```

```
if (digitalRead(2) == 1 && estadomiz3 == true) {
    msgmiz3.add(65);
    msgmiz3.add(0);
    Udp.beginPacket(outIp, outPort);
    msgmiz3.send(Udp);
    Udp.endPacket();
    estadomiz3 = false;
}
```

```
//Minino Z4
```

```
if (digitalRead(12) == 0 && estadomiz4 == false) {
    msgmiz4.add(66);
    msgmiz4.add(127);
    Udp.beginPacket(outIp, outPort);
    msgmiz4.send(Udp);
    Udp.endPacket();
    msgmiz4.empty();
    estadomiz4 = true;
}
```

```
if (digitalRead(12) == 1 && estadomiz4 == true) {
    msgmiz4.add(66);
    msgmiz4.add(0);
}
```

```
Udp.beginPacket(outIp, outPort);  
msgmiz4.send(Udp);  
Udp.endPacket();  
estadomiz4 = false;  
}
```

APÊNDICE B – CÓDIGO FONTE EM PROCESSING

```
import oscP5.*;
import netP5.*;
import processing.video.*;

Movie video0;
Movie video1;
Movie video2;
Movie video3;
Movie video4;

String msgOSC = "init";

OscP5 oscP5;
NetAddress myRemoteLocation;

void setup() {
  fullScreen();
  frameRate(30);

  oscP5 = new OscP5(this,12000);

  myRemoteLocation = new NetAddress("127.0.0.1",12000);

  video0 = new Movie(this, "1080 (4).mov");
  video0.loop();
  video1 = new Movie(this, "1080 (1).mov");
  video1.loop();
  video2 = new Movie(this, "1080 (2).mov");
  video2.loop();
  video3 = new Movie(this, "1080 (3).mov");
  video3.loop();

  background(0);

}
```

```

void draw() {

    if(msgOSC.equals("/Direita/Anelar/z1/")){
        video0.loop();
        video1.stop();
        video2.stop();
        video3.stop();
        imageMode(CENTER);
        image(video0, 683, 384);
    }

    if(msgOSC.equals("/Direita/Anelar/z2/")){
        video0.stop();
        video1.loop();
        video2.stop();
        video3.stop();
        imageMode(CENTER);
        image(video1, 683, 384);
    }

    if(msgOSC.equals("/Direita/Anelar/z3/")){
        video0.stop();
        video1.stop();
        video2.loop();
        video3.stop();
        imageMode(CENTER);
        image(video2, 683, 384);
    }

    if(msgOSC.equals("/Direita/Anelar/z4/")){
        video0.stop();
        video1.stop();
        video2.stop();
        video3.loop();
        imageMode(CENTER);
        image(video3, 683, 384);
    }

}

```

```
/* incoming osc message are forwarded to the oscEvent method. */  
void oscEvent(OscMessage theOscMessage) {  
    /* print the address pattern and the typetag of the received OscMessage */  
    print("### received an osc message.");  
    print(" addrpattern: "+theOscMessage.addrPattern());  
    println(" typetag: "+theOscMessage.typetag());  
    msgOSC = theOscMessage.addrPattern();  
    println(msgOSC);  
}
```

```
void movieEvent(Movie m) {  
    m.read();  
}
```

MANUAL DE CONFIGURAÇÃO E UTILIZAÇÃO DO LUMU

1. Baixar e instalar a suíte PurrData disponível no endereço: <https://github.com/agraef/purr-data/releases>
2. Baixar e instalar o Ableton Live 11 Trial, disponível no endereço: <https://www.ableton.com/en/trial>
3. Executar o arquivo "Receiver.pd" disponível na pasta "LuMU – Experimento"
4. Executar o arquivo "LuMU.als" disponível na pasta "LuMU – Experimento"
5. Alterar entre canal 1 (para sons percussivos) e canal 2 (para sons melódicos).

APÊNDICE D – QUESTIONÁRIO AVALIATIVO

Eu acredito que o dispositivo é de fácil configuração. *

1 2 3 4 5

Discordo totalmente Concordo totalmente

Eu acredito que o dispositivo é de fácil utilização. *

1 2 3 4 5

Discordo totalmente Concordo totalmente

Na sua opinião o aprendizado para a utilização do dispositivo foi intuitivo? *

1 2 3 4 5

Discordo totalmente Concordo totalmente

Eu acredito que a utilização de recursos auxiliares na sala de aula pode desempenhar um fator de aumento do interesse e participação dos alunos? *

1 2 3 4 5

Discordo totalmente Concordo totalmente

Você concorda com a seguinte afirmação? "A utilização do dispositivo em sala de aula pode ter um impacto positivo no processo de aprendizagem dos alunos?" *

1 2 3 4 5
Discordo totalmente Concordo totalmente

A possibilidade de execução de sons percussivos e melódicos em um mesmo dispositivo é um fator positivo para o aprendizado musical? *

1 2 3 4 5
Discordo totalmente Concordo totalmente

Levar o dispositivo (LuMU + Computador + Caixa de som) para sala de aula é algo impeditivo para a utilização do dispositivo? *

1 2 3 4 5
Discordo totalmente Concordo Totalmente

Segundo sua opinião o LuMU pode ser útil em: *

- Aulas de musicalização infantil
- Aulas de musicalização de jovens e adultos
- Aulas de regência
- Aulas de contraponto e/ou harmonia
- Aulas de instrumentação-específicas (teclas, sopros...)
- Aulas de percussão
- Outro: _____

Você tem alguma sugestão para o desenvolvimento futuro do dispositivo?

Sua resposta

Questionário de validação do projeto LuMU

***** LEIA ATENTAMENTE *****

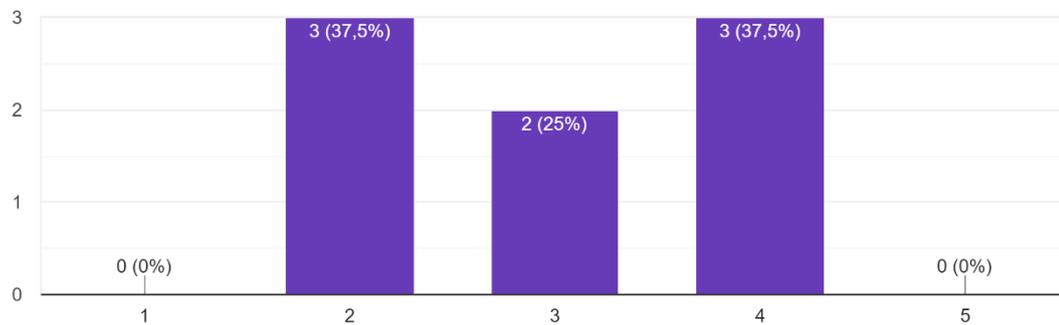
Declaro que ao prosseguir no preenchimento deste formulário, concordei em participar na pesquisa referente ao projeto/pesquisa intitulado(a) "LUMU: DESENVOLVIMENTO DE UM INSTRUMENTO MUSICAL CONTROLADOR SEM FIOS COMO RECURSO AUXILIAR EM SALA DE AULA PARA PROFESSORES DE MÚSICA" desenvolvida(o) por Francisco Breno Moura Alves. Fui informado(a), ainda, de que a pesquisa é orientada pelo professor Dr. CARLOS EDUARDO COELHO FREIRE BATISTA. Afirmando que aceitei participar por minha própria vontade, sem receber qualquer incentivo financeiro ou ter qualquer ônus e com a finalidade exclusiva de colaborar para o sucesso da pesquisa. Fui informado(a) dos objetivos estritamente acadêmicos do estudo. Minha colaboração se fará de forma anônima, por meio do preenchimento deste formulário.

APÊNDICE F – RESULTADOS DO QUESTIONÁRIO

F-1

Eu acredito que o dispositivo é de fácil configuração.

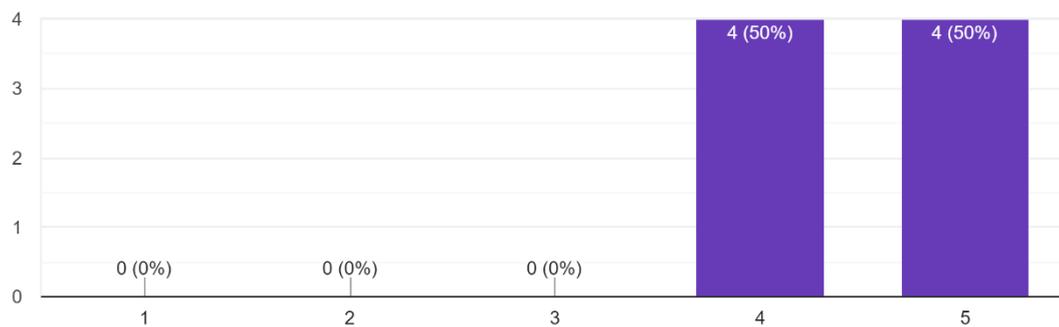
8 respostas



F-2

Eu acredito que o dispositivo é de fácil utilização.

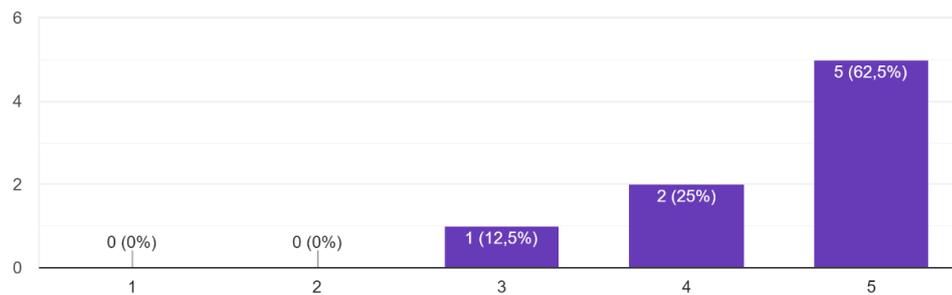
8 respostas



F-3

Na sua opinião o aprendizado para a utilização do dispositivo foi intuitivo?

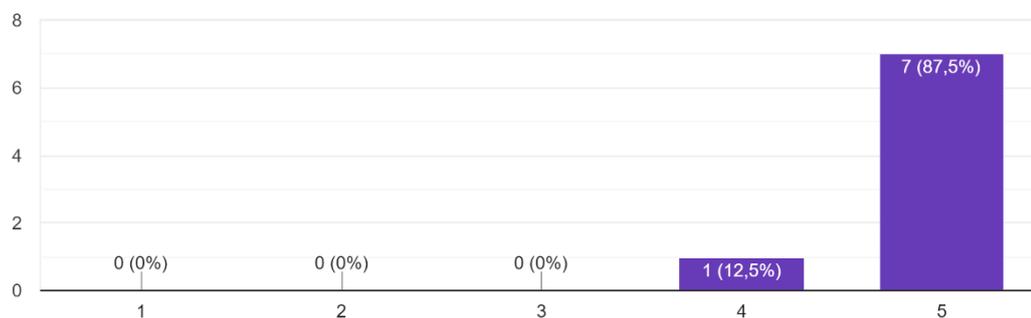
8 respostas



F-4

Eu acredito que a utilização de recursos auxiliares na sala de aula pode desempenhar um fator de aumento do interesse e participação dos alunos?

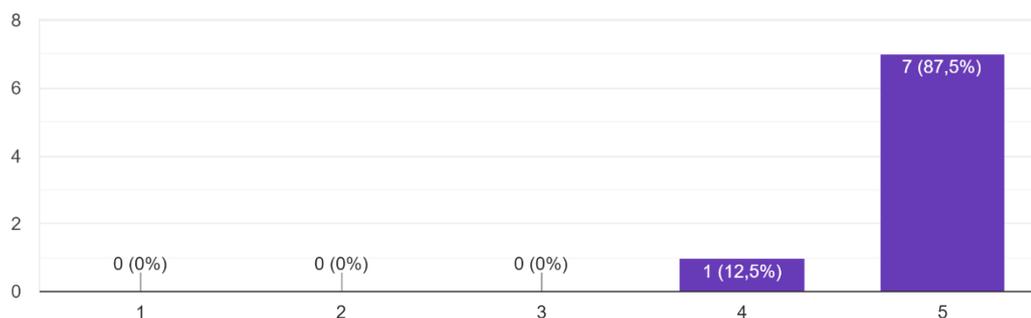
8 respostas



F-5

Você concorda com a seguinte afirmação? "A utilização do dispositivo em sala de aula pode ter um impacto positivo no processo de aprendizagem dos alunos?"

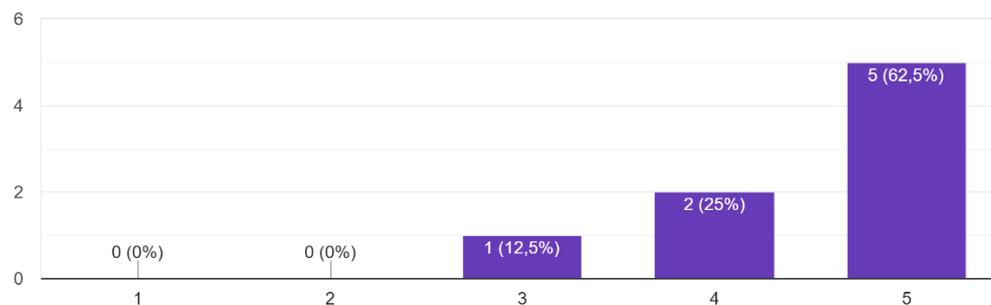
8 respostas



F-6

A possibilidade de execução de sons percussivos e melódicos em um mesmo dispositivo é um fator positivo para o aprendizado musical?

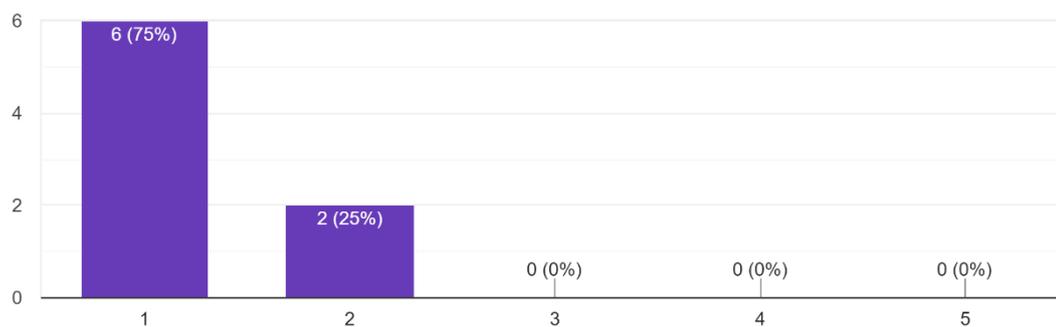
8 respostas



F-7

Levar o dispositivo (LuMU + Computador + Caixa de som) para sala de aula é algo impeditivo para a utilização do dispositivo?

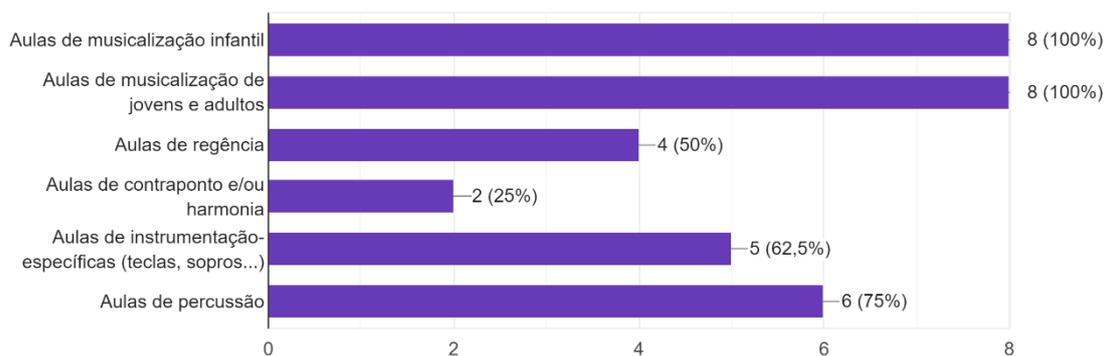
8 respostas



F-8

Segundo sua opinião o LuMU pode ser útil em:

8 respostas



F-9

Você tem alguma sugestão para o desenvolvimento futuro do dispositivo?

2 respostas

Criar um aplicativo para celular ao invés do uso com o computador

Que a luva possa ser vestida e retirada mais facilmente