



UNIVERSIDADE ESTADUAL PAULISTA
Instituto de Ciência e Tecnologia de Sorocaba
Campus São João da Boa Vista

VINICIUS FERNANDO DE SOUZA

DESENVOLVIMENTO DE UM BISTURI CIRÚRGICO HÁPTICO
INTEGRADO COM REALIDADE VIRTUAL IMERSIVA UTILIZANDO
QUEST 2

Sorocaba

2025

VINICIUS FERNANDO DE SOUZA

**DESENVOLVIMENTO DE UM BISTURI CIRÚRGICO HÁPTICO
INTEGRADO COM REALIDADE VIRTUAL IMERSIVA UTILIZANDO
QUEST 2**

Dissertação apresentada como parte dos requisitos para obtenção de título de Mestre em Engenharia Elétrica, junto ao Programa de Pós-Graduação em Engenharia Elétrica, da Universidade Estadual Paulista “Júlio de Mesquita Filho”.

Área de concentração: Sistemas Eletrônicos.

Orientador: Prof. Dr. Ivando Severino Diniz

Sorocaba

2025

S729d	Souza, Vinicius Fernando de Desenvolvimento de um bisturi cirúrgico háptico integrado com realidade virtual imersiva utilizando Quest 2 / Vinicius Fernando de Souza. -- Sorocaba, 2025 129 p. : tabs., fotos Dissertação (mestrado) - Universidade Estadual Paulista (UNESP), Instituto de Ciência e Tecnologia, Sorocaba Orientador: Ivando Severino Diniz 1. Realidade virtual. 2. Inovações médicas. 3. Ambientes virtuais compartilhados. 4. Internet das coisas. I. Título.
-------	---

CERTIFICADO DE APROVAÇÃO

TÍTULO DA DISSERTAÇÃO: DESENVOLVIMENTO DE UM BISTURI CIRÚRGICO HÁPTICO INTEGRADO COM REALIDADE VIRTUAL IMERSIVA UTILIZANDO QUEST 2

AUTOR: VINICIUS FERNANDO DE SOUZA

ORIENTADOR: IVANDO SEVERINO DINIZ

Aprovado como parte das exigências para obtenção do Título de Mestre em Engenharia Elétrica, área: Sistemas Eletrônicos pela Comissão Examinadora:

Documento assinado digitalmente
 **IVANDO SEVERINO DINIZ**
Data: 29/05/2025 12:30:03-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. IVANDO SEVERINO DINIZ (Participação Presencial)

Departamento de Engenharia de Controle e Automação / Instituto de Ciência e Tecnologia / UNESP / Sorocaba

Documento assinado digitalmente
 **SERGIO SHIMURA**
Data: 29/05/2025 13:01:52-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. SERGIO SHIMURA (Participação Presencial)

Departamento de Indústria / Instituto Federal de Educação, Ciência e Tecnologia de São Paulo - Campus Sorocaba

Documento assinado digitalmente
 **MARIA GLORIA CANO DE ANDRADE**
Data: 02/06/2025 11:06:58-0300
Verifique em <https://validar.iti.gov.br>

Prof^a. Dr^a. MARIA GLÓRIA CAÑO DE ANDRADE (Participação Presencial)

Departamento de Engenharia de Controle e Automação / Instituto de Ciência e Tecnologia UNESP Campus de Sorocaba

Sorocaba, 29 de maio de 2025

AGRADECIMENTOS

Aos meus pais, avó, esposa, sogro e sogra, que apesar de todas as dificuldades enfrentadas, sempre acreditaram e me apoiaram em todas as minhas escolhas da vida.

Ao meu orientador, Ivando Severino Diniz, pelo valioso apoio e orientação, sempre disponível para transmitir seu conhecimento e incitar reflexões indispensáveis para a realização desse trabalho com êxito.

A professora Marilza, por sempre apoiar e fomentar a pesquisa e desenvolvimento de novos projetos. Obrigado pela confiança e pelo grande trabalho realizado com todos os alunos.

“A tecnologia é apenas uma ferramenta. Em termos de conseguir as crianças a trabalhar juntas e motivá-las, o professor é o mais importante.” Bill Gates.

RESUMO

Este trabalho apresenta o desenvolvimento de um bisturi para ambientes de realidade virtual, com o objetivo de melhorar a qualidade do treinamento médico e a aprendizagem prática. O projeto utiliza o headset Meta Quest 2 e o módulo Seeed XIAO nRF52840 Sense, integrando sensores como acelerômetro, giroscópio e mini motores de vibração para proporcionar feedback háptico aos usuários. Os componentes principais incluem um acelerômetro analógico ADXL335, trimpot, botões, resistores, transistores, diodos e uma bateria recarregável. A montagem do circuito foi realizada em uma protoboard de 400 pinos, com o firmware desenvolvido para coletar dados dos sensores, transmitir os dados utilizando *Bluetooth Low Energy* (BLE) e controlar os atuadores. A comunicação entre o bisturi e o Meta Quest 2 é feita via BLE, garantindo a sincronização de dados em tempo real. A sala de cirurgia virtual foi desenvolvida na plataforma Unity, permitindo uma interação imersiva e intuitiva com o ambiente e os objetos virtuais. Testes experimentais mostraram que o dispositivo apresenta precisão e sensibilidade adequadas dos sensores, além de um tempo de resposta eficiente na comunicação BLE. O desenvolvimento deste bisturi representa um avanço significativo na integração de tecnologias para realidade virtual, com potencial para contribuir para o avanço da educação e treinamento médicos. A base sólida estabelecida por este projeto abre portas para futuras pesquisas e aprimoramentos no campo da realidade virtual.

Palavras-chave: Realidade virtual; Interface de usuário 3D; dispositivos hápticos; realidade aumentada; simulações médicas; Internet das Coisas; BLE; Simulações Médicas

ABSTRACT

This work presents the development of a surgical instrument for virtual reality environments, aiming to improve the quality of medical training and practical learning. The project uses the Meta Quest 2 headset and the Seeed XIAO nRF52840 Sense module, integrating sensors such as an accelerometer, gyroscope and mini vibration motors to provide haptic feedback to users. The main components include an ADXL335 analog accelerometer, trimpot, buttons, resistors, transistors, diodes and a rechargeable battery. The circuit was assembled on a 400-pin breadboard, with the firmware developed to collect data from the sensors, transmit the data using Bluetooth Low Energy (BLE) and control the actuators. Communication between the surgical instrument and the Meta Quest 2 is done via BLE, ensuring real-time data synchronization. The virtual operating room was developed on the Unity platform, allowing immersive and intuitive interaction with the environment and virtual objects. Experimental tests have shown that the device presents adequate precision and sensitivity of the sensors, in addition to an efficient response time in BLE communication. The development of this scalpel represents a significant advance in the integration of virtual reality technologies, with the potential to contribute to the advancement of medical education and training. The solid foundation established by this project opens doors for future research and improvements in the field of virtual reality.

Keywords: Virtual Reality; 3DUI; haptic devices; Augmented Reality; medical simulations; Internet of Things; BLE; Medical Simulatio

LISTA DE ILUSTRAÇÕES

Figura 1 — Trabalhos similares.....	23
Figura 2 —Trabalhos utilizados.....	24
Figura 3 — Código em Python	25
Figura 4 — Contagem de palavras de interesse	25
Figura 5 — Código em Python	27
Figura 6 — Palavras mais frequentes	28
Figura 7 — Comparativo XR: VR, AR e MR.....	33
Figura 8 — Realidade aumentada.....	34
Figura 9 — Ambiente de realidade virtual	35
Figura 10 — Ambiente de realidade virtual	37
Figura 11 — Cirurgião com Microsoft HoloLens.....	38
Figura 12 — Meta Quest 2	45
Figura 13 — Recursos suportados.....	46
Figura 14 — Pinagem XIAO Seeed nRF52840 Sense.....	46
Figura 15 — Módulo XIAO Seeed nRF52840 Sense	47
Figura 16 — Módulo ADXL335 (EVAL-ADXL335).....	48
Figura 17 — Chip LSM6DS3TR-C	50
Figura 18 — Captura de tela do ambiente de desenvolvimento Unity.....	52
Figura 19 — Diagrama de blocos.....	58
Figura 20 — Montagem circuito (Fritzing)	61
Figura 21 — Montagem física do circuito	62
Figura 22 — Esquemático circuito.....	63
Figura 23 — Placa de circuito impresso.	66
Figura 24 — Modelo 3D Placa de circuito impresso.....	67
Figura 25 — Modelo 3D Placa de circuito impresso.....	67
Figura 26 — Estrutura main.h	69
Figura 27 — Estrutura principal do código	70
Figura 28 — Perspectiva frontal bisturi	73
Figura 29 — Perspectiva traseira bisturi	73
Figura 30 — Vista explodida bisturi e componentes	75
Figura 31 — Gerenciador de pacotes	77
Figura 32 — Assets instalados.....	78
Figura 33 — Sala de cirurgia com modelo humano virtual anatômico	79
Figura 34 — Sala de cirurgia com modelo humano virtual anatômico	80
Figura 35 — Script ScannerBLE	81
Figura 36 — Script BLEManager.....	82
Figura 37 — Script SendBLECommand.....	82
Figura 38 — Script SurgicalCutter.....	83
Figura 39 — GameObject e componentes	84
Figura 40 — SurgicalCutter atribuído como componente ao objeto Scalpel1_N10...84	
Figura 41— Script SurgicalCutterMesh	86
Figura 42 — Modelos humano virtuais com suas malhas visíveis	87
Figura 43 — Script SurgicalCutterMesh adicionado ao modelo humano virtual.....	87
Figura 44 — Placa produzida.....	89

Figura 45 — Primeira placa parcialmente montada em funcionamento	90
Figura 46 — Bisturi montado em funcionamento	91
Figura 47 — Bisturi completo montado	92
Figura 48 — Adicionar um novo dispositivo Bluetooth	94
Figura 49 — Dispositivos adicionados	95
Figura 50 — Dispositivos adicionados	95
Figura 51 — Testes para calibração do controle	96
Figura 52 — Modelo de mensagem GATT 16 Bytes	97
Figura 53 — Modelo de mensagem GATT 3 Bytes	97
Figura 54 — Função de inscrição em característica do BLE	98
Figura 55 — Função de escrita em característica do BLE	99
Figura 56 — Dados recebidos do bisturi	104
Figura 57 — Dados recebidos do bisturi	105

LISTA DE TABELAS

Tabela 1 — Leitura dos sensores.....	93
Tabela 2 — Consumo de corrente dos componentes (Caso 1)	106
Tabela 3 — Consumo de corrente dos componentes (Caso 2)	107
Tabela 4 — Consumo de corrente dos componentes (Caso 3)	107
Tabela 5 — Erros médios absolutos de cada eixo	108
Tabela 6 — Comparação entre o sistema de vibração	109
Tabela 7 — Comparação entre sistemas desenvolvidos	110

LISTA DE ABREVIATURAS E SIGLAS

3DUI	Interface de usuário 3D
AR	<i>Augmented Reality</i> (Realidade Aumentada)
ADC	<i>Analog-to-Digital Converter</i> (Conversor Analógico-Digital)
BLE	<i>Bluetooth Low Energy</i> (Bluetooth de Baixa Energia)
CI	Circuito Integrado
CPU	<i>Central Processing Unit</i> (Unidade Central de Processamento)
dBm	Decibels milli-watts (nível de potência expresso em decibéis com referência a 1mW)
DC	<i>Direct Current</i> (Corrente Contínua)
DOF	<i>Degrees of Freedom</i> (Graus de liberdade)
GND	<i>Ground</i> (Terra)
GPIO	<i>General-Purpose Input/Output</i> (Entrada/Saída de Propósito Geral)
I2C	<i>Inter-Integrated Circuit</i> (Circuito Inter-Integrado)
IDE Integrado)	<i>Integrated Development Environment</i> (Ambiente de Desenvolvimento Integrado)
IoT	<i>Internet of Things</i> (Internet das Coisas)
LED	<i>Light Emitting Diode</i> (Diodo Emissor de Luz)
Li-Po	<i>Lithium Polymer</i> (Polímero de Lítio)
LSM6DS3	Giroscópio e Acelerômetro de 6 eixos
mA/mAh	Milésimo de Amper / Milésimo de Amper hora
ms	Milissegundo
GATT	<i>Generic ATtribute Profile</i> (Perfil de Atributo Genérico)
GHz	Gigahertz
HID	<i>Human Interface Device</i> (Dispositivo de interface Humana)
MR	<i>Mixed Reality</i> (Realidade Mista)
nRF52840	Microcontrolador Nordic com Bluetooth 5.0
PCB	<i>Printed Circuit Board</i> (Placa de Circuito Impresso)
PCI	Placa de Circuito Impresso
PWM	<i>Pulse Width Modulation</i> (Modulação por Largura de Pulso)

RAM	<i>Random Access Memory</i> (Memória de Acesso Aleatório)
ROM	<i>Read-Only Memory</i> (Memória Somente de Leitura)
SPI	<i>Serial Peripheral Interface</i> (Interface Periférica Serial)
UART	<i>Universal Asynchronous Receiver/Transmitter</i> (Receptor/Transmissor Universal Assíncrono)
USB	<i>Universal Serial Bus</i> (Barramento Serial Universal)
UUID	<i>Universally Unique Identifier</i> (Identificador Único Universal)
V	Volts (Tensão)
VCC	<i>Voltage on the Collector Circuit</i> (Tensão no circuito coletor)
VR	<i>Virtual Reality</i> (Realidade Virtual)
XR	<i>Extended Reality</i> (Realidade Estendida)

LISTA DE SIMBOLOS

Ω	Ohm (Resistência elétrica)
$^{\circ}\text{C}$	Graus Celsius (Temperatura)
%	Porcentagem
$^{\circ}$	Grau (Orientações angulares)
$\pm$	Mais ou menos (faixa de variação)

SUMÁRIO

1	INTRODUÇÃO	18
1.1.	OBJETIVOS.....	19
1.1.1.	Objetivo Geral.....	19
1.1.2.	Objetivos Específicos	19
1.2.	JUSTIFICATIVA.....	20
1.3.	ORGANIZAÇÃO / ESTRUTURA DA DISSERTAÇÃO	20
2	REVISÃO BIBLIOGRÁFICA	22
2.1.	METODOLOGIA DA REVISÃO	22
2.2.	FUNDAMENTAÇÃO TEÓRICA	29
2.2.1.	Interface de Usuário 3D (3DUI)	30
2.2.1.1.	<i>Realidade Estendida (XR)</i>	31
2.2.1.2.	<i>Realidade Aumentada (AR)</i>.....	33
2.2.1.3.	<i>Realidade Virtual (VR)</i>	34
2.2.1.4.	<i>Realidade Mista (MR)</i>.....	36
2.2.2.	Cirurgias em Realidade Virtual	37
2.2.2.1.	<i>Resposta Tátil (Feedback Haptico)</i>	38
2.2.3.	Internet Das Coisas (IoT).....	39
2.2.3.1.	<i>Bluetooth Low Energy</i>.....	40
2.2.4.	Acelerômetro	41
2.2.5.	Giroscopio	41
3	MATERIAIS E MÉTODOS.....	43
3.1.	DESCRIÇÃO DOS COMPONENTES UTILIZADOS.....	43
3.1.1.	Meta Quest 2	44
3.1.2.	Seeed Xiao Ble Sense Nrf52840	45
3.1.3.	ADXL335	47
3.1.4.	LSM6DS3TR-C.....	49
3.2.	FUNCIONALIDADE DOS COMPONENTES	50
3.3.	AMBIENTE UNITY.....	51
3.4.	PROTOCOLO DE COMUNICAÇÃO BLE.....	53
3.4.1.	Configuração e Integração com Meta Quest 2	54
4	DESENVOLVIMENTO.....	57
4.1.	DIAGRAMA DO CIRCUITO.....	58

4.1.1.	Circuito eletrônico do Bisturi	60
4.2.	PROJETO E IMPLEMENTAÇÃO DO BISTURI	61
4.2.1.	Montagem do circuito	61
4.2.2.	Diagrama esquemático do circuito.....	63
4.2.3.	Design do Hardware	65
4.3.	FIRMWARE E TESTES.....	68
4.4.	Criação do design mecânico	72
4.5.	AMBIENTE VIRTUAL E INTEGRAÇÃO COM META QUEST 2.....	76
4.5.1.	Design da Interface de Usuário	78
4.6.	DESENVOLVIMENTO SALA DE REALIDADE VIRTUAL EM UNITY ...	79
4.6.1.	Scripts Bluetooth Low Energy.....	80
4.6.2.	Simulação de corte	85
5	PROTÓTIPO EXPERIMENTAL.....	88
5.1.	IMPLEMENTAÇÃO DO BISTURI	88
5.2.	SISTEMA DE PROCESSAMENTO DE TESTES	92
5.3.	SISTEMA DE PROCESSAMENTO IMPLEMENTADO.....	96
5.4.	CONSIDERAÇÕES PARCIAIS.....	100
6	RESULTADOS EXPERIMENTAIS.....	102
6.1.	TEMPO DE RESPOSTA DA COMUNICAÇÃO BLE.....	102
6.2.	TEMPO DE BATERIA.....	106
6.3.	PRECISÃO E SENSIBILIDADE DOS SENSORES	107
6.4.	RESPOSTA TÁTIL E SIMULAÇÃO DE CORTES VIRTUAIS.....	109
6.5.	COMPARAÇÃO COM O ESTADO DA ARTE.....	109
6.6.	CONSIDERAÇÕES PARCIAIS.....	111
7	CONCLUSÕES FINAIS.....	113
8	SUGESTÕES DE TRABALHOS FUTUROS.....	114
9	PUBLICAÇÕES REALIZADAS.....	115
	REFERÊNCIAS.....	116
	APENDICE A — FIRMWARE	124

1 INTRODUÇÃO

Nesse capítulo será apresentada a motivação do desenvolvimento do presente trabalho, abordando os principais objetivos e sua organização.

A realidade virtual (VR) detém o poder de proporcionar aos usuários experiências imersivas além dos limites do mundo real. Para desbloquear plenamente o potencial da tecnologia VR, são necessárias mudanças substanciais na abordagem dos designers ao criar ambientes virtuais (Mostafa et al., 2014). O interesse crescente na utilização de tecnologias imersivas, impulsionado pelos avanços em hardware e software de alta qualidade, tem promovido a evolução deste campo (Fonnet & Prié, 2021).

Essas tecnologias emergentes de realidade-virtualidade têm aplicações em diferentes campos da vida, ou seja, saúde, engenharia e design, robótica, automotivo, industrial, manufatura e design, marketing e publicidade, construção, educação e treinamento agrícola, operação militar, entretenimento e turismo, etc (Mirza et al. 2022).

A VR, juntamente com a realidade aumentada (AR) e a realidade mista (MR), oferece uma nova maneira de apoiar a educação médica, ajudando estudantes a aprender anatomia humana e técnicas cirúrgicas (Xu et al., 2023). Tradicionalmente, o treinamento cirúrgico é realizado sob a supervisão de um cirurgião experiente, sendo caro, demorado e de eficácia variável. A utilização de simuladores de VR é uma opção complementar ao treinamento padrão (Chaabra et al., 2023).

Os avanços recentes na tecnologia de VR também têm possibilitado a criação, aplicação, avaliação e entrega de aplicações interativas a um custo menor (González et al., 2020). A imersão é uma característica definidora da VR, permitindo que o usuário seja transportado para um mundo totalmente virtual, distanciando-se da realidade física, principalmente através da visão e audição (Xie et al., 2021).

Em sistemas de treinamento de habilidades médicas baseados em VR/AR, a interação háptica pode melhorar drasticamente a qualidade de aprendizagem dos alunos. A inserção de agulhas, por exemplo, não é apenas uma habilidade básica para os cirurgiões, mas também está altamente relacionada aos resultados do tratamento. O feedback tátil da inserção da agulha ajuda o cirurgião a perceber as propriedades dos tecidos do corpo.

A percepção háptica de outras ferramentas, como o bisturi, é semelhante à da agulha, permitindo interações com objetos virtuais no ambiente virtual através de dispositivos hápticos que produzem estímulos multimodais, incluindo forças, vibração e estímulos térmicos. Para manter alta consistência entre as sensações reais e virtuais, o feedback tátil fornecido deve ter baixa taxa de erro e latência computacional (Li et al., 2022).

Este projeto consiste em um trabalho experimental com materiais, aparelhos e medidas. Desenvolveu-se a metodologia para aquisição da aceleração de movimento e velocidade angular do equipamento sob estudo. Será estudada a placa XIAO SEED nRF52840 SENSE, que utiliza o controlador nRF52840 da Nordic, de modo a estabelecer uma sistemática de programação e de implementação de uma comunicação via Bluetooth que possa se conectar ao óculos meta quest 2, transmitindo os dados tratados do acelerômetro e giroscópio ao óculos, o dispositivo terá 4 botões digitais e um trimpot.

1.1. OBJETIVOS

1.1.1. Objetivo Geral

Desenvolver um bisturi cirúrgico com feedback aplicado ao ambiente de realidade virtual.

1.1.2. Objetivos Específicos

Os objetivos específicos do trabalho são:

- Desenvolver um sistema embarcado que adquise os dados de um acelerômetro analógico e um giroscópio digital, comandos através dos botões e acione os motores de vibração, processe esses valores e transmita via Bluetooth Low Energy.
- Desenvolver um ambiente de realidade virtual em Unity, que receba os dados via Bluetooth Low Energy, processe e execute os movimentos.
- Embarcar o ambiente de realidade virtual (RV) desenvolvido em Unity nos óculos Meta Quest 2.

1.2. JUSTIFICATIVA

A crescente adoção de tecnologias de realidade virtual (VR) no treinamento médico tem impulsionado o desenvolvimento de ambientes imersivos para simulação de procedimentos cirúrgicos. No entanto, ao realizar uma análise aprofundada da literatura técnica e científica, observou-se que nenhum dos trabalhos revisados abordou especificamente o desenvolvimento eletrônico de um bisturi físico com integração háptica e comunicação sem fio voltado à interação direta com ambientes de realidade virtual.

A maioria das abordagens existentes se concentra na simulação gráfica ou no uso de controladores genéricos, sem explorar o potencial de dispositivos customizados que repliquem a ergonomia e a resposta tátil de um bisturi real. Essa ausência revela uma lacuna na literatura atual, sobretudo no que diz respeito à utilização de sensores inerciais e motores de vibração para simular a resistência dos tecidos durante cortes virtuais, além da comunicação em tempo real com plataformas como o Unity por meio do *protocolo Bluetooth Low Energy* (BLE).

Diante disso, justifica-se o desenvolvimento de uma solução integrada, de alta imersão, que permita a criação de instrumentos hápticos especializados para VR médica, contribuindo não apenas para a evolução tecnológica da área, mas também para a formação prática de estudantes e profissionais da saúde em ambientes seguros e controlados.

1.3. ORGANIZAÇÃO / ESTRUTURA DA DISSERTAÇÃO

Este trabalho está organizado em sete capítulos, conforme descrito a seguir:

No capítulo 2 - Revisão Bibliográfica, é discutida a literatura existente relacionada ao tema do estudo, abordando os principais trabalhos e pesquisas que fundamentam o desenvolvimento do bisturi para ambientes de realidade virtual e a fundamentação teórica, são apresentados os conceitos e teorias essenciais para o projeto, incluindo Realidade Estendida (XR), Realidade Aumentada (AR), Realidade Virtual (VR), Realidade Mista (MR), Internet das Coisas (IoT) e comunicação *Bluetooth Low Energy* (BLE), além de uma análise detalhada dos sensores e atuadores utilizados.

O capítulo 3 - Materiais e Métodos detalha os componentes e materiais utilizados no desenvolvimento do bisturi, incluindo o Meta Quest 2, o controlador Seeed XIAO BLE Sense nRF52840, acelerômetros, giroscópios, botões e trimpot, além dos descritivos do bisturi, o ambiente Unity e o protocolo de comunicação BLE.

No capítulo 4 - Desenvolvimento, é descrito o projeto e a implementação do bisturi, abrangendo o design do hardware, firmware e testes. Também são abordadas a criação da sala de realidade virtual, o design da interface de usuário, a implementação e testes em Unity, e a integração dos componentes, incluindo a comunicação entre o bisturi e o Meta Quest 2 e a sincronização de dados em tempo real.

O Capítulo 5 - Protótipo Experimental apresenta os circuitos eletrônicos implementados, a adequação de sensores e o sistema de processamento implementado, além de considerações parciais sobre o protótipo.

Os Capítulos 6 e 7 apresentam, respectivamente, os resultados experimentais obtidos e as conclusões finais do trabalho.

Essa estrutura visa proporcionar uma visão clara e organizada do desenvolvimento do projeto, facilitando a compreensão das etapas e dos resultados alcançados.

2 REVISÃO BIBLIOGRÁFICA

Neste capítulo, abordaremos a revisão da literatura, que foi a base para a definição do tema e a base teórica para o desenvolvimento deste projeto.

O avanço da tecnologia tem impulsionado o desenvolvimento de dispositivos cirúrgicos inovadores, especialmente em ambientes de realidade virtual (VR) e realidade aumentada (AR). Essas tecnologias permitem a criação de simuladores realistas, fundamentais para o treinamento médico, reduzindo a necessidade de prática direta em pacientes reais e possibilitando a realização de procedimentos cirúrgicos complexos em um ambiente seguro e controlado. Entretanto, a integração eficaz de interfaces 3D e feedback háptico ainda representa um desafio significativo.

Este capítulo apresenta uma revisão da literatura sobre os principais conceitos, metodologias e tecnologias relacionadas ao desenvolvimento de interfaces 3D e dispositivos cirúrgicos com resposta tátil para ambientes de realidade virtual. A revisão narrativa junto a busca manual conduzida foi essencial para identificar lacunas na pesquisa existente e fundamentar o desenvolvimento do presente estudo.

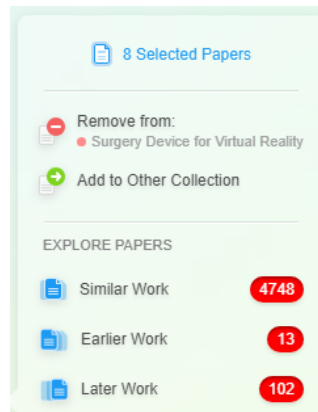
2.1. METODOLOGIA DA REVISÃO

Para o desenvolvimento da revisão bibliográfica, foi adotado o método de revisão narrativa. Uma pergunta clara e específica foi formulada para guiar todo o processo. A pergunta de pesquisa estabelecida foi: “Quais são as abordagens e tecnologias mais eficazes para desenvolver interfaces 3D e dispositivos cirúrgicos com feedback tátil em ambientes de realidade virtual?”

Com a pergunta definida, foram selecionados 8 artigos de referência recentes (publicados nos últimos 10 anos), relacionados a Realidade Virtual, Realidade Aumentada, Realidade Mista, Realidade Estendida, Internet das Coisas, dispositivos cirúrgicos/médicos para realidade virtual, resposta tátil, e háptico, nas seguintes bases de dados: IEEE Xplore, PubMed, Web of Science e Google Scholar.

Estes 8 artigos serviram como base para alimentar a plataforma Research Rabbit App, uma ferramenta de inteligência artificial que auxilia na busca de artigos científicos similares. A ferramenta encontrou 4748 trabalhos similares aos artigos de referência, considerando diversas bases de dados. A Figura 1 ilustra a quantidade de trabalhos similares identificados.

Figura 1 — Trabalhos similares



Fonte: Elaboração própria, 2024.

Dos 4748 trabalhos, foi aplicado um filtro temporal, considerando apenas os trabalhos dos últimos 10 anos, o número de artigos foi reduzido para 1799. Dos 1799 artigos, foram considerados os 300 mais relevantes sugeridos pela própria plataforma. Destes 300 artigos que foram selecionados, foi aplicado um critério considerando a presença de palavras-chave específicas no seu contexto, como: “Interface de usuário 3D (3DUI)”, “Resposta Tátil”, “Realidade Virtual”, “Realidade Aumentada”, “Realidade Mista”, “Realidade Estendida”, “Dispositivo Cirúrgico”, “Dispositivo Médico”, “Haptico”, “Internet das Coisas”.

Após aplicar os critérios acima e realizar uma análise detalhada dos trabalhos, a seleção final resultou em 53 artigos relevantes para a revisão. A Figura 2 apresenta uma ligação entre os trabalhos utilizados como base para esta dissertação.

Figura 3 — Código em Python

```

from bibtexparser.bparser import BibTexParser
from bibtexparser.customization import convert_to_unicode
from collections import Counter
import os

# Carregar o arquivo .bib
with open('/content/ResearchRabbit_Export_07_07_2024.bib', 'r',
          encoding='utf-8') as bibfile:
    parser = BibTexParser()
    parser.customization = convert_to_unicode
    bib_data = parser.parse_file(bibfile)

# Extrair abstracts
abstracts = []
for entry in bib_data.entries:
    if 'abstract' in entry:
        abstracts.append(entry['abstract'])

# Definir palavras-chave
keywords_of_interest = ["feedback", "tactil", "device", "virtual reality",
                        "augmented reality", "mixed reality",
                        "extended reality", "surgery device", "surgery",
                        "medical", "hapitc", "internet", "things", "3DUI"]

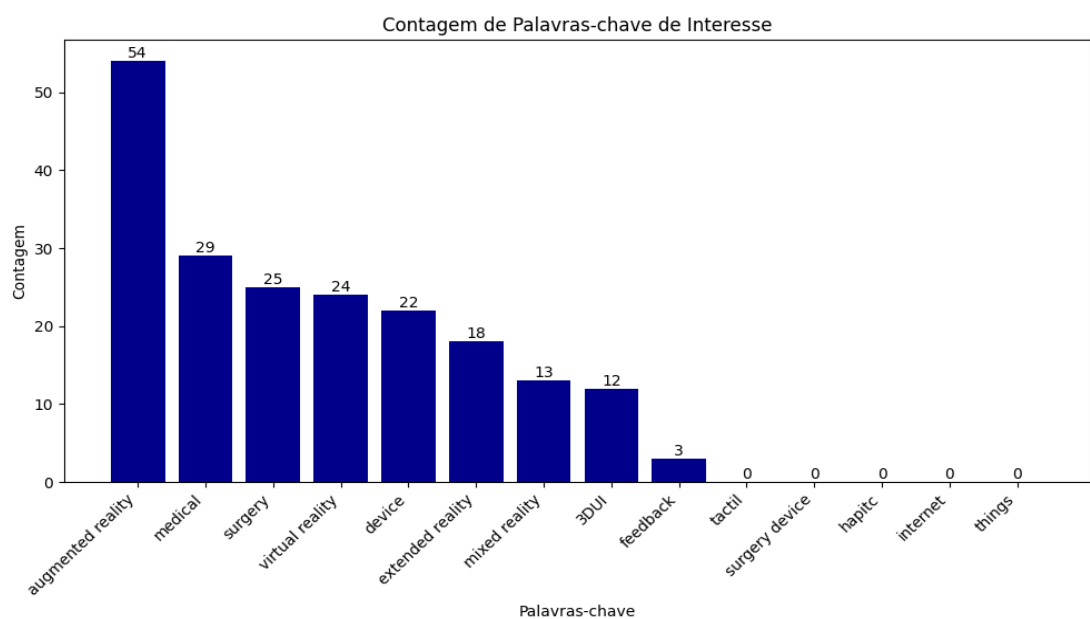
# Analisar abstracts
all_abstracts = ' '.join(abstracts)
keyword_counts = Counter({keyword: all_abstracts.count(keyword) for keyword in
                           keywords_of_interest})

```

Fonte: Elaboração própria, 2024.

O resultado da contagem de palavras podem ser visualizados na Figura 4.

Figura 4 — Contagem de palavras de interesse



Fonte: Elaboração própria, 2024.

Para validar se as palavras consultadas estão representando o conteúdo dos artigos, foi desenvolvido um segundo código em Python, afim de analisar a frequência com que as palavras aparecem nos artigos, sendo considerada apenas uma contagem por artigo e sendo excluídas da contagem as seguintes palavras: *"the", "a", "an", "as", "of", "with", "and", "in", "on", "for", "at", "to", "from", "by", "is", "are", "was", "were", "has", "have", "that", "this", "it", "be", "using", "we", "studies", "applications", "use", "review", "these", "can", "results", "research", "data", "their", "study", "which", "new", "our", "used", "however", "literature", "not", "or"*. Em processamento de linguagem natural, tais termos são denominados *'stop words'* (palavras de parada) em processamento de linguagem natural. Elas são extremamente comuns na linguagem e geralmente não contribuem significativamente para o significado do texto em análise. Essas palavras são frequentemente removidas durante a etapa de pré-processamento de textos para análise de dados textuais, como em análises de frequência de palavras. A Figura 5 apresenta o código desenvolvido para consulta das palavras.

Figura 5 — Código em Python

```

def preprocess_text(text):
    # Remover caracteres especiais e números, manter apenas palavras
    text = re.sub(r'^\w\s', '', text)
    # Converter para minúsculas
    text = text.lower()
    return text

# Carregar o arquivo .bib
with open('/content/ResearchRabbit_Export_07_07_2024.bib', 'r',
          encoding='utf-8') as bibfile:
    parser = BibTexParser()
    parser.customization = convert_to_unicode
    bib_data = parser.parse_file(bibfile)

# Dicionário para armazenar ocorrências únicas por artigo
unique_words_per_article = {}

# Extrair todos os abstracts e pré-processá-los
for entry in bib_data.entries:
    if 'abstract' in entry:
        abstract = entry['abstract']
        abstract = preprocess_text(abstract)

        # Encontrar todas as palavras únicas no abstract atual
        words = set(re.findall(r'\b\w+\b', abstract))

        # Adicionar palavras únicas ao dicionário por artigo
        unique_words_per_article[entry['ID']] = words

# Contar globalmente as ocorrências de palavras únicas
global_word_counts = Counter()
for words_set in unique_words_per_article.values():
    global_word_counts.update(words_set)

# Lista de palavras comuns que serão desconsideradas
common_words = ["the", "a", "an", "as", "of", "with", "and", "in", "on", "for",
                 "at", "to", "from", "by", "is", "are", "was", "were", "has",
                 "have", "that", "this", "it", "be", "using", "we", "studies",
                 "applications", "use", "review", "these", "can", "results",
                 "research", "data", "their", "study", "which", "new", "our",
                 "used", "however", "literature", "not", "or"]

# Filtrar as palavras comuns
filtered_word_counts = {word: count for word, count in
                        global_word_counts.items() if word not in common_words}

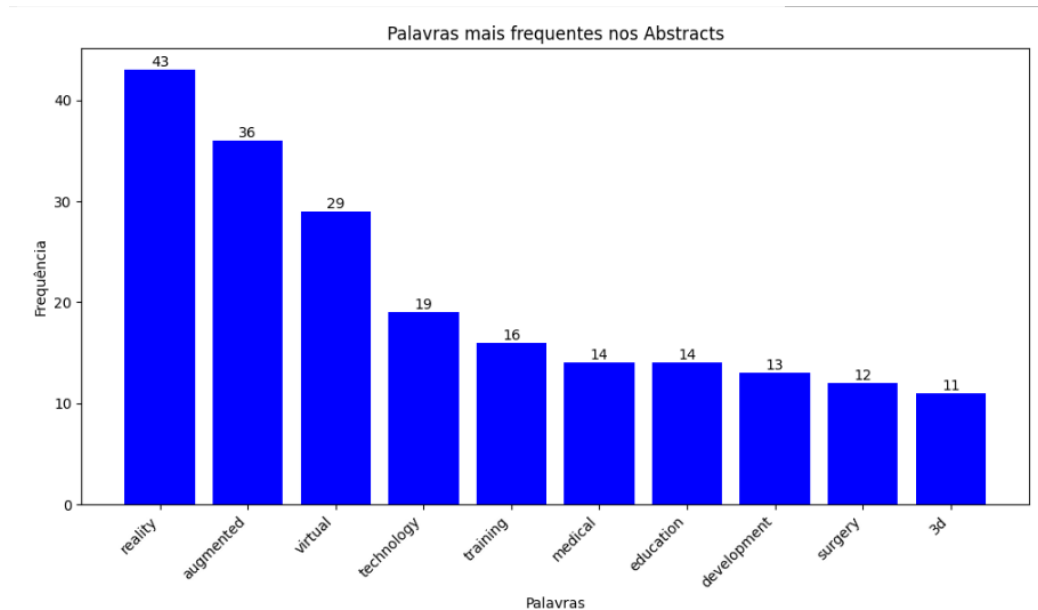
# Obter as 10 palavras mais frequentes (ajuste conforme necessário)
most_common_words = Counter(filtered_word_counts).most_common(12)

```

Fonte: Elaboração própria, 2024.

Foi solicitado as 12 palavras que mais se repetem, como resultado obtivemos: “reality: 43”, “augmented: 36”, “virtual: 29”, “technology: 19”, “training: 16”, “medical: 14”, “education: 14”, “development: 13”, “surgery: 12”, “3d: 11”. A Figura 6 apresenta o resultado em formato de gráfico.

Figura 6 — Palavras mais frequentes



Fonte: Elaboração própria, 2024.

Ao comparar os resultados da contagem de palavras e a frequência das palavras mais relevantes, podemos observar algumas conclusões interessantes:

"*Augmented reality*" é a palavra mais frequente tanto na contagem específica (54) quanto na análise geral (36). "*Virtual reality*" é outra palavra-chave com alta frequência em ambas as análises (24 na contagem específica e 29 na geral). Outras palavras-chave como "*medical*", "*surgery*" e "*device*" também mostram uma alta relevância em ambas as análises, indicando que são temas centrais nos resumos revisados.

Algumas palavras como "*extended reality*" e "*mixed reality*" têm contagens específicas mais baixas na primeira análise (18 e 13, respectivamente), comparadas com a análise geral onde "*reality*" e "*augmented*" têm frequências mais altas.

A repetição das palavras-chave como "technology", "training", "education" e "development" na análise geral indica uma ênfase em aspectos de tecnologia, treinamento e desenvolvimento em ambientes de realidade virtual e aumentada.

Além da revisão narrativa conduzida, foi realizada uma busca manual em diversas bases de dados científicas, incluindo *IEEE Xplore*, *PubMed*, *Web of Science* e *Google Scholar*, com os termos "*scalpel in virtual reality*" e "*physical scalpel for virtual reality environments*".

No entanto, não foram encontrados trabalhos acadêmicos que abordassem diretamente essa temática, reforçando a necessidade e a originalidade da pesquisa proposta neste estudo.

Após a análise aprofundada dos 53 artigos selecionados, observou-se uma ampla gama de estudos focados em dispositivos médicos, interface de usuário 3D, realidade virtual, aumentada e estendida, além de feedback em ambientes cirúrgicos virtuais e treinamento em ambientes de realidade virtual. No entanto, nenhum dos artigos revisados abordou especificamente o desenvolvimento de um bisturi para ambientes de realidade virtual, aumentada ou estendida. Este achado evidencia uma lacuna na literatura atual. Identificar essa ausência de desenvolvimento específico de um bisturi nesses ambientes tecnológicos destaca a relevância e a inovação do presente estudo. Este gap justifica a necessidade de aprofundar a pesquisa e o desenvolvimento de dispositivos cirúrgicos virtuais, particularmente de um bisturi adaptado para esses ambientes, visando aprimorar a precisão e a eficácia das cirurgias realizadas com auxílio de tecnologias imersivas.

2.2. FUNDAMENTAÇÃO TEÓRICA

Esta seção aborda os conceitos e teorias fundamentais para o desenvolvimento do bisturi destinado a ambientes de realidade virtual e sua integração com o Meta Quest 2. Primeiramente, serão discutidos os aspectos de Interface de Usuário 3D (3DUI) junto a Realidade Estendida (XR), incluindo Realidade Aumentada (AR), Realidade Mista (MR) e Realidade Virtual (VR), destacando suas características e aplicações. Em seguida, é explorada a Internet das Coisas (IoT) e sua importância para a comunicação entre dispositivos, com ênfase no uso do *Bluetooth Low Energy* (BLE) para a transmissão de dados.

Adicionalmente, será apresentada uma análise detalhada dos sensores e atuadores utilizados no projeto, tais como acelerômetros analógicos e digitais e giroscópios digitais, explicando suas funcionalidades e relevância para a precisão e eficácia do bisturi. A fundamentação teórica também incluirá o controlador XIAO SEEED nRF52840 SENSE, responsável pela integração e gerenciamento dos diversos componentes do sistema. Por fim, será descrita a implementação da sala de realidade virtual desenvolvida em Unity, destinada a simulações cirúrgicas, destacando as técnicas de modelagem, interatividade e controle empregadas.

Esta seção proporciona a base teórica necessária para compreender as escolhas tecnológicas e metodológicas adotadas no desenvolvimento do projeto, contextualizando os resultados e conclusões apresentados nas seções subsequentes.

2.2.1. Interface de Usuário 3D (3DUI)

As Interfaces de Usuário 3D (3DUI) representam uma das áreas mais dinâmicas e promissoras dentro do campo da interação humano-computador, especialmente no contexto da realidade virtual (VR). Diferente das interfaces bidimensionais tradicionais, as 3DUI permitem interações mais naturais e intuitivas, pois exploram a profundidade e a espacialidade, o que é crucial para aplicações que exigem um alto grau de imersão, como as simulações cirúrgicas (Mostafa et al., 2014; Takala et al., 2012).

O desenvolvimento de 3DUI tem se intensificado nos últimos anos, impulsionado pela evolução das tecnologias de VR e pela necessidade de criar ambientes virtuais que permitam aos usuários manipular objetos e realizar tarefas complexas de maneira eficiente. As interfaces 3D não apenas facilitam a navegação e a interação em espaços tridimensionais, mas também melhoram a precisão e a eficácia em atividades que demandam coordenação motora e percepção espacial (Fonnet et al., 2021).

Em aplicações cirúrgicas, as 3DUI têm se mostrado particularmente eficazes, permitindo que cirurgiões e estudantes de medicina pratiquem procedimentos complexos em ambientes simulados antes de realizá-los em situações reais. Essa capacidade de simulação não apenas aumenta a segurança dos pacientes, mas também melhora o treinamento e a preparação dos profissionais de saúde (Xu et al., 2023).

Além disso, as 3DUI possibilitam a criação de sistemas que combinam feedback tátil e visual, oferecendo uma experiência de usuário mais rica e próxima da realidade. A integração dessas interfaces com dispositivos de feedback háptico tem o potencial de revolucionar o treinamento médico, proporcionando uma imersão mais profunda e uma maior precisão nas simulações (Li et al., 2022).

Com o avanço contínuo da tecnologia, o campo das 3DUI devem continuar a evoluir, abrindo novas possibilidades para o desenvolvimento de aplicações em áreas como a educação, a medicina e o entretenimento. Estudos recentes apontam para o uso crescente dessas interfaces em simulações que exigem não apenas visualização tridimensional, mas também interações complexas que simulam atividades do mundo real (Bektic et al., 2021).

Abaixo está uma lista de recursos comuns de 3DUI que podem ser usados para caracterizar aproximadamente os aplicativos 3DUI, que compilaram um conjunto de requisitos que são necessários para um kit de ferramentas de pesquisa 3DUI:

- Estereografia 3D
- Renderização de visualização rastreada pela cabeça
- Interação de corpo inteiro
- Interação de duas mãos
- Interação de dedos
- Reconhecimento de gestos
- Técnicas de navegação
- Técnicas de manipulação de objetos
- Mecanismo de física
- Áudio 3D

Deve-se notar que a lista acima não pretende ser completa. Existem muitos outros aspectos importantes para kits de ferramentas 3DUI, como abstrações de dispositivos de entrada, interface de programação e tratamento de erros (Takala et al., 2012).

2.2.1.1. Realidade Estendida (XR)

A Realidade estendida (XR – Extended Reality) é um termo genérico usado para descrever três tecnologias geradas por computador, incluindo realidade virtual, realidade aumentada e realidade mista (Ludlow, B. L., 2015).

Para Ong et al (2021) a capacidade da realidade estendida de criar ambientes que de outra forma seriam impossíveis no mundo real.

A realidade estendida é uma tecnologia emergente que tem sido utilizada em muitos domínios de alto desempenho, com o objetivo de melhorar as habilidades perceptivo-cognitivas e as habilidades motoras (Le Noury et al., 2022).

Embora as aplicações de XR sejam eminentemente visuais, outros sentidos estão sendo explorados na literatura em torno de interações multissensoriais, como feedback auditivo, olfativo ou tátil (Pons, P. et al., 2022).

Para Palmas et al (2020) as tecnologias XR podem ser resumidas da seguinte forma:

VR – Realidade virtual: é um ambiente virtual gerado por computador que permite que os usuários interajam, se movam e sejam completamente imersos em um ambiente virtual. Ao combinar diferentes tipos de hardware, experiências imersivas com recepção sensorial inteiramente simulada por computador podem ser alcançadas.

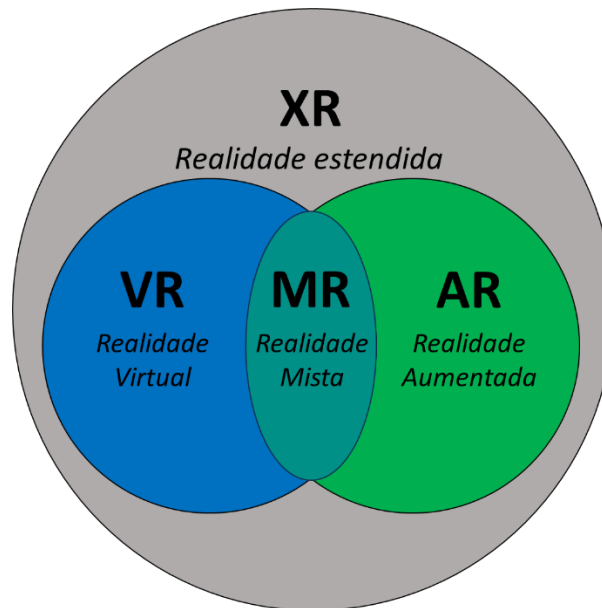
AR – Realidade aumentada: é representado por uma sobreposição de conteúdo digital em uma realidade física. Ele não pode interagir diretamente com o ambiente e é renderizado por um meio que exhibe o mundo real e o conteúdo digital simultaneamente, permitindo que o usuário experimente ambos ao mesmo tempo.

MR – Realidade mista: é uma forma híbrida de xR, criada pela combinação de realidade virtual e aumentada. Ela emprega uma sobreposição de conteúdo virtual que pode interagir com o ambiente real e, portanto, facilita a interação entre realidades como resultado da mistura do mundo físico e digital.

XR – Realidade estendida: une todos os três tipos mencionados de realidades alternativas (AR, VR, MR) e possíveis futuras (X), que ao longo do tempo expandirão o alcance das realidades, sob um único termo. Isso resulta em possibilidades futuras virtualmente ilimitadas para XR.

A Figura 7 ilustra como a realidade virtual (VR), realidade aumentada (AR) e realidade mista (MR) são parte do conceito maior de realidade estendida (XR).

Figura 7 — Comparativo XR: VR, AR e MR



Fonte: Elaboração própria, 2024.

Esta representação mostra claramente como a realidade virtual (VR), realidade aumentada (AR) e realidade mista (MR) se relacionam e são partes do conceito maior de realidade estendida (XR), com MR como uma combinação/interseção de VR e AR.

2.2.1.2. Realidade Aumentada (AR)

A realidade aumentada (AR) permite que os usuários experimentem recursos de um mundo virtual integrados com informações digitais por meio de um visor montado na cabeça ou outros sistemas de exibição, como smartphones. Dessa forma, os usuários podem interagir simultaneamente com o mundo real e com objetos virtuais (Eckert et al., 2019).

Na AR, as pessoas são capazes de enriquecer o mundo real com elementos virtuais, proporcionando uma fusão do digital com o físico (Tokarchuk et al., 2023). Segundo Fialho (2018), a realidade aumentada não nos transporta para mundos alternativos; ela simplesmente aumenta nosso estado de presença dentro da realidade cotidiana.

A Figura 8 apresenta um exemplo prático de realidade aumentada, onde existe um personagem fictício inserido em um ambiente real, isto sendo mostrado na tela do *smartphone*.

Figura 8 — Realidade aumentada



Fonte: Niantic, 2017.

2.2.1.3. Realidade Virtual (VR)

A Realidade Virtual (VR) é um ambiente tridimensional gerado por tecnologia que as pessoas podem experimentar com seus sentidos sincronizados, proporcionando uma sensação de imersão, existem diferentes tipos de VR, cada um exigindo equipamento de exibição específico (Vince, 2004). Ela faz uso de equipamentos especiais de realidade virtual, como óculos, capacetes, etc., que são equipados com sensores. Dependendo do nível de imersão, as simulações de RV podem ser categorizadas em três tipos: imersivas, semi-imersivas e totalmente imersivas (Brigham, 2017). Segundo Marks e Thomas (2022), a VR é uma tecnologia emergente que impacta diversos setores, incluindo a educação.

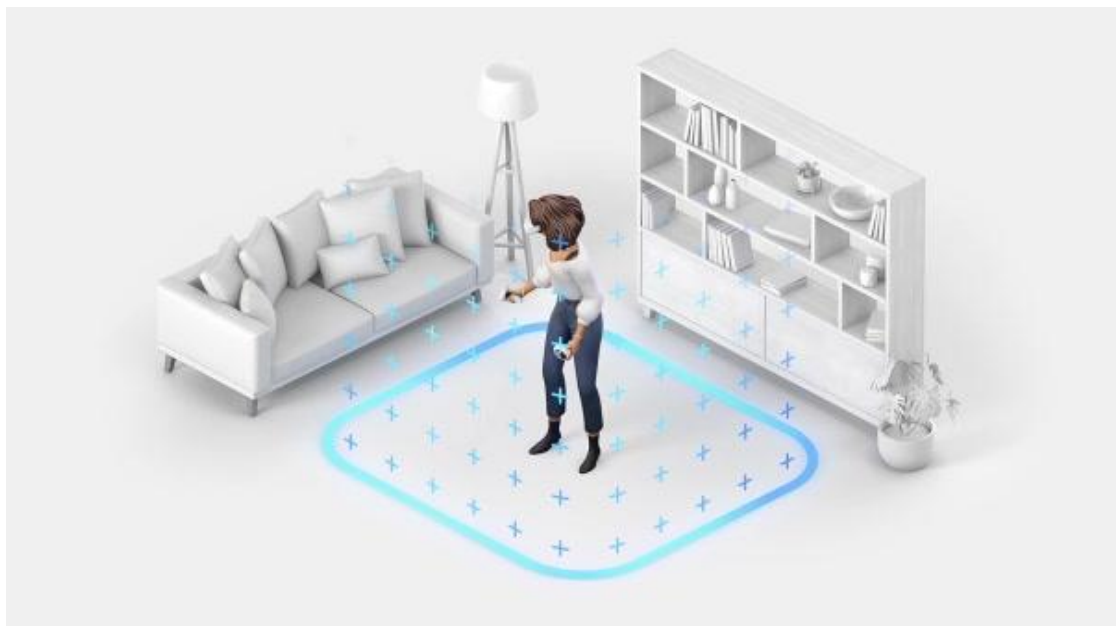
A realidade virtual fornece aos usuários uma representação visual da simulação, permitindo-lhes interagir com objetos no espaço 3D (Huang et al., 2019).

A interação humano-computador oferece vários modos de interação para a VR, de acordo com diferentes funções e propósitos (Tu et al., 2018).

A natureza imersiva da VR abre oportunidades únicas para a entrega de conteúdo educacional, desde visitas de campo virtuais até simulações científicas complexas (Dai, Garcia e Olave-Encina, 2023). Além disso, a VR é uma forma interativa de distração, onde o usuário se torna um participante ativo em um ambiente virtual (Wong et al., 2021).

A Figura 9 apresenta um exemplo prático de realidade virtual, onde existe um cenário com personagem fictício inserido em um ambiente virtual, isto sendo mostrado na tela do meta quest 2.

Figura 9 — Ambiente de realidade virtual



Fonte: Meta, 2019.

A realidade virtual tem sido amplamente utilizada, principalmente no campo da educação médica, pois permite simulação realista e está associada a uma melhor retenção de informações do que a leitura convencional de texto (Suh et al. 2005). A simulação de realidade virtual pode fornecer compreensão aprofundada do material educacional, com possibilidades de maior compreensão e aplicação na prática clínica. No geral, o objetivo principal da utilização da realidade virtual na educação é estimular os alunos para que possam aprender de forma mais eficaz (Yamakawa et al. 2020).

De acordo com uma revisão narrativa de 31 estudos, em comparação com outros métodos de ensino, como a educação digital online ou offline, a RV é mais eficaz em termos de atitudes de aprendizagem dos profissionais de saúde, satisfação de aprendizagem, relação custo-eficácia, formação clínica e mudança comportamental (Kyaw et al. 2019).

2.2.1.4. Realidade Mista (MR)

A realidade mista é uma combinação dos mundos físico e digital, revelando as interações naturais e intuitivas 3D entre humanos, computador e ambiente, essa nova realidade se baseia em aprimoramentos na pesquisa visual computacional, no processamento gráfico, na tecnologia de vídeo, em sistemas de entrada e na computação em nuvem (Microsoft, 2023).

A realidade mista é a combinação de tecnologias de RV e RA para construir um ambiente híbrido que mistura ambientes físicos e virtuais. Em tal ambiente, objetos físicos e virtuais coexistem e interagem no mesmo espaço, criando ilusões de realismo. Embora essa tecnologia possa criar uma experiência imersiva semelhante à realidade virtual, a interação com os objetos será mais intuitiva e natural. Ela traz o mundo virtual para o seu ambiente da vida real (Roo et al. 2017; Sanna et al. 2019).

Tanto os dispositivos de realidade aumentada como os de realidade virtual mostram-se promissores, no entanto, existem limitações significativas que retardaram a sua adoção generalizada. Estes incluem a incapacidade de interagir com pacotes de dados tridimensionais em realidade aumentada e a exclusão do ambiente do mundo real em realidade virtual. A realidade mista mescla muitos dos benefícios da realidade virtual e da realidade aumentada (Tepper et al. 2017).

A Figura 10 apresenta um exemplo de realidade mista, onde existe um cenário real com pessoas e alguns objetos fictícios virtuais posicionados em um ambiente real, isto sendo visto pelo dispositivo HoloLens da Microsoft.

Figura 10 — Ambiente de realidade virtual



Fonte: Microsoft, 2023.

2.2.2. Cirurgias em Realidade Virtual

As cirurgias em realidade virtual (VR) têm se tornado uma área de grande interesse devido à capacidade dessa tecnologia de proporcionar um ambiente controlado e seguro para o treinamento e a prática médica. A VR permite que cirurgiões e estudantes de medicina pratiquem procedimentos complexos em um ambiente simulado que replica a anatomia e as condições de uma cirurgia real, sem os riscos associados ao treinamento em pacientes reais.

Estudos demonstram que a realidade virtual é uma ferramenta eficaz no aprimoramento das habilidades cirúrgicas, permitindo a repetição de procedimentos e a correção de erros em tempo real. A imersão proporcionada pela VR possibilita uma resposta visual e tátil sincronizada à simulação, onde os usuários podem interagir com modelos tridimensionais de órgãos e tecidos, melhorando a compreensão e a precisão cirúrgica (Marks & Thomas, 2022).

Além disso, a utilização da VR em cirurgias oferece uma plataforma para a colaboração remota, onde especialistas de diferentes partes do mundo podem participar e oferecer orientações em tempo real durante uma simulação. Essa capacidade de telepresença e telementoria expande as possibilidades de treinamento, oferecendo acesso a expertise global (Tepper et al., 2017).

A realidade virtual não só melhora o treinamento, mas também está sendo explorada como uma ferramenta auxiliar em cirurgias reais, com a superposição de modelos 3D e guias visuais que auxiliam o cirurgião durante o procedimento. Isso proporciona maior precisão e pode levar a melhores resultados para os pacientes (Vávra et al., 2017).

Tepper et al (2017) apresenta na Figura 11 um cirurgião usando o óculos da Microsoft HoloLens na sala de cirurgia (esquerda) e a imagem vista pelo HoloLens (direita) mostrando a comparação do modelo holográfico de uma mandíbula ressecada com um modelo impresso 3D.

Figura 11 — Cirurgião com Microsoft HoloLens



Fonte: Tepper et al, 2017.

2.2.2.1. Resposta Tátil (Feedback Haptico)

O feedback háptico, ou resposta tátil, é uma tecnologia que simula a sensação de toque e resistência, permitindo que os usuários "sintam" objetos virtuais como se estivessem interagindo com objetos reais. Essa tecnologia é crucial em simulações médicas, especialmente em ambientes de realidade virtual, onde a imersão e a interação realista são fundamentais para o treinamento eficaz.

Em cirurgias simuladas, o feedback háptico oferece uma camada adicional de realismo, permitindo que os profissionais sintam a textura dos tecidos, a resistência ao corte, e a pressão necessária para manipular instrumentos cirúrgicos.

Estudos mostram que a adição de feedback háptico a simulações de VR melhora significativamente a precisão e a destreza dos cirurgiões em treinamento (Li et al., 2022).

A integração do feedback háptico com interfaces de usuário 3D (3DUI) e ambientes de realidade virtual representa um avanço significativo, possibilitando uma experiência mais rica e realista. Tecnologias de feedback háptico já estão sendo exploradas em dispositivos como bisturis virtuais, onde a resistência simulada pode replicar as sensações de cortar diferentes tipos de tecidos (Bektic et al., 2021).

A resposta tátil também desempenha um papel importante na correção de erros durante o treinamento, permitindo que os usuários sintam imediatamente se um procedimento foi realizado incorretamente e ajustem suas técnicas em tempo real. Isso não apenas melhora a aprendizagem, mas também pode reduzir os erros em cirurgias reais (González et al., 2020).

2.2.3. Internet Das Coisas (IoT)

A Internet das Coisas (IoT) é um conceito que descreve a interconexão de dispositivos físicos pela internet, permitindo a coleta e troca de dados entre eles. Esses dispositivos, que incluem sensores, atuadores, e outros componentes eletrônicos, estão equipados com tecnologias de comunicação que os tornam capazes de interagir com sistemas computacionais mais amplos. A IoT tem revolucionado diversos setores, incluindo a saúde, a automação industrial, a agricultura e as cidades inteligentes, ao permitir que informações em tempo real sejam acessadas e utilizadas para decisões mais precisas e eficientes (Taleb et al., 2023).

Na área da saúde, a IoT deu origem ao conceito de *Internet of Medical Things* (IoMT), que se refere à rede de dispositivos médicos conectados, como monitores de sinais vitais, dispositivos de diagnóstico remoto e sistemas de gestão hospitalar. Esses dispositivos permitem um monitoramento contínuo e personalizado dos pacientes, melhorando a qualidade do atendimento e reduzindo custos operacionais. Além disso, a IoT facilita a integração de dados entre diferentes sistemas, promovendo uma visão holística das operações de saúde e permitindo intervenções mais rápidas e precisas (Li et al., 2022).

No contexto de dispositivos médicos e cirúrgicos, a IoT possibilita o desenvolvimento de ferramentas inteligentes que podem se comunicar com outras plataformas e sistemas, fornecendo feedback em tempo real e ajustando parâmetros operacionais conforme necessário. Isso é especialmente relevante em ambientes de realidade virtual e aumentada, onde a precisão e a integração de dados são essenciais para simulações realistas e eficazes (Xu et al., 2023).

2.2.3.1. Bluetooth Low Energy

O Bluetooth Low Energy (BLE) é uma tecnologia de comunicação sem fio projetada para fornecer conectividade com baixo consumo de energia, mantendo uma longa vida útil da bateria dos dispositivos que o utilizam. Originalmente introduzido como parte do padrão Bluetooth 4.0, o BLE foi desenvolvido especificamente para aplicações de IoT, onde a comunicação eficiente e de baixo consumo é crucial (Nordic, 2024).

Operando na banda de 2,4 GHz, o BLE é capaz de transmitir dados em curtas distâncias com um consumo de energia significativamente menor em comparação ao Bluetooth clássico. Essa eficiência energética torna o BLE ideal para dispositivos que precisam operar por longos períodos sem recarga, como sensores vestíveis, monitores de saúde e dispositivos domésticos inteligentes (SEED Studio, 2023).

No desenvolvimento de dispositivos cirúrgicos para ambientes de realidade virtual, como o bisturi virtual descrito neste trabalho, o BLE desempenha um papel fundamental na comunicação entre o dispositivo e a plataforma de realidade virtual, permitindo a transmissão de dados sensoriais em tempo real com baixa latência. Essa capacidade é crucial para manter a imersão e a precisão durante as simulações, garantindo que o feedback dos sensores seja imediatamente refletido no ambiente virtual (Li et al., 2022).

Além disso, o BLE permite a criação de redes de sensores e atuadores que podem ser facilmente integradas em sistemas maiores, fornecendo um meio eficiente de coleta e processamento de dados em tempo real. A tecnologia continua a evoluir, com melhorias na segurança, alcance e capacidade de dados, tornando-se uma escolha cada vez mais comum em aplicações de IoT e dispositivos médicos conectados (Xu et al., 2023).

2.2.4. Acelerômetro

O acelerômetro é um dispositivo que mede a aceleração linear de um objeto em uma ou mais direções. Esses sensores são amplamente utilizados em diversas aplicações, desde dispositivos móveis até sistemas de navegação e equipamentos médicos. A função principal de um acelerômetro é detectar variações na velocidade e direção do movimento, o que é crucial para a estabilidade e controle em sistemas dinâmicos, como os encontrados em dispositivos médicos e de realidade virtual.

Existem dois tipos principais de acelerômetros: digitais e analógicos. Os acelerômetros analógicos produzem uma saída de tensão contínua que é proporcional à aceleração medida. Um exemplo popular é o ADXL335, um acelerômetro de 3 eixos que fornece leituras precisas e é amplamente utilizado em projetos de hobby e aplicações industriais leves (Analog Devices, 2022).

Por outro lado, os acelerômetros digitais fornecem dados de aceleração em formato digital, geralmente via interfaces como I2C ou SPI, o que facilita a integração com microcontroladores e outros sistemas digitais. Um exemplo de acelerômetro digital é o MPU-6050, que combina um acelerômetro e um giroscópio de 3 eixos, oferecendo alta precisão e facilidade de uso em projetos que exigem controle de movimento (InvenSense, 2023). Este tipo de acelerômetro é comumente utilizado em aplicações de realidade virtual e aumentada, onde a detecção precisa do movimento é essencial para a criação de ambientes imersivos.

A utilização de acelerômetros em dispositivos médicos e cirúrgicos, como o bisturi para realidade virtual, permite a detecção e a análise de movimentos em tempo real, contribuindo para a precisão e o controle durante os procedimentos simulados. Isso é particularmente importante em sistemas que exigem um alto grau de sensibilidade ao movimento, garantindo que as simulações reflitam com precisão as condições reais (Xu et al., 2023).

2.2.5. Giroscópio

O giroscópio é um dispositivo que utiliza a conversão de momento angular para determinar a rotação de um objeto em torno de um ou mais eixos. Assim como o acelerômetro, o giroscópio é um sensor fundamental em sistemas de navegação, estabilização e controle de movimento.

Ele é frequentemente utilizado em conjunto com acelerômetros para fornecer uma estimativa completa da orientação e do movimento de um objeto no espaço tridimensional.

Existem giroscópios analógicos e digitais, que diferem na forma como transmitem os dados medidos. Os giroscópios analógicos produzem uma saída de tensão proporcional à taxa de rotação medida. Um exemplo comum é o ADXRS453, que é amplamente utilizado em aplicações de navegação e estabilização, especialmente em sistemas que requerem alta precisão e baixa latência (Analog Devices, 2022).

Os giroscópios digitais, por sua vez, fornecem os dados em formato digital, permitindo uma integração mais direta com sistemas de controle digitais. O MPU-6050, mencionado anteriormente como exemplo de acelerômetro digital, também inclui um giroscópio de 3 eixos, tornando-o uma escolha popular para aplicações que exigem a medição precisa tanto da aceleração quanto da rotação. Outro exemplo de giroscópio digital é o L3GD20, que oferece alta sensibilidade e precisão em uma ampla gama de aplicações, desde dispositivos móveis até sistemas de estabilização em drones (STMicroelectronics, 2023).

Em dispositivos médicos e cirúrgicos, como o bisturi virtual desenvolvido para realidade virtual, os giroscópios são utilizados para monitorar e corrigir a orientação e a rotação do instrumento, garantindo que os movimentos realizados no ambiente virtual correspondam com precisão aos movimentos desejados. Isso é crucial para criar simulações realistas e para treinar profissionais de saúde em ambientes controlados e seguros (Vávra et al., 2017).

3 MATERIAIS E MÉTODOS

A seguir serão apresentados os materiais e métodos utilizados no desenvolvimento do bisturi e sala de cirurgia em ambiente de realidade virtual.

Para o desenvolvimento do bisturi cirúrgico háptico integrado com realidade virtual, exigiu a seleção de materiais e metodologias para garantir precisão, responsividade e imersão. Neste capítulo, são apresentados os componentes utilizados no projeto, assim como os métodos empregados.

O sistema proposto combina hardware e software de maneira integrada, utilizando sensores inerciais, motores de vibração e comunicação *Bluetooth Low Energy* (BLE) para permitir um controle preciso do bisturi virtual e fornecer feedback tátil realista ao usuário. Dessa forma, a escolha dos componentes foi baseada na compatibilidade, eficiência energética e desempenho necessário para uma simulação imersiva e interativa.

A seguir, serão detalhados os materiais utilizados na construção do protótipo do bisturi, bem como as metodologias empregadas para garantir a aquisição e validação dos dados de movimento, além da comunicação eficiente com o ambiente de realidade virtual no Meta Quest 2.

3.1. DESCRIÇÃO DOS COMPONENTES UTILIZADOS

Para o desenvolvimento do dispositivo, foram utilizados os seguintes materiais:

- 1 Meta Quest 2
- 1 Módulo Seeed Xiao Ble Sense Nrf52840
- 1 Módulo Acelerômetro analógico Adxl335
- 1 Trimpot 10k Ω
- 1 Bateria 3,7V 500mAh 501240
- 4 Push Buttons
- 6 Resistores 10k Ω
- 2 Transistores NPN
- 2 Diodos 1N4007
- 2 Mini motores de vibração
- 1 Protoboard de 400 pinos
- Jumpers

Cada um desses componentes possui uma função específica no desenvolvimento do dispositivo, sendo essencial para o seu correto funcionamento. Nas subseções a seguir, serão detalhados os principais componentes utilizados, abordando suas características técnicas e seu papel.

3.1.1. Meta Quest 2

O Meta Quest 2, anteriormente conhecido como Oculus Quest 2, é um dispositivo de realidade virtual (VR) desenvolvido pela Meta, lançado em outubro de 2020. Ele é um headset autônomo, ou seja, não requer um PC ou console para funcionar, embora possa ser conectado a um computador para experiências de VR mais complexas, como jogos ou aplicativos que demandam maior capacidade gráfica.

Um dos principais diferenciais do Meta Quest 2 é sua combinação de preço acessível e alta qualidade de hardware. O dispositivo conta com um processador Qualcomm Snapdragon XR2 e 6 GB de RAM, permitindo a execução de aplicações de VR mais exigentes. Além disso, possui uma tela de alta resolução, com 1832 x 1920 pixels por olho, o que melhora significativamente a imersão visual e a nitidez das imagens (Meta, 2020).

O Meta Quest 2 também utiliza um sistema de rastreamento de movimentos sem a necessidade de sensores externos, empregando câmeras integradas ao dispositivo para mapear o espaço ao redor do usuário. Este sistema de *inside-out tracking* permite que o usuário se mova livremente em um ambiente físico, enquanto seus movimentos são replicados com precisão no ambiente virtual (Meta, 2020). O headset também inclui controladores de movimento que oferecem interações precisas com objetos virtuais, sendo essenciais para simulações em campos como entretenimento, educação e treinamentos profissionais, incluindo práticas cirúrgicas em VR.

Adicionalmente, o dispositivo possui um sistema de áudio espacial integrado, proporcionando uma experiência sonora imersiva sem a necessidade de fones de ouvido externos (Meta, 2021). Este recurso é essencial para experiências mais imersivas, uma vez que permite ao usuário captar sons com precisão direcional dentro dos ambientes virtuais.

Para expandir a aplicabilidade do Quest 2 em cenários mais robustos, a Meta oferece o Oculus Link, um recurso que permite conectar o Quest 2 a um PC através de um cabo USB-C, transformando-o em um headset tethered capaz de executar conteúdos de VR mais exigentes, como jogos de alta performance ou aplicações profissionais de design 3D e simulações cirúrgicas. Isso amplia a flexibilidade do dispositivo, permitindo que ele atenda tanto usuários casuais quanto desenvolvedores e profissionais que necessitam de maior poder computacional (Meta, 2021).

Headsets de RV como o Meta Quest 2 permitem executar a renderização de aplicativos localmente ou explorar o descarregamento para um servidor na nuvem e, em um cenário de gerenciamento de desastres, o consumo de energia do dispositivo é um conhecimento fundamental para fazer escolhas racionais sobre qual dos dois tipos de renderização executar (L. Romagnoli et al., 2023).

Podemos visualizar a imagem do dispositivo Meta Quest 2 na Figura 12.

Figura 12 — Meta Quest 2



Fonte: Meta, 2019.

3.1.2. Seeed Xiao Ble Sense Nrf52840

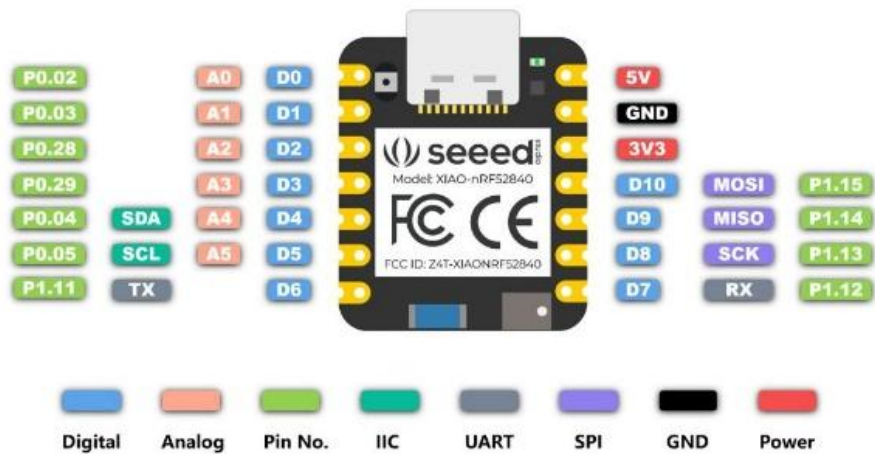
A Seeed XIAO BLE Sense é uma placa de desenvolvimento baseada no controlador nRF52840 ARM Cortex-M4F da Nordic Semiconductor, possui LEDs integrados, porta USB, flash QSPI, comunicação *Bluetooth Low Energy*, acelerômetro 3D e giroscópio 3D digital (LSM6DS3TR-C 6D), carregador de bateria (BQ25101) e uma variedade de entradas e saídas divididas em 11 pinos, além de outros recursos suportados, conforme mostra as Figuras 13 e 14 (Nordic, 2024; Seeed Studio, 2023).

Figura 13 — Recursos suportados

Interface	Controller	Driver/Component
ADC	on-chip	adc
CLOCK	on-chip	clock_control
FLASH	on-chip	flash, QSPI flash
GPIO	on-chip	gpio
I2C(M)	on-chip	i2c
MPU	on-chip	arch/arm
NVIC	on-chip	arch/arm
PWM	on-chip	pwm
RADIO	on-chip	Bluetooth, ieee802154
RTC	on-chip	system clock
SPI(M/S)	on-chip	spi
UART	on-chip	serial
USB	on-chip	usb
WDT	on-chip	watchdog

Fonte: Nordic, 2024.

Figura 14 — Pinagem XIAO Seeed nRF52840 Sense.



Fonte: Seeed Studio, 2023.

A placa Seeed XIAO nRF52840 Sense é uma pequena placa de desenvolvimento com as seguintes dimensões: 21 mm x 17,5 mm, possui conectividade Bluetooth 5.0. Tendo um formato compacto que pode ser usado para dispositivos vestíveis e projetos de Internet das Coisas (Seeed Studio, 2023). A Figura 15 apresenta a placa Seeed XIAO nRF52840 Sense.

Figura 15 — Módulo XIAO Seeed nRF52840 Sense



Fonte: Seeed Studio, 2023.

A escolha do SEEED XIAO BLE Sense nRF52840 como unidade de processamento principal do bisturi cirúrgico foi fundamentada em suas características técnicas avançadas e compatibilidade com o sistema proposto. O módulo oferece um balanço ideal entre desempenho, consumo energético e integração com dispositivos de realidade virtual, sendo essencial para a comunicação em tempo real e a resposta tátil do bisturi. Entre os principais fatores que justificam sua seleção, destacam-se:

- Conectividade *Bluetooth Low Energy* (BLE 5.0).
- Processamento otimizado para sensores inerciais.
- Sensores embutidos (acelerômetro e giroscópio - LSM6DS3TR-C).
- Tamanho compacto e consumo energético reduzido.
- Capacidade de controle de atuadores (motores de vibração).
- Compatibilidade com o desenvolvimento em Unity e BLE GATT.

3.1.3. ADXL335

O ADXL335 é um acelerômetro analógico triaxial de baixa potência fabricado pela Analog Devices, amplamente utilizado em projetos de engenharia e sistemas embarcados para medir aceleração linear em três eixos: X, Y, e Z. Este componente é capaz de detectar tanto movimentos estáticos (como inclinações) quanto dinâmicos (como choques e vibrações), e possui características que o tornam adequado para uma variedade de aplicações, desde dispositivos portáteis até sistemas de controle de movimento.

O ADXL335 opera com uma faixa de tensão de 1,8V a 3,6V e consome tipicamente 350 μ A, o que o torna ideal para sistemas de baixa potência, como dispositivos móveis e wearables. Ele fornece uma faixa dinâmica de medição de $\pm 3g$ (onde "g" é a aceleração gravitacional), e a saída de cada eixo é proporcionada por três pinos analógicos, cujos valores de tensão variam em função da aceleração detectada.

A principal característica deste acelerômetro é sua saída analógica, o que significa que os sinais gerados podem ser lidos diretamente por conversores analógico-digital (ADC) de microcontroladores. A saída analógica facilita a integração com esses sistemas, permitindo que os valores de aceleração sejam monitorados em tempo real e utilizados em algoritmos de controle ou análise de dados.

O ADXL335 opera com o princípio básico de medir a força exercida pela aceleração em pequenas estruturas internas do sensor. Cada eixo do acelerômetro possui um capacitor móvel cujo movimento em resposta à aceleração provoca mudanças em sua capacitância, que são convertidas em sinais elétricos processados pelo circuito interno do componente. O sensor pode detectar movimentos em uma faixa de frequência de até 1600 Hz nos eixos X e Y, e até 550 Hz no eixo Z, proporcionando uma resposta rápida a mudanças de movimento (Analog Devices, 2021). A Figura 16 apresenta um módulo com o ADXL335.

Figura 16 — Módulo ADXL335 (EVAL-ADXL335)



Fonte: Analog Devices, 2021.

A escolha do ADXL335, como sensor principal para o bisturi, foi baseada em características específicas que o tornam mais adequado para a aplicação em comparação com o acelerômetro digital embutido no SEED XIAO BLE Sense nRF52840 (LSM6DS3TR-C). A seguir, são destacadas as razões para essa escolha e as vantagens do ADXL335 para este projeto.

- Precisão e Estabilidade nas Leituras Analógicas.
- Latência Menor e Resposta Mais Rápida.
- Maior Controle sobre o Processamento dos Dados.
- Ruído Menor e Melhor Sensibilidade para Pequenos Movimentos.
- Faixa de Medição Adequada.

3.1.4. LSM6DS3TR-C

O chip LSM6DS3TR-C, é um sensor de movimento fabricado pela STMicroelectronics. Ele é um módulo IMU - *Inertial Measurement Unit* (Unidade de medida inercial) que combina um acelerômetro triaxial e um giroscópio triaxial em um único encapsulamento. O acelerômetro mede a aceleração linear nos eixos X, Y, e Z. Ele pode detectar variações de movimento e inclinação, possibilitando funcionalidades como detecção de quedas ou controle de orientação. O giroscópio mede a taxa de rotação angular em torno dos três eixos, fornecendo informações sobre o movimento rotacional do dispositivo.

Este sensor é otimizado para baixo consumo de energia, o que o torna adequado para dispositivos alimentados por bateria. Isso é crítico em aplicações como dispositivos portáteis, onde a eficiência energética é fundamental. Segundo a STMicroelectronics (2023), ele possui modos de operação configuráveis, permitindo ajustar o consumo de acordo com os requisitos da aplicação.

O LSM6DS3TR-C é projetado para fornecer medições de alta precisão, com uma faixa dinâmica de aceleração que varia de $\pm 2g$ a $\pm 16g$ e uma faixa de sensibilidade angular de ± 125 dps a ± 2000 dps. Essa versatilidade permite que ele seja utilizado em uma ampla gama de aplicações que requerem detecção precisa de movimento.

Além disso, ele pode ser configurado para gerar interrupções de movimento quando certos eventos, como quedas ou movimentos bruscos, são detectados, o que reduz a carga de processamento no microcontrolador principal.

A Figura 17 ilustra o encapsulamento do chip LSM6DS3TR-C, utilizado no módulo escolhido.

Figura 17 — Chip LSM6DS3TR-C



Fonte: ST, 2017.

3.2. FUNCIONALIDADE DOS COMPONENTES

O meta quest 2 é o dispositivo de realidade virtual que irá simular a sala de cirurgia e imergir o usuário ao ambiente imersivo.

A placa controladora Sseeed Xiao Ble Sense Nrf52840 é responsável por coletar todas as informações processar e enviar via *Bluetooth Low Energy* para o óculos de realidade virtual Meta Quest 2.

O Módulo Acelerômetro analógico Adxl335 é responsável por medir a aceleração em X, Y, e Z. Então gerar um sinal analógico proporcional ao que foi medido de aceleração em cada eixo.

Trimpot de 10k Ω é responsável por controlar a velocidade do cursor.

A Bateria 3,7V 500mAh (501240) é a fonte de energia do circuito.

Os 4 Push Buttons são a interface de controle para o usuário executar ações no ambiente de realidade virtual.

Os Resistores de 10k Ω são para técnica de *pull-down* nos botões e 2 são para base dos transistores.

Os 2 Transistores NPN são necessários para acionar os mini motores de vibração.

Os 2 Diodos 1N4007 são para eliminar a tensão reversa do motor quando é parado.

Os 2 Mini motores de vibração são para feedback tátil.

A Protoboard de 400 pinos será a interface para conexão de todos os componentes eletrônicos, exceto o Meta Quest 2.

Os Jumpers são para conexão dos componentes.

3.3. AMBIENTE UNITY

O Unity é uma plataforma de desenvolvimento amplamente utilizada para a criação de experiências interativas em tempo real, como jogos, simulações e aplicações de realidade virtual (VR) e aumentada (AR). Originalmente lançada em 2005 pela Unity Technologies, a engine Unity se destaca por sua flexibilidade, escalabilidade e facilidade de uso, sendo atualmente uma das ferramentas mais populares no desenvolvimento de ambientes virtuais imersivos (Unity, 2023).

Uma das principais vantagens do Unity é seu suporte multiplataforma, permitindo que aplicações sejam desenvolvidas uma única vez e implementadas em uma ampla variedade de dispositivos, como PC, consoles, dispositivos móveis, e óculos de realidade virtual como o Meta Quest 2. Essa capacidade de exportação multiplataforma torna o Unity uma ferramenta eficiente para desenvolvedores que buscam atingir diferentes mercados com um único código-base (Cervantes, 2021).

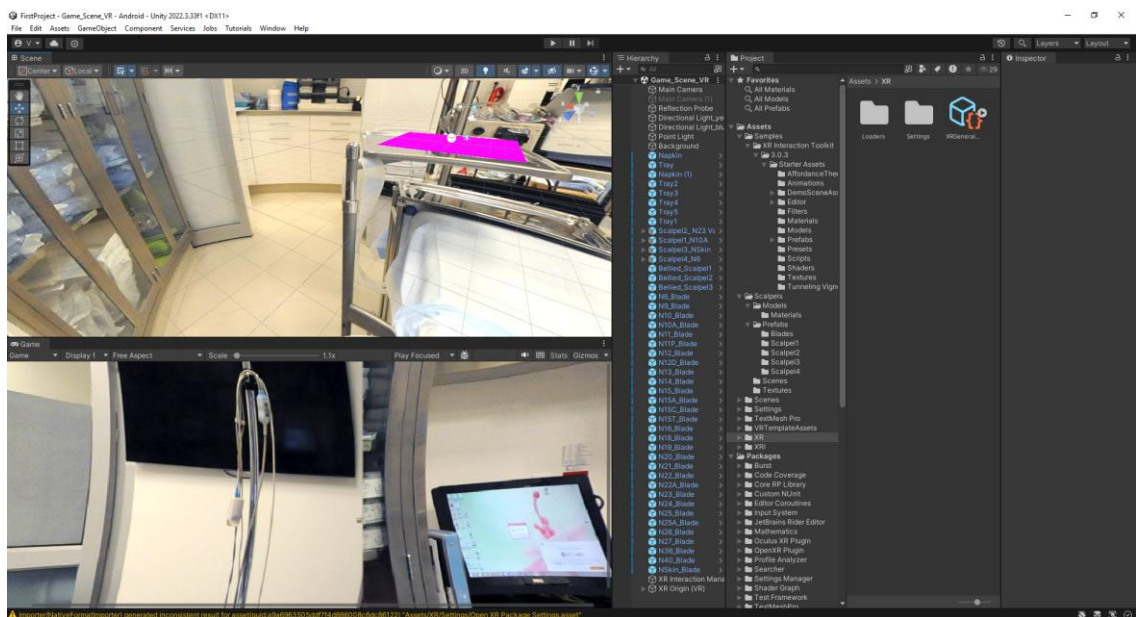
O Unity oferece um ambiente de desenvolvimento integrado robusto, com uma interface intuitiva que combina ferramentas gráficas e de codificação, facilitando a criação de ambientes 3D e 2D. Sua linguagem de script é o C#, o que possibilita a criação de interações complexas e comportamentos personalizados para objetos no ambiente virtual (McFadden, 2020). Existe uma série de pacotes de integração VR, como o XR *Interaction Toolkit*, que facilita a configuração de interações complexas com objetos virtuais através de dispositivos de controle, como controladores manuais e sistemas de rastreamento (Unity, 2022).

Para o desenvolvimento de simuladores cirúrgicos ou outras aplicações médicas em realidade virtual, o Unity oferece suporte a renderizações gráficas de alta qualidade, física avançada e interações em tempo real, essenciais para criar simulações realistas e interativas. O uso do feedback háptico em conjunto com o Unity, por exemplo, pode melhorar ainda mais a imersão do usuário, oferecendo sensações físicas em resposta às interações no ambiente virtual (Thompson & Smith, 2021). Além disso, a plataforma permite a integração com bibliotecas externas e APIs especializadas, como sistemas de inteligência artificial e renderização avançada, o que o torna ideal para projetos que exigem soluções personalizadas (Goldstone, 2020).

Em projetos como o desenvolvimento de simulações cirúrgicas, o Unity oferece a flexibilidade e os recursos necessários para criar uma experiência imersiva, interativa e realista.

A Figura 18 apresenta a utilização do Unity no desenvolvimento de um ambiente de realidade virtual imersiva voltado para aplicações médicas, especificamente em simulações de procedimentos cirúrgicos, também mostra uma cena de simulação médica em realidade virtual. Ilustrando o editor da Unity com uma cena em VR composta por objetos médicos e interativos, permitindo o desenvolvimento e teste de interações no ambiente simulado. Na janela de pré-visualização, pode-se observar diferentes perspectivas do cenário, configurando a visão do usuário durante a simulação. À direita, está o painel de hierarquia de objetos, onde os elementos 3D, scripts e materiais que compõem a cena são organizados e gerenciados.

Figura 18 — Captura de tela do ambiente de desenvolvimento Unity



Fonte: Unity, 2023.

3.4. PROTOCOLO DE COMUNICAÇÃO BLE

O *Bluetooth Low Energy* (BLE) é uma tecnologia sem fio projetada para permitir comunicação de dados entre dispositivos com baixo consumo de energia. Desde sua introdução com o Bluetooth 4.0, o BLE tem sido amplamente utilizado em dispositivos como *wearables*, sensores, equipamentos médicos, e dispositivos de realidade virtual, como o Meta Quest 2. A grande vantagem do BLE sobre o Bluetooth clássico é sua eficiência energética, sendo ideal para dispositivos que operam com baterias pequenas ou que precisam durar longos períodos sem recarga frequente.

Características do BLE:

- **Baixo Consumo de Energia:** BLE foi projetado para ser utilizado por dispositivos que precisam de operações de longo prazo, minimizando o consumo de energia durante a transmissão de dados. Dispositivos alimentados por baterias, como sensores e módulos de controle, podem operar por meses ou até anos em uma única carga (Qodenext, 2023; Bluetooth, 2024).
- **Baixa Latência:** A tecnologia BLE tem latência baixa, o que significa que a conexão entre os dispositivos é estabelecida rapidamente, permitindo uma comunicação quase instantânea. Isso é crucial para aplicações como realidade virtual, onde é essencial que os comandos e respostas sensoriais aconteçam em tempo real (Stormotion, 2024).
- **Alcance e Velocidade:** A taxa de dados do BLE pode atingir até 2 Mbps, dependendo da versão do Bluetooth utilizada, e o alcance típico pode variar de 10 a 30 metros, embora dispositivos modernos com Bluetooth 5 possam atingir até 400 metros sob condições ideais (Novel Bits, 2023).

O BLE oferece várias vantagens importantes, especialmente em aplicações de realidade virtual:

- **Eficiência energética:** O BLE permite que os dispositivos transmitam dados com consumo mínimo de energia, garantindo que dispositivos portáteis, como o módulo Seeed XIAO BLE Sense nRF52840, funcionem por longos períodos sem a necessidade de recarga frequente (Stormotion, 2024).

- Conectividade em Tempo Real: Com uma latência mínima, o BLE facilita a interação em tempo real entre os dispositivos, o que é fundamental para garantir que as ações no mundo real sejam refletidas instantaneamente no ambiente virtual (Bluetooth, 2024).
- Custo-benefício: O hardware necessário para habilitar a comunicação BLE é relativamente barato, o que o torna uma escolha ideal para dispositivos de larga escala, como wearables e sistemas VR (Qodenext, 2023).

O módulo Seeed XIAO BLE Sense nRF52840 utiliza BLE para se comunicar com o Meta Quest 2, permitindo que dados sensoriais, como a aceleração captada pelo acelerômetro, sejam transmitidos para o headset VR em tempo real. Além disso, a ativação dos motores de vibração para proporcionar feedback háptico ao usuário também é controlada via BLE, garantindo uma experiência de imersão mais completa. A transmissão de dados ocorre de forma eficiente, sem comprometer a performance do sistema, e o baixo consumo de energia garante a operação contínua do dispositivo durante longos períodos, essencial para aplicações que exigem autonomia e mobilidade (Qodenext, 2023; Bluetooth, 2024).

O BLE também suporta conexões seguras, utilizando criptografia AES de 128 bits para garantir que a comunicação entre o módulo e o Meta Quest 2 seja protegida, prevenindo acessos não autorizados durante as simulações (Bluetooth, 2024).

A utilização do *Bluetooth Low Energy* no sistema permite que os dados dos sensores sejam transmitidos de forma eficiente e com baixo consumo de energia, enquanto proporciona uma experiência de imersão sem interrupções para o usuário. A capacidade de sincronizar ações e fornecer feedback tátil em tempo real torna o BLE a solução ideal para simulações médicas em realidade virtual, aumentando a interatividade e a precisão da plataforma (Stormotion, 2024).

3.4.1. Configuração e Integração com Meta Quest 2

O Unity é amplamente utilizado para desenvolvimento de aplicações de realidade virtual, e sua compatibilidade com o Meta Quest 2 facilita a criação e integração de simulações interativas, como ambientes cirúrgicos. O Meta Quest 2 usa a plataforma Oculus (Meta XR) para integração, que pode ser configurada com pacotes específicos no Unity, como XR Interaction Toolkit e Oculus XR Plugin.

XR Plugin Management é utilizado para gerenciar diferentes plataformas de XR (Realidade Estendida), permitindo que o Unity construa a aplicação diretamente para o Meta Quest 2.

Instalação de Pacotes Necessários no Unity

- XR Plugin Management: A primeira etapa é ativar o XR Plugin Management para habilitar suporte a dispositivos VR. Isso é feito por meio das configurações de Project Settings do Unity, onde o suporte para dispositivos VR é ativado e o plugin Oculus é instalado.
- XR Interaction Toolkit: Pacote que facilita a interação do usuário com objetos virtuais em RV, sendo útil para capturar a manipulação do bisturi, feedback tátil e interação com o ambiente virtual.
- Meta Quest SDK: O Meta Quest SDK (antigo Oculus SDK) é integrado ao Unity para fornecer suporte à plataforma Meta Quest 2, permitindo o uso de funcionalidades específicas, como rastreamento de mão e controle háptico.

Configuração da Cena no Unity

- O XR Rig é o objeto que representa o ponto de vista do usuário no ambiente de realidade virtual, controlando a posição da câmera e a interação com os controladores. No caso do Meta Quest 2, é fundamental configurar o XR Rig corretamente para capturar o movimento do bisturi e a interação do usuário no ambiente cirúrgico.
- Para integrar o bisturi à simulação, o modelo tridimensional do bisturi precisa ser colocado de forma precisa no ambiente virtual. O XR Rig permite que o usuário veja o ambiente a partir de uma perspectiva em primeira pessoa, onde ele poderá usar o bisturi com controles de movimento.
- Controle e Feedback Tátil: O bisturi, utilizando o controlador desenvolvido com o módulo SEEED XIAO nRF52840, se comunica via BLE com o Meta Quest 2. A integração do feedback tátil (vibração) e controle de posição é feita utilizando os dados capturados pelos sensores (acelerômetro e giroscópio) no bisturi, transmitidos para o ambiente Unity.

O Unity pode receber esses dados e aplicar ações no ambiente, como simular cortes em um modelo de paciente virtual, baseando-se na posição e movimento do bisturi, além de controlar o motor de vibração para fornecer feedback ao usuário.

Configuração de Build para Meta Quest 2

- **Android como Plataforma:** O Meta Quest 2 utiliza um sistema operacional baseado em Android, portanto, é necessário configurar o Unity para exportar a build como uma aplicação Android.
- **Configuração do Build Settings:** Definir a plataforma como Android e ajustar as configurações de compatibilidade com o Meta Quest 2. Isso inclui definir o Scripting Backend como IL2CPP e selecionar o formato de arquitetura ARM64, além de garantir que a versão mínima do SDK Android seja compatível com o Meta Quest.
- **Deploy para Meta Quest 2:** Após a configuração, a build pode ser transferida diretamente para o dispositivo Meta Quest 2 via ADB (Android Debug Bridge) ou carregada na loja Oculus para testes.

Testes e Ajustes de Desempenho

- **Otimização Gráfica:** É importante garantir que a aplicação funcione com uma taxa de quadros estável no Meta Quest 2, visando ao menos 72 FPS para uma experiência fluida. Ajustes de post-processing e redução de polígonos em modelos 3D podem ser necessários para atingir o desempenho desejado.
- **Interação e Responsividade:** Testes contínuos são necessários para ajustar a sensibilidade do bisturi virtual, garantindo que o corte e o feedback tátil respondam de forma precisa aos movimentos do usuário.

4 DESENVOLVIMENTO

A seguir serão apresentados os componentes que compõem a arquitetura do sistema de transmissão de dados via Bluetooth Low Energy.

A implementação do bisturi integrado com o ambiente de realidade virtual requer uma abordagem estruturada que englobe hardware, firmware e software, garantindo a sincronização entre o dispositivo físico e o ambiente virtual. O desenvolvimento do sistema foi baseado em um conjunto de componentes eletrônicos, protocolos de comunicação *Bluetooth Low Energy* (BLE) e integração com o ambiente Unity para simulação cirúrgica no Meta Quest 2.

Este capítulo apresenta as etapas de construção do protótipo, desde o projeto do circuito eletrônico, desenvolvimento da placa de circuito impresso (PCI), firmware para processamento de sensores e comunicação BLE, até a configuração do ambiente virtual e a implementação de scripts para integração do bisturi físico com a simulação em VR.

A abordagem de desenvolvimento seguiu as seguintes etapas:

1. Projeto e montagem do hardware – Definição dos componentes, elaboração do circuito eletrônico e fabricação da PCI.
2. Desenvolvimento do firmware – Implementação do código embarcado para captura de dados dos sensores, acionamento dos motores de vibração e comunicação BLE.
3. Configuração do ambiente de realidade virtual – Implementação da sala de cirurgia no Unity, modelagem do bisturi virtual e integração com o Meta Quest 2.
4. Implementação dos scripts BLE no Unity – Desenvolvimento de código para comunicação entre o bisturi e a simulação, garantindo que os movimentos do bisturi físico sejam refletidos corretamente no ambiente virtual.

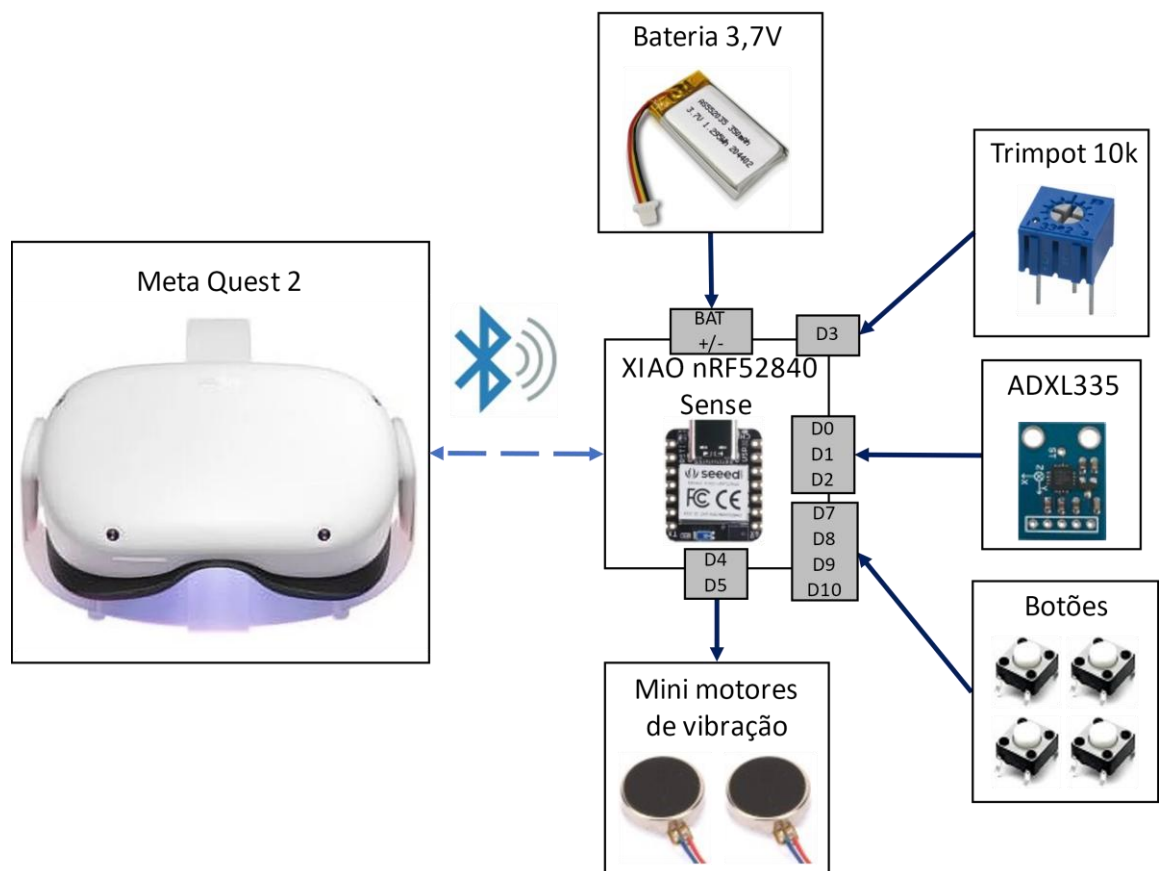
Além disso, para garantir baixo consumo de energia e comunicação eficiente, o protocolo *Bluetooth Low Energy* (BLE) foi utilizado para a transmissão de dados entre o bisturi e o Meta Quest 2. Isso possibilita que informações como movimentos do bisturi, acionamento de botões e feedback tátil via motores de vibração sejam enviados em tempo real, criando uma experiência de simulação cirúrgica imersiva e realista.

Os próximos tópicos detalham cada etapa do desenvolvimento, desde o diagrama do circuito até a implementação dos scripts e testes de validação do sistema.

4.1. DIAGRAMA DO CIRCUITO

A Figura 19 apresenta o diagrama do circuito que ilustra a integração dos componentes eletrônicos com o dispositivo de realidade virtual Meta Quest 2 por meio de comunicação *Bluetooth Low Energy*. O sistema é composto por uma placa microcontroladora Sseeed XIAO BLE Sense nRF52840, sensores, atuadores e uma bateria de alimentação.

Figura 19 — Diagrama de blocos



Fonte: Elaboração própria, 2024.

O diagrama da Figura 19 ilustra a estrutura do sistema de simulação médica em realidade virtual imersiva, com o Meta Quest 2 como interface principal. O microcontrolador Sseeed XIAO BLE Sense nRF52840 é o responsável por gerenciar a comunicação via *Bluetooth Low Energy* (BLE), recebendo e enviando dados de sensores e atuadores para o ambiente virtual em tempo real.

O Sseeed XIAO BLE Sense nRF52840, localizado no centro do diagrama, conecta-se aos seguintes componentes principais:

- Acelerômetro ADXL335 (portas D0, D1 e D2): Captura a aceleração nos eixos X, Y e Z, permitindo o monitoramento de movimentos e inclinações, essenciais para a interação no ambiente virtual.
- Trimpot de 10k Ω (porta D3): Permite ajustar a sensibilidade dos movimentos ou a velocidade de resposta do sistema, proporcionando uma experiência personalizada para o usuário.
- Motores de vibração (portas D4 e D5): Oferecem feedback háptico ao usuário, simulando sensações físicas durante a interação com objetos virtuais. Isso aumenta a imersão e realismo da simulação, especialmente em cenários cirúrgicos.
- Push buttons (portas D7 a D10): Atuam como interface de controle manual, permitindo ao usuário executar comandos específicos, como seleções ou ajustes no ambiente virtual.
- A bateria Li-Po fornece energia para todo o sistema, conectando-se ao microcontrolador para garantir uma operação autônoma e contínua, com um circuito de carregamento integrado ao módulo Seeed XIAO BLE Sense.

O sistema foi projetado para integrar dados de movimento e comandos de entrada ao ambiente de realidade virtual executado no Meta Quest 2, permitindo comunicação em tempo real com baixa latência por meio do protocolo BLE. A comunicação via *Bluetooth Low Energy* garante a transmissão de dados com baixa latência, essencial para que as interações no ambiente virtual sejam replicadas de forma precisa e em tempo real.

O feedback háptico, proporcionado pelos motores de vibração, é sincronizado com as ações realizadas no ambiente virtual, simulando, por exemplo, o toque de instrumentos cirúrgicos em tecidos, aumentando a fidelidade da simulação. Essa integração entre sensores, atuadores e o ambiente de VR melhora significativamente a imersão, e a personalização dos parâmetros de controle, como a sensibilidade ajustada pelo trimpot, garante uma experiência flexível tanto para iniciantes quanto para usuários avançados.

4.1.1. Circuito eletrônico do Bisturi

O circuito representado na Figura 20 apresenta uma placa de ensaio (*proto-board*), com o módulo de controle XIAO Seeed nRF52840 Sense (no centro-esquerdo do circuito) que é responsável por receber entradas e controlar saídas. Ele é alimentado por uma bateria Li-Po de 3,7V com capacidade de 110mAh (à esquerda), que fornece energia ao sistema. Ele se comunica com os sensores e os atuadores conectados por meio de suas portas GPIO e possui um acelerômetro e giroscópio digital (LSM6DS3TR-C) internamente, que foi utilizado.

O acelerômetro ADXL335 (localizado abaixo do microcontrolador) mede a aceleração nos eixos X, Y, e Z. Os dados deste sensor são enviados ao microcontrolador para que sejam processados e utilizados em aplicações como controle de movimento ou detecção de inclinação.

A comunicação entre o acelerômetro e o microcontrolador ocorre por meio de conexões analógicas (saídas X, Y e Z) e o pino VCC é conectado à alimentação de 3,3V, enquanto o GND está conectado ao terra.

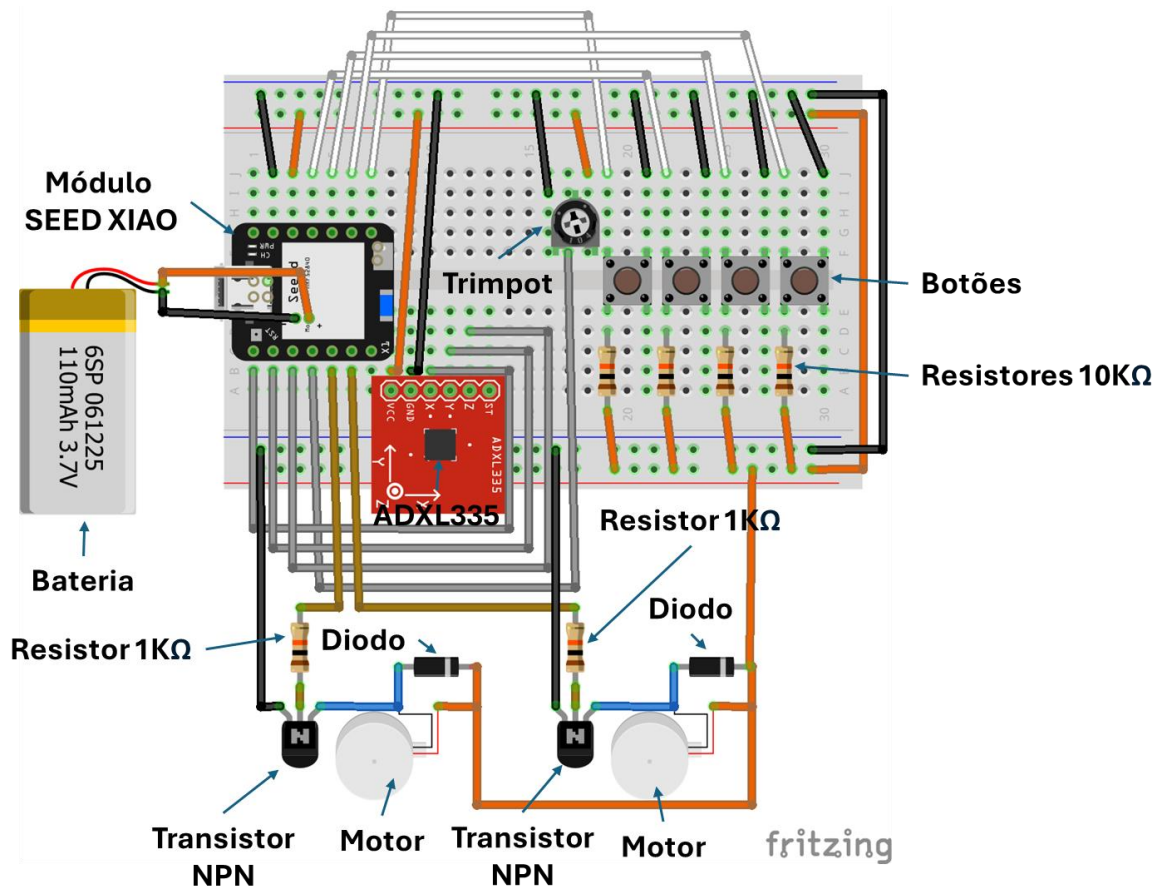
Há um conjunto de 4 botões dispostos na parte superior direita do circuito, que são utilizados para entrada de controle manual. Cada botão tem um resistor pull-down conectado ao terra para evitar leituras indesejadas quando não pressionados.

Na parte inferior do circuito, dois motores *Vibracall* são conectados e controlados pelo sistema. Eles são acionados por meio de transistores, que atuam como interruptores eletrônicos. O controle dos motores é realizado pela ativação/desativação dos transistores com base nos sinais do microcontrolador.

Cada motor tem diodos e transistores NPN conectados para controlar a direção da corrente e, conseqüentemente, o funcionamento do motor. Os diodos protegem o circuito de picos de tensão gerados quando o motor é desligado.

O circuito é alimentado por uma bateria de 3,7V Li-Po (mostrada à esquerda), conectada ao microcontrolador. Essa bateria fornece energia tanto para os sensores quanto para os atuadores. O regulador de tensão interno do microcontrolador ajusta a tensão para níveis adequados para os componentes.

Figura 20 — Montagem circuito (Fritzing)



Fonte: Autor, 2024

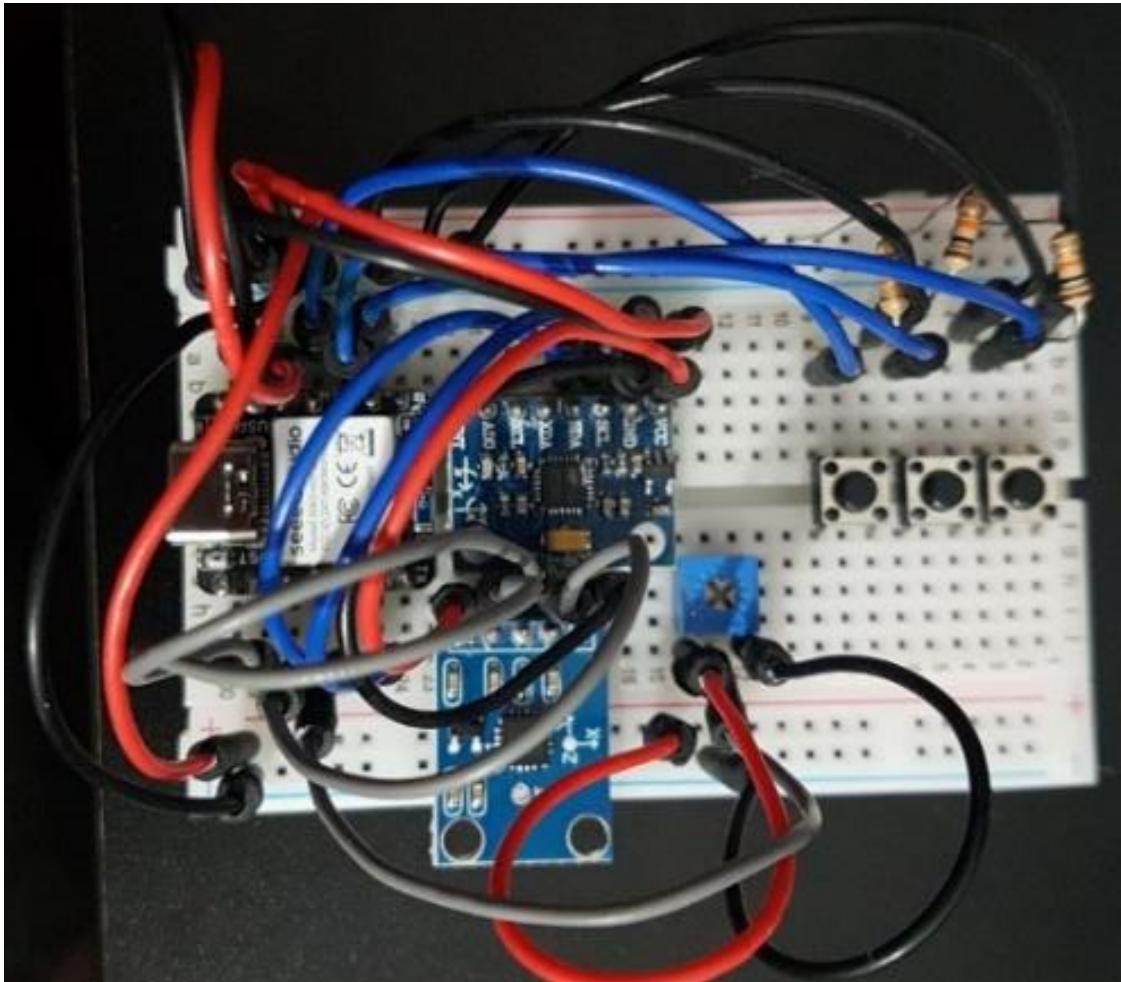
4.2. PROJETO E IMPLEMENTAÇÃO DO BISTURI

Para o desenvolvimento do esquemático eletrônico e a placa de circuito impresso, foi utilizado o software KiCad, por ser gratuito (*open source*) e possuir diversos recursos.

4.2.1. Montagem do circuito

A Figura 21 representa a montagem de prova de conceito do circuito para o desenvolvimento do bisturi. Para garantir a funcionalidade dos componentes antes da produção da placa de circuito impresso, foi utilizada uma protoboard para conectar os elementos do sistema, incluindo o Seead XIAO BLE Sense nRF52840, sensores e atuadores.

Figura 21 — Montagem física do circuito



Fonte: Elaboração própria, 2024.

Nesta montagem, foram utilizados os seguintes componentes:

- Seeed XIAO BLE Sense nRF52840: Atua como o microcontrolador central, responsável pela coleta de dados dos sensores e controle dos atuadores.
- Acelerômetro: Responsável pela detecção de movimentos nos eixos X, Y e Z, essencial para a simulação de movimento no ambiente virtual.
- Trimpot: Utilizado para ajustar a sensibilidade do sistema, permitindo alterar a velocidade do cursor.
- Botões de pressão (*push buttons*): Funcionam como a interface de controle manual, permitindo que o usuário execute comandos no sistema de realidade virtual.

- Resistores: Conectados aos botões como *pull-downs*, garantindo a estabilidade dos sinais quando os botões não estão pressionados.
- Fios jumper: Utilizados para conectar todos os componentes de maneira flexível na protoboard.

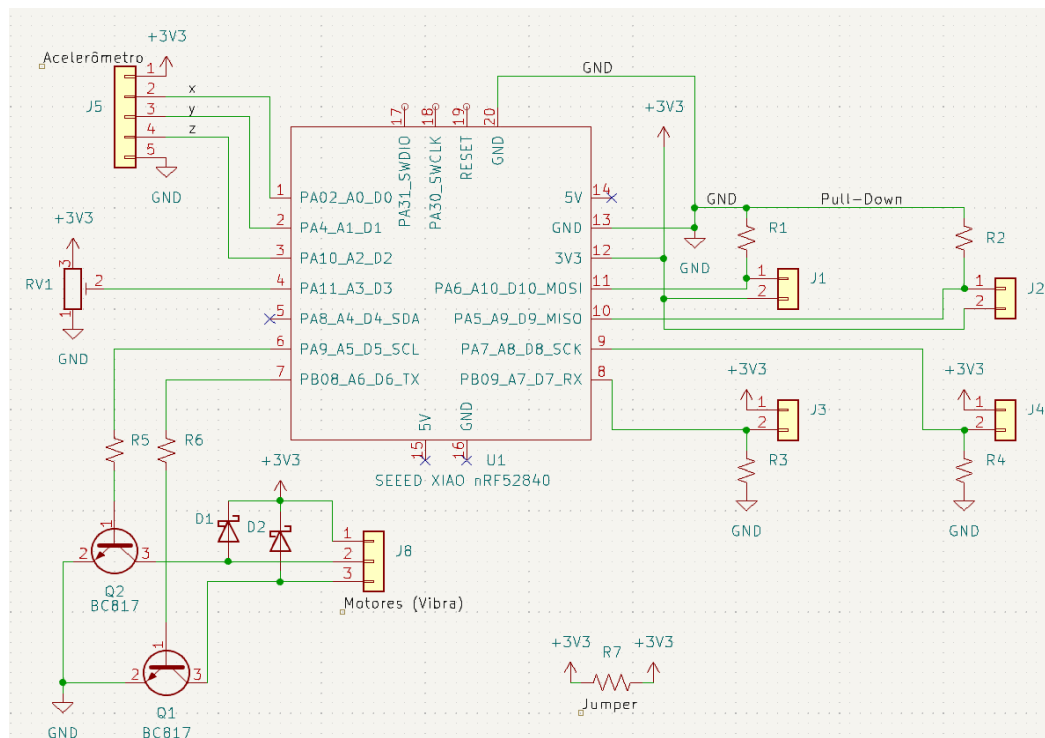
O uso da protoboard facilitou a interligação de todos os componentes sem a necessidade de solda, permitindo ajustes rápidos e a verificação das funcionalidades do sistema em um ambiente de teste.

Essa montagem de prova de conceito foi essencial para validar as conexões e garantir que todos os componentes estivessem funcionando conforme o esperado antes de avançar para a fabricação da placa de circuito impresso (PCI).

4.2.2. Diagrama esquemático do circuito

O diagrama esquemático apresentado na Figura 22 ilustra o circuito responsável pela comunicação entre o módulo Seeed XIAO BLE Sense nRF52840, sensores, atuadores e outros componentes no sistema.

Figura 22 — Esquemático circuito



Fonte: Elaboração própria, 2024.

A seguir, são descritos os principais elementos e conexões do circuito.

- O módulo Seeed XIAO BLE Sense nRF52840 (U1) é o núcleo do sistema. Ele recebe dados de entrada dos sensores e controla os atuadores (motores de vibração). O módulo é alimentado com 3,3V e possui várias conexões digitais para controlar os diferentes componentes conectados.
- O sensor de aceleração ADXL335 (conectado no conector J5) mede a aceleração nos eixos X, Y, e Z. Esses dados são enviados ao microcontrolador através das portas PA02 (A0), PA04 (A1) e PA10 (A2), que estão ligadas aos pinos de saída do acelerômetro correspondentes a cada eixo.
- Um trimpot de 10k Ω (RV1) está conectado ao pino PA11 (A3). Ele permite que o usuário ajuste a sensibilidade do sistema, responsável pela velocidade do cursor durante a interação com o ambiente virtual. Esse ajuste é lido como um valor analógico pelo microcontrolador.
- Quatro botões de pressão (*push buttons*) conectados aos conectores J1, J2, J3, J4 são utilizados para a interface de controle manual do sistema. Esses botões são conectados aos pinos digitais do módulo (PA06 a PA09), com resistores *pull-down* (R1 a R4) para garantir que os sinais dos botões sejam estáveis, evitando leituras incorretas quando não pressionados.
- O sistema inclui dois motores de vibração conectados ao conector J8, que proporcionam feedback tátil ao usuário. Os motores são controlados por dois transistores NPN (Q1 e Q2) do tipo BC817, que atuam como interruptores para ativar ou desativar os motores com base nos sinais recebidos do microcontrolador nos pinos PA08 (D4) e PA09 (D5). Diodos 1N4148 (D1 e D2) são usados para proteger o circuito contra a tensão reversa gerada pelos motores.
- O circuito é alimentado por uma bateria de 3,7V a 4,2V e possui reguladores de tensão, de modo a fixar o valor em 3,3V e conectado aos componentes periféricos. A alimentação é distribuída uniformemente pelo circuito para garantir o funcionamento correto de todos os componentes.

- O jumper R7 é utilizado para conectar dois pontos de alimentação, permitindo uma maior flexibilidade no projeto, caso haja a necessidade de alterar a configuração da fonte de alimentação no futuro.

Este circuito integra o acelerômetro, que detecta movimentos em três eixos, e os motores de vibração, que fornecem feedback tátil. Os botões e o trimpot fornecem controle manual e ajustes de sensibilidade ao sistema. O módulo Seeed XIAO BLE Sense nRF52840 coordena todas essas entradas e saídas e transmite dados para o Meta Quest 2 via *Bluetooth Low Energy* (BLE), permitindo a interação do usuário com o ambiente de realidade virtual em tempo real.

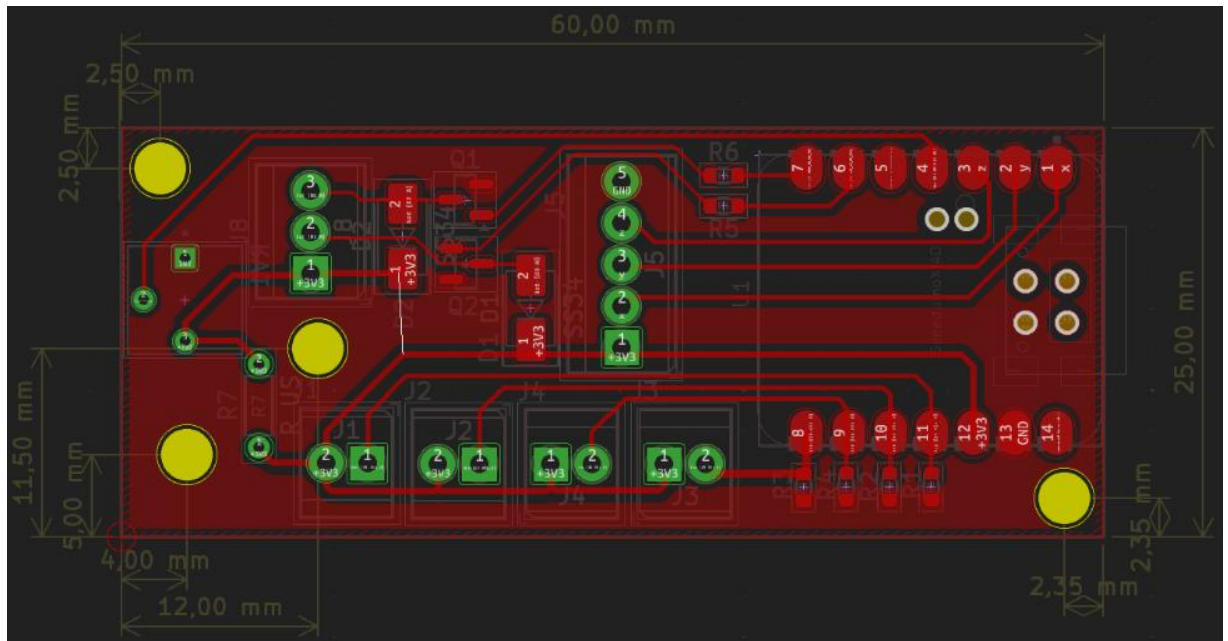
4.2.3. Design do Hardware

O design do hardware para o projeto, inclui o desenvolvimento e a fabricação de uma PCI - placa de circuito impresso. A PCI foi projetada para integrar o microcontrolador Seeed XIAO BLE Sense nRF52840 com os sensores, atuadores e outros componentes do sistema, proporcionando uma plataforma compacta e eficiente para comunicação e controle em um ambiente de simulação médica de realidade virtual.

A Figura 23 apresenta o layout da PCI, sendo ela apenas de face simples, onde estão definidos os caminhos para a alimentação, sinais de controle e conexões de dados entre os diversos componentes do sistema.

A placa possui dimensões aproximadas de 60 mm x 25 mm, com trilhas bem definidas para evitar interferências e garantir a integridade dos sinais. A organização dos componentes é feita de forma a otimizar o espaço, mantendo uma boa separação entre os circuitos de alimentação e controle. Os componentes foram distribuídos estrategicamente na placa para minimizar interferências eletromagnéticas e garantir o desempenho ideal do módulo SEEED XIAO BLE Sense. Em especial, a antena BLE foi posicionada de forma a evitar proximidade excessiva com trilhas de sinal e componentes eletrônicos que possam gerar ruídos ou degradação do sinal de rádio.

Figura 23 — Placa de circuito impresso.

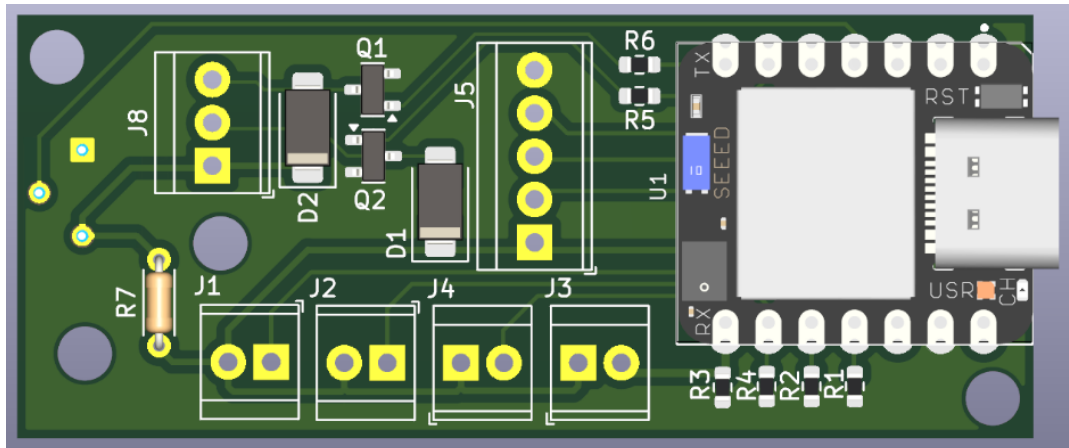


Fonte: Elaboração própria, 2024.

A Figura 24, apresenta a disposição dos principais componentes na PCB.

- O conector J8 está alocado para a conexão dos motores de vibração, que fornecem feedback tátil ao usuário.
- Os conectores J1 a J4 são utilizados para a conexão de botões de controle, que permitem ao usuário interagir diretamente com o sistema.
- O acelerômetro é conectado no conector J5, monitorando os movimentos nos eixos X, Y e Z.
- Diodos D1 e D2 são utilizados para proteger o circuito contra picos de tensão gerados pelos motores de vibração, enquanto transistores BC817 (Q1 e Q2) controlam a ativação dos motores.
- O módulo Seeed XIAO BLE Sense nRF52840 (U1) é o componente central da PCB, recebendo os sinais dos sensores e controlando os atuadores.

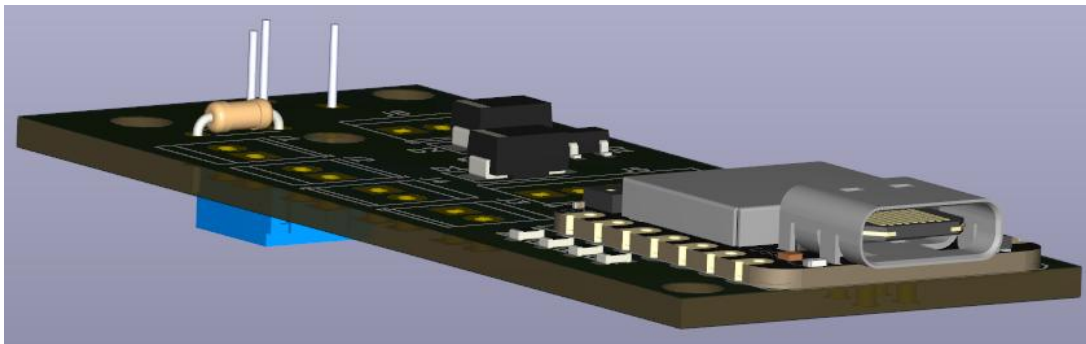
Figura 24 — Modelo 3D Placa de circuito impresso



Fonte: Elaboração própria, 2024.

A Figura 25, oferece uma visualização 3D da PCB montada, destacando a disposição dos componentes na superfície da placa e o conector USB do módulo Sseed XIAO BLE Sense. Este design compacto e otimizado permite que a placa seja facilmente instalada em dispositivos portáteis ou vestíveis, como um bisturi para simulação em realidade virtual.

Figura 25 — Modelo 3D Placa de circuito impresso



Fonte: Elaboração própria, 2024.

O design do hardware foi cuidadosamente desenvolvido para maximizar a eficiência, minimizando o tamanho da PCB sem comprometer a funcionalidade. O uso de componentes compactos, como transistores e diodos de montagem em superfície, além do microcontrolador integrado com BLE, garante que o sistema seja capaz de se comunicar com o Meta Quest 2 via *Bluetooth Low Energy*, proporcionando uma experiência de simulação responsiva e imersiva.

4.3. FIRMWARE E TESTES

Após a montagem física do circuito, foi necessário o desenvolvimento de um firmware para embarcar no módulo Seeed XIAO nRF52840, de modo que seja possível coletar os dados dos sensores, ler os valores dos botões e do trimpot, acionar os motores e transmitir os dados via BLE (*Bluetooth Low Energy*) o código está disponível em Apêndice A – Firmware desenvolvido.

O ambiente de desenvolvimento escolhido para desenvolver este código foi a IDE do arduino.

Inicialmente foi realizada a leitura dos dados do acelerômetro (ADXL335) verificando a transmissão dos valores de aceleração nos eixos X, Y e Z para o microcontrolador.

Foi realizado teste no controle dos atuadores (motores de vibração), verificando a correta ativação dos motores de vibração por meio de comandos processados pelo microcontrolador.

Os botões de controle foram testados, garantindo que os comandos manuais pudessem ser enviados e processados corretamente.

Essa fase também possibilitou a realização de testes preliminares de comunicação via *Bluetooth Low Energy* (BLE) com o Meta Quest 2, verificando a transmissão dos dados coletados pelos sensores e a ativação dos motores de vibração como resposta ao feedback tátil.

Com os primeiros testes realizados, foi desenvolvido todo o firmware, de modo a integrar todos os componentes do hardware, o resultado pode ser visualizado a Figura 26 apresenta a inclusão de algumas bibliotecas e criação de macros e variáveis que servirão para o programa principal.

Figura 26 — Estrutura main.h

```

nRF52840_BLE_GATT_v2.ino  main.h
1  #include <bluefruit.h>
2  #include <Adafruit_TinyUSB.h>
3  #include <Adafruit_Sensor.h>
4  #include <Wire.h>
5  #include <LSM6DS3.h>
6
7  #define X 0
8  #define Y 1
9  #define Z 2
10 #define VEL_PIN 3
11 #define VIBRATION_1 5
12 #define VIBRATION_2 6
13 #define FOURTH_BUTTON 7
14 #define THIRD_BUTTON 8
15 #define SECOND_BUTTON 9
16 #define FIRST_BUTTON 10
17 #define VIN 3.3
18 #define MID_VIN 1.65
19 #define ZERO 0
20 #define ONE 1
21 #define DECIMAL 2
22 #define INPS_ARRAY 8
23 #define MIN_VALUE 15
24 #define MIN_ACC_THRESHOLD 1.5
25 #define THRESHOLD 5
26 #define MAX_HID 127
27 #define MIN_HID -127
28 #define RESOLUTION 1024.0
29 #define TIME_SLEEP 120000UL
30
31 unsigned long lastMillis = 0, motorStartTime = 0;
32 int motorDuration = 0, pwmValue = 0, velocity = 1;
33 bool motorActive = true;
34 float previousAccX = 0, previousAccY = 0, previousAccZ = 0;
35 float previousGyroX = 0, previousGyroY = 0, previousGyroZ = 0;
36 float filterAlpha = 0.1;
37 float biasGyroX = 0, biasGyroY = 0, biasGyroZ = 0;
38
39 const uint8_t pinInps[INPS_ARRAY] = {X, Y, Z, VEL_PIN, FOURTH_BUTTON, THIRD_BUTTON, SECOND_BUTTON, FIRST_BUTTON};
40 float accX, accY, accZ, gyroX, gyroY, gyroZ;
41 const int RawMin = 0, RawMax = 1023, numSamples = 200;
42 const float biasAccX = 0.51, biasAccY = -0.37, biasAccZ = -0.53, Volts = 0.33;
43 uint8_t buttons = 0;
44
45 void initPeripherals();
46 float updateAxisData(int axisPin, float bias);
47 uint8_t checkButtons();
48 void calibrateGyroscope();
49 void int1ISR();
50 void onMotorControlWrite(uint16_t conn_hdl, BLECharacteristic* chr, uint8_t* data, uint16_t len);

```

Fonte: Elaboração própria, 2024.

A Figura 27 apresenta a estrutura principal do firmware desenvolvido, utilizando todos os recursos da estrutura “main.h”.

Figura 27 — Estrutura principal do código

```

nRF52840_BLE_GATT_v2.ino  main.h
1  #include "main.h"
2
3  BLEDis bledis;
4  BLEService customService = BLEService(0x1234); // UUID personalizado para seu serviço
5  BLECharacteristic allDataChar = BLECharacteristic(0xFFFF);
6  BLECharacteristic motorControlChar = BLECharacteristic(0x2C03);
7  LSM6DS3 internal_IMU(I2C_MODE, 0x6A);
8
9  > void setup() { ...
36 }
37
38 > void loop() { ...
114 }
115
116 > void initPeripherals() { ...
155 }
156
157 > float updateAxisData(int axisPin, float bias) { ...
162 }
163
164 > uint8_t checkButtons() { ...
173 }
174
175 > void calibrateGyroscope() { ...
187 }
188
189 > void int1ISR() { ...
194 }
195
196 > void onMotorControlWrite(uint16_t conn_hdl, BLECharacteristic* chr, uint8_t* data, uint16_t len) { ...
221 }

```

Fonte: Elaboração própria, 2024.

O código faz uso de diversas bibliotecas cruciais para a operação do dispositivo, como a “bluefruit.h” para controle de comunicação BLE, “Adafruit_TinyUSB.h” para suporte USB, e bibliotecas de sensores como “Adafruit_MPU6050.h” e “LSM6DS3.h” para gerenciamento dos dados do giroscópio e acelerômetro. Essas bibliotecas permitem a interação com sensores de movimento e controle do dispositivo via BLE, facilitando a integração de hardware e software no contexto de realidade virtual (RV) e feedback tátil.

As classes BLEDis bledis, BLEHidGamepad blegamepad, e BLEUart bleuart configuram o dispositivo BLE para fornecer informações de dispositivos HID (*Human Interface Device*) e comunicar-se via UART BLE. A variável internal_IMU é utilizada para acessar os dados do sensor IMU (Unidade de Medida Inercial), fornecendo dados de aceleração e rotação essenciais para capturar os movimentos do bisturi.

No setup(), o dispositivo é inicializado com BLE e é ativado. A taxa de transmissão BLE é configurada para 4dBm (para melhorar o alcance) e o nome do dispositivo BLE é configurado como “Bisturi”, facilitando sua identificação ao ser emparelhado com um dispositivo externo.

A biblioteca do sensor IMU (LSM6DS3) é inicializada para medir os valores do giroscópio, valores essenciais para monitorar o movimento do bisturi virtual. A função `analogReference(AR_INTERNAL_3_0)` define o referencial analógico em 3.0V para as leituras do ADC, ajustando a precisão e evitando ruídos excessivos no circuito.

A função `startAdv()` é responsável por iniciar a publicidade BLE. Ela configura o dispositivo para ser visível para conexões BLE e permite o envio de dados HID (como controle de *gamepads*) e UART (para comunicação geral), crucial para comunicar o estado e os comandos recebidos do bisturi.

O loop principal é onde ocorrem as leituras contínuas dos sensores e o envio dos dados via BLE:

Primeiramente, é verificado se o dispositivo está conectado via BLE, essencial para garantir a comunicação contínua.

O acelerômetro e giroscópio são lidos periodicamente usando a função `ReadAxis()`. Aqui, o valor do acelerômetro é ajustado em relação à sensibilidade configurada e a velocidade de movimento é ajustada com base no trimpot. Os dados do giroscópio são diretamente adquiridos do sensor LSM6DS3.

Um filtro passa-baixa é aplicado aos dados de aceleração e giroscópio para atenuar ruídos indesejados, como a vibração do motor háptico. Essa técnica melhora a precisão dos dados dos sensores durante a operação.

Após as leituras dos sensores e o processamento dos dados, os valores de aceleração e rotação são enviados via `blegamepad.report()`, transformando o dispositivo em um *gamepad* BLE, onde os valores de movimento são transmitidos em tempo real para o sistema de RV.

Se comandos BLE forem recebidos, eles são processados pela função `parseCommand()`, permitindo o controle dinâmico do motor de vibração.

A função `CheckButtons()` lê o estado dos botões físicos conectados ao bisturi. Ela usa leitura digital para verificar se algum dos botões foi pressionado e retorna um valor específico, dependendo do botão pressionado, configurando diferentes ações (como botões de controle de um *gamepad*).

A função `parseCommand()` é responsável por interpretar os comandos enviados via BLE para controlar o tempo e a intensidade do motor de vibração. O comando recebido especifica a duração (em milissegundos) e o valor PWM (ciclo de trabalho) que será aplicado ao motor, permitindo diferentes intensidades de vibração com base no feedback tátil desejado.

A função `feedback()` é um exemplo de como o motor de vibração pode ser acionado ou desligado com base nos comandos recebidos. Essa função é fundamental para fornecer feedback tátil ao usuário durante o uso do bisturi virtual em um ambiente de RV.

As interrupções são configuradas com `nrf_gpio_cfg_sense_input()`, que coloca o microcontrolador no estado "sensível", ou seja, ele ficará monitorando os pinos de entrada e acordará quando um estado de subida (RISING) for detectado em qualquer botão.

Esta abordagem de gerenciamento de energia otimiza o consumo do microcontrolador nRF52840 durante a inatividade, um aspecto essencial para dispositivos portáteis e sistemas hápticos. A aplicação do modo de baixa potência reduz significativamente o consumo de energia, garantindo maior autonomia da bateria, crucial para dispositivos vestíveis e operados em RV. Além disso, a combinação de leitura contínua dos sensores de movimento (acelerômetro e giroscópio) com a detecção de interrupções proporciona um controle robusto, permitindo que o dispositivo reaja imediatamente quando um botão for pressionado ou um movimento significativo for detectado, sem comprometer a experiência do usuário.

O firmware aqui apresentado integra de forma eficiente o controle de hardware (acelerômetros, giroscópios, motores de vibração e botões) com a comunicação BLE, possibilitando a criação de uma interface háptica para simulações de cirurgias em ambientes de realidade virtual.

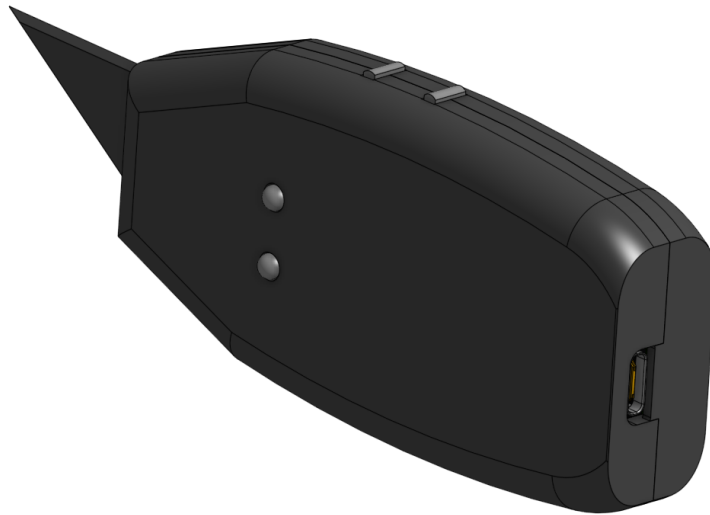
A utilização de um filtro passa-baixa para compensar o impacto das vibrações, juntamente com o envio de dados em tempo real via BLE, faz deste firmware uma solução viável para simulações interativas e imersivas. A modularidade e a escalabilidade do código permitem sua expansão para outros dispositivos e funções, tornando-o adequado para aplicações futuras em interfaces homem-máquina em ambientes médicos simulados.

4.4. CRIAÇÃO DO DESIGN MECÂNICO

A criação do design mecânico do bisturi foi uma etapa fundamental para garantir a acomodação dos componentes eletrônicos e funcionalidade do dispositivo no ambiente de realidade virtual.

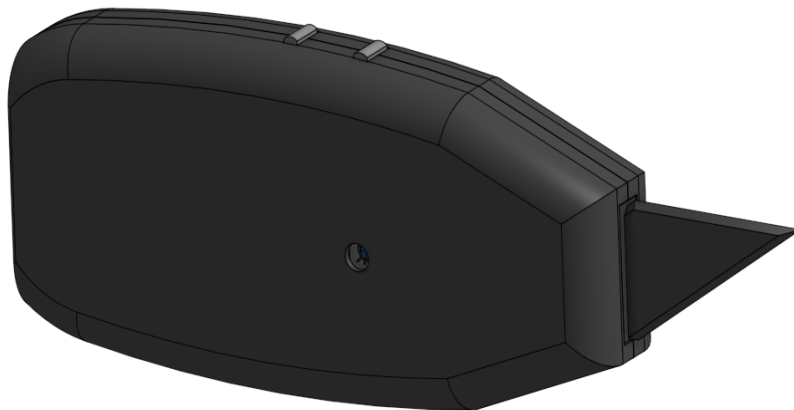
O projeto foi desenvolvido utilizando o software web OnShape de modelagem 3D, com o objetivo de otimizar a forma, peso e distribuição interna dos componentes eletrônicos, como mostrado nas Figuras 28 e 29.

Figura 28 — Perspectiva frontal bisturi



Fonte: Elaboração própria, 2024.

Figura 29 — Perspectiva traseira bisturi



Fonte: Elaboração própria, 2024.

O design do bisturi levou em consideração a necessidade de garantir o conforto do usuário durante o manuseio prolongado, bem como a durabilidade dos materiais utilizados.

O invólucro foi projetado para ser impresso em uma impressora 3D, utilizando materiais leves como PLA (ácido polilático), que são conhecidos por sua resistência e baixo custo.

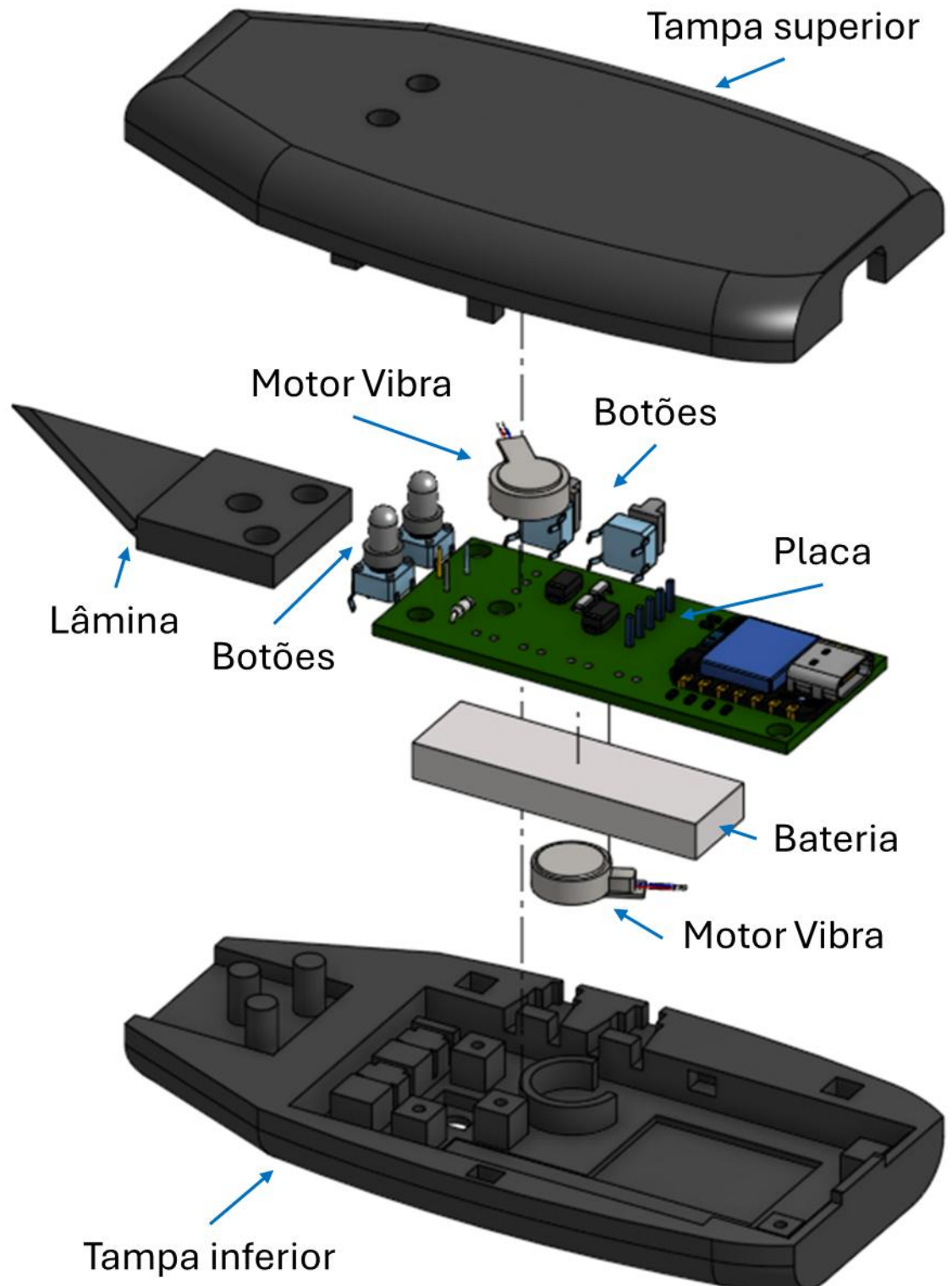
A forma do bisturi foi projetada para permitir uma pegada firme e confortável, com botões estrategicamente posicionados para facilitar o acesso sem que o usuário precise desviar sua atenção do ambiente virtual. A inclusão de dois motores de vibração internamente também foi planejada de forma a garantir a melhor distribuição de peso e proporcionar feedback tátil ao usuário.

Além disso, o design considerou os aspectos de dissipação de calor e ventilação dos componentes internos, uma vez que o microcontrolador, os motores e a bateria podem gerar calor durante o uso. O suporte para a bateria foi ajustado de forma a facilitar sua remoção e substituição, garantindo a manutenção do bisturi de forma prática.

Após a conclusão do design, o bisturi foi impresso em 3D e montado com todos os componentes internos, como a placa com seus componentes, motores de vibração, e os botões.

A Figura 30 ilustra o design final do bisturi, com uma vista explodida que mostra a distribuição dos componentes internos e como eles se encaixam na estrutura externa.

Figura 30 — Vista explodida bisturi e componentes



Fonte: Elaboração própria, 2024.

4.5. CONFIGURAÇÃO DO AMBIENTE VIRTUAL E INTEGRAÇÃO COM META QUEST 2

Para a o desenvolvimento da sala de cirurgia no Unity, o foco será o desenvolvimento de um ambiente interativo onde o usuário possa realizar operações simuladas. A integração com o Meta Quest 2 será feita utilizando o *XR Interaction Toolkit* para garantir a interação com objetos virtuais, como tecidos e instrumentos cirúrgicos, por meio do bisturi e dos controladores do Quest 2.

Em *Package Manager* (*Window -> Package Manager*) foi instalado 3 pacotes para facilitar o desenvolvimento em ambiente de realidade, esses pacotes são:

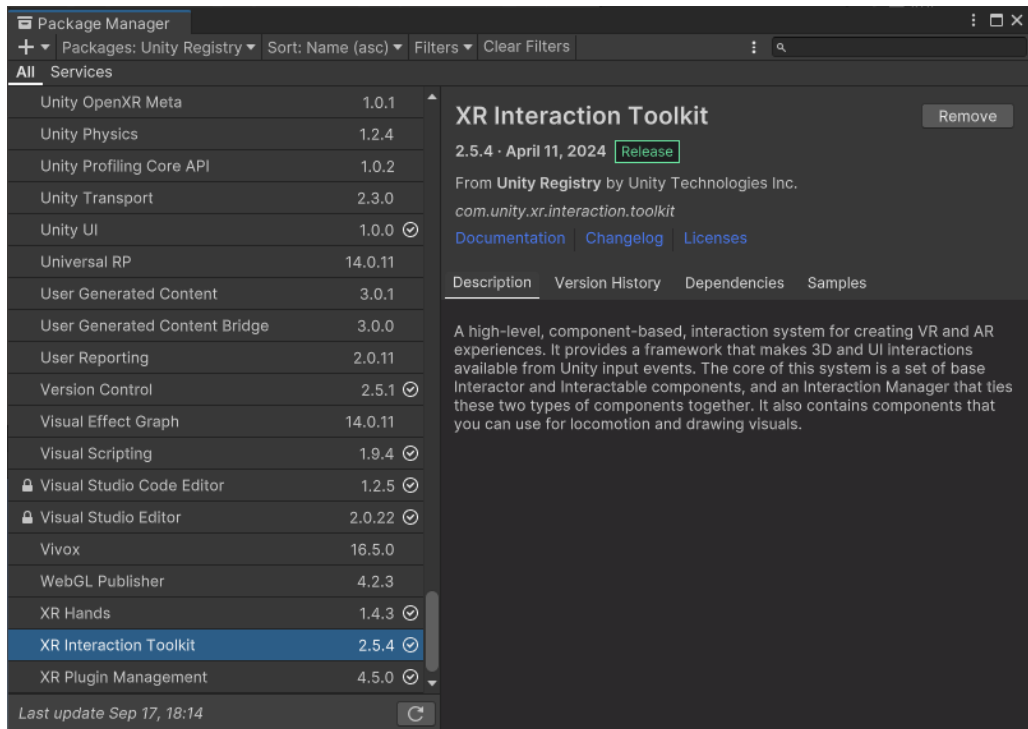
XR Hands: Permite o uso de rastreamento de mãos em experiências de realidade estendida (XR). Ele é usado para capturar os movimentos e gestos das mãos do usuário, possibilitando interações sem o uso de controladores físicos, como o Meta Quest 2. Esse sistema é fundamental para criar interações naturais e imersivas em VR.

XR Interaction Toolkit: Facilita a criação de interações em realidade virtual e aumentada. Ele fornece componentes pré-construídos para interações comuns em VR/AR, como pegar objetos, pressionar botões virtuais e mover-se pelo ambiente. Esse *toolkit* oferece suporte para controladores, mãos rastreadas e outras interfaces, reduzindo a necessidade de escrever código específico para cada interação.

XR Plugin Management: É uma ferramenta de gerenciamento de plugins do Unity que facilita a configuração da plataforma XR. Ele permite escolher e gerenciar as plataformas XR que o projeto deve suportar, assegurando que o Unity compile corretamente para cada uma delas. Essa ferramenta simplifica a integração de plugins de realidade estendida.

Na Figura 31 podem ser visto o gerenciador de pacotes.

Figura 31 — Gerenciador de pacotes



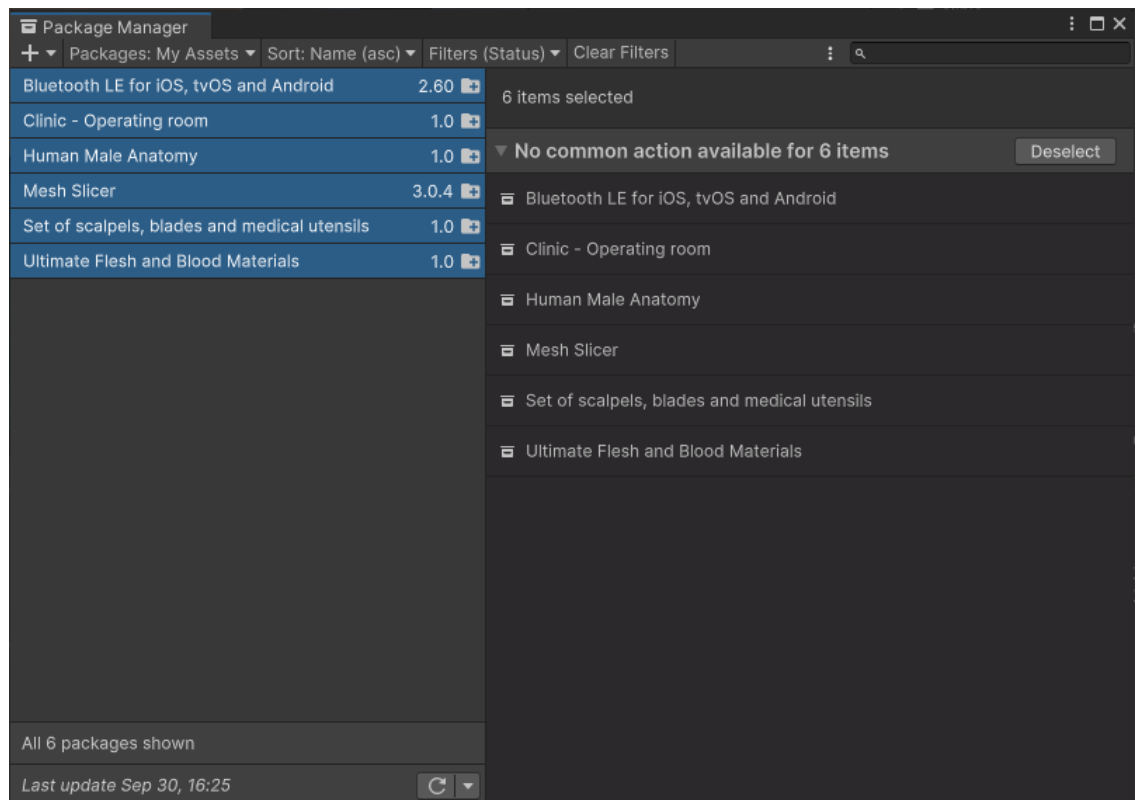
Fonte: Elaboração própria, 2024.

Foram adquiridos 6 Assets na *assets.store* da Unity, esses Assets foram:

- *Bluetooth LE for iOS, tvOS and Android*: Assets para comunicação Bluetooth.
- *Clinic – Operation room*: Assets da sala de cirurgia pronta.
- *Human Male Anatomy*: Assets do corpo humano masculino.
- *Mesh Slicer*: Assets para cortar e deformar.
- *Set of scalpels, blades and medical utensils*: Assets de bisturis prontos.
- *Ultimate Flesh and Blood Materials*: Assets para efeito de sangue.

A Figura 32 apresenta os assets instalados:

Figura 32 — Assets instalados



Fonte: Elaboração própria, 2024.

4.5.1. Design da Interface de Usuário

O design da interface foi pensado para ser intuitivo e fácil de usar, para que o usuário possa interagir facilmente com o bisturi virtual e o ambiente de cirurgia. O design foi feito para garantir que o usuário permaneça imerso na experiência de realidade virtual. O uso de feedback tátil para simular o toque do bisturi, bem como a interação direta com os objetos virtuais (como o paciente), são importantes para reforçar a sensação de imersão.

Os elementos da interface (como o bisturi, o modelo do paciente, e as ferramentas cirúrgicas) foram distribuídos na tela para garantir que o usuário possa focar na tarefa cirúrgica sem distrações.

O feedback de vibração quando o bisturi corta camadas de tecido, por exemplo, melhora a experiência do usuário e proporciona uma simulação mais realista de uma cirurgia.

O design da interface levou em consideração a ergonomia. Ferramentas e opções de menus são posicionadas de forma que o usuário possa acessá-las facilmente sem interromper o fluxo da cirurgia.

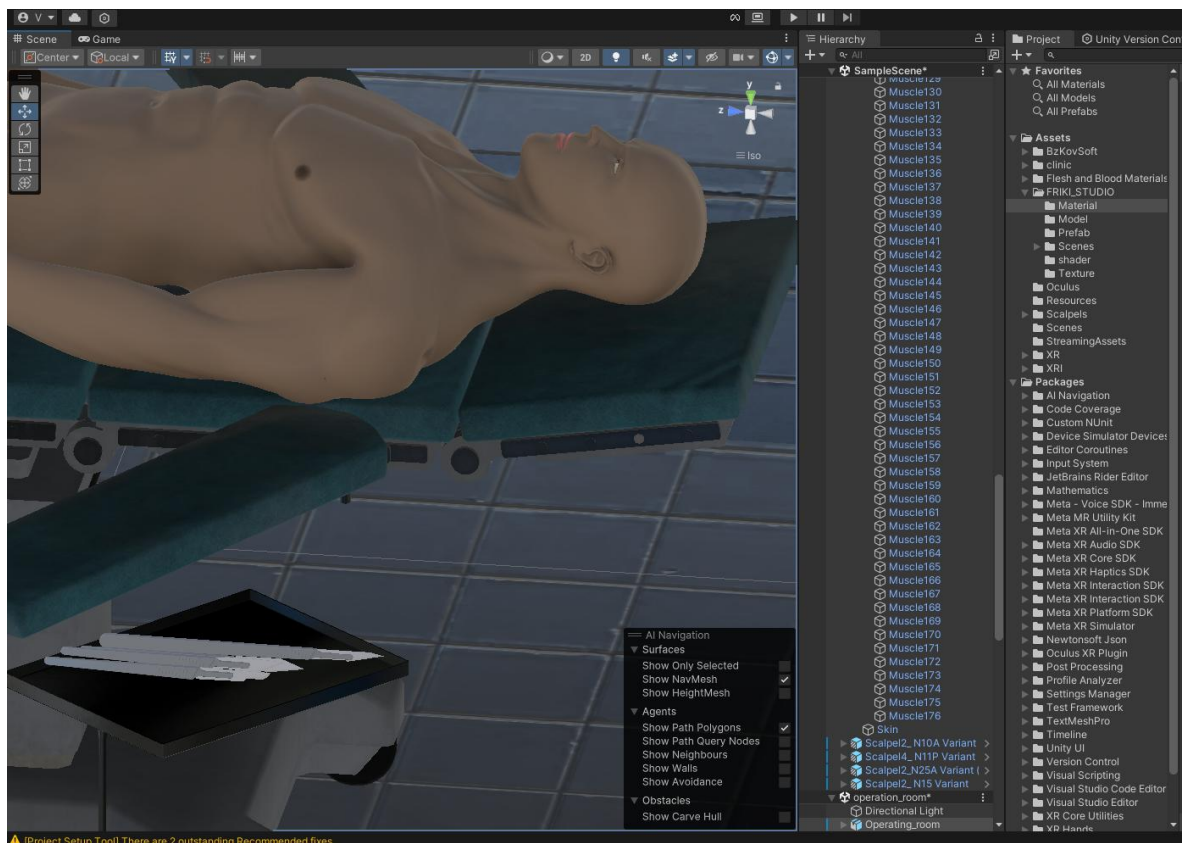
O design da interface foi pensado para se adaptar a diferentes dispositivos, como o Meta Quest 2.

4.6. DESENVOLVIMENTO DA SALA DE REALIDADE VIRTUAL EM UNITY

Usando o assets *Clinic – Operation room* como base para a sala de cirurgia, foi adicionado o *Human Male Anatomy* no ambiente da sala de cirurgia, de modo que este modelo humano virtual anatômico fique posicionado na maca da sala de cirurgia, algumas de suas texturas foram alteradas, incluindo textura de gordura com o assets *Ultimate Flesh and Blood Materials*.

A Figura 33 apresenta o resultado do modelo humano virtual posicionado na maca da sala de cirurgia.

Figura 33 — Sala de cirurgia com modelo humano virtual anatômico



Fonte: Elaboração própria, 2024.

Na cena foi aplicada uma luz direcional utilizando o objeto “Directional Light” e inserido o bisturi como assets: *Set of scalpels, blades and medical utensils*. A Figura 34 apresenta o resultado final do bisturi posicionado na cena, junto ao modelo humano virtual anatômico, a posição padrão da camera e a luz aplicada ao ambiente.

Figura 34 — Sala de cirurgia com modelo humano virtual anatômico



Fonte: Elaboração própria, 2024.

4.6.1. Scripts Bluetooth Low Energy

O assets *Bluetooth LE for iOS, tvOS and Android*, possui diversos plugins para comunicação *Bluetooth Low Energy*, com o auxílio destes plugins, foram desenvolvidos 3 scripts para conexão com o bisturi e envio de mensagens.

A Figura 35 apresenta o script “ScannerBLE.cs”, responsável por escanear os dispositivos BLE e estabelecer a conexão com o qual possua o prefixo “Bisturi-” em seu nome. Além disso, o script realiza a inscrição na característica com UUID: “0000ffff-0000-1000-8000-00805f9b34fb” para receber as notificações de dados provenientes do acelerômetro, giroscópio, sensor de temperatura e botões. Essas informações são transmitidas em pacotes de 16 bytes, processadas pelo script e disponibilizadas para o ambiente de simulação.

Figura 35 — Script ScannerBLE

```

1  using System;
2  using System.Net;
3  using System.Text;
4  using UnityEngine;
5  // Script do Unity (1 referência de ativo) | 3 referências
6  public class BLEAutoConnect : MonoBehaviour
7  {
8      private string targetDeviceName = "Bisturi-";
9      private string targetServiceUUID = "00001234-0000-1000-8000-00805f9b34fb";
10     private string combinedDataCharUUID = "0000ffff-0000-1000-8000-00805f9b34fb";
11     private string motorControlChar = "00002c03-0000-1000-8000-00805f9b34fb";
12     private string connectedAddress;
13     public GameObject scalpel;
14     public float currentTemperature { get; private set; }
15     public int currentButtons { get; private set; }
16     public Vector3 currentAccelData { get; private set; }
17     public Vector3 currentGyroData { get; private set; }
18
19     // Mensagem do Unity | 0 referências
20     void Start() {
21         BluetoothLEHardwareInterface.Initialize(true, false, () => {
22             BluetoothLEHardwareInterface.ScanForPeripheralsWithServices(null, (address, name) => { }, null);
23         }, (error) => { });
24     }
25
26     // 1 referência
27     void ConnectToDevice(string address) {
28         BluetoothLEHardwareInterface.ConnectToPeripheral(address, (connectedAddress) => { }, null, (connectedAddress, serviceUUID, characteristicUUID) => {
29             if (serviceUUID == targetServiceUUID) {
30                 BluetoothLEHardwareInterface.SubscribeCharacteristic(connectedAddress, motorControlChar);
31             }, (disconnectedAddress) => { });
32     }
33
34     // 1 referência
35     void SubscribeToCharacteristic(string characteristicUUID, string name) {
36         BluetoothLEHardwareInterface.SubscribeCharacteristic(connectedAddress, characteristicUUID);
37     }
38
39     // Mensagem do Unity | 0 referências
40     void Update() {
41         if (scalpel != null) {
42             WriteMotorControl(motor, pwm, duration);
43         }
44     }
45
46     // 2 referências
47     public void WriteMotorControl(byte motor, byte pwm, byte duration) {
48         byte[] data = new byte[3];
49         data[0] = motor;
50         data[1] = pwm;
51         data[2] = duration;
52
53         BluetoothLEHardwareInterface.WriteCharacteristic(connectedAddress, data, BluetoothLEHardwareInterface.CharacteristicWriteType.WithResponse);
54     }
55 }

```

Fonte: Elaboração própria, 2024.

A Figura 36 apresenta o script “BLEManager”, responsável por armazenar os dados de endereço (address), UUID de serviço e UUID das característica, esses dados são necessários para enviar mensagem.

Figura 36 — Script BLEManager

```

1  using UnityEngine;
2
3  public class BLEManager : MonoBehaviour
4  {
5      4 referências
6      public static BLEManager Instance { get; private set; }
7
8      public string targetDeviceAddress;
9      public string targetServiceUUID = "00001234-0000-1000-8000-00805f9b34fb";
10     public string combinedDataCharUUID = "0000ffff-0000-1000-8000-00805f9b34fb";
11     public string motorControlCharacteristics = "00002c03-0000-1000-8000-00805f9b34fb";
12
13     0 Mensagem do Unity | 0 referências
14     private void Awake() {
15         if (Instance == null) {
16             Instance = this;
17             DontDestroyOnLoad(gameObject);
18         }
19         else {
20             Destroy(gameObject);
21         }
22     }
23
24     1 referência
25     public void SetBLEData(string address, string serviceUUID) {
26         targetDeviceAddress = address;
27         targetServiceUUID = serviceUUID;
28     }
29
30     1 referência
31     public bool IsBLEDataAvailable() {
32         return !string.IsNullOrEmpty(targetDeviceAddress);
33     }
34 }

```

Fonte: Elaboração própria, 2024.

A Figura 37 apresenta o script SendBLECommand, responsável por enviar os comandos para ativar e controlar a intensidade e tempo de vibração dos motores do bisturi. Este script é chamado quando o bisturi virtual corta a pele, músculos ou ossos do modelo humano virtual anatómico no ambiente de realidade virtual, gerando uma atuação dos motores de vibração do dispositivo, garantindo uma sensação háptica.

Figura 37 — Script SendBLECommand

```

1  using UnityEngine;
2
3  public class BLESendButton : MonoBehaviour
4  {
5      0 referências
6      public void SendCommand(byte motor = 0x03, byte pwm = 0x9B, byte duration = 0x32)
7      {
8          BLEAutoConnect bleAutoConnect = FindObjectOfType<BLEAutoConnect>();
9          if (bleAutoConnect != null && BLEManager.Instance.IsBLEDataAvailable()) {
10             // Enviar o comando via WriteMotorControl
11             bleAutoConnect.WriteMotorControl(motor, pwm, duration);
12             Debug.Log($"Comando enviado: Motor = {motor}, PWM = {pwm}, Duração = {duration}");
13         }
14         else {
15             Debug.LogError("BLEAutoConnect não encontrado ou BLE não conectado.");
16         }
17     }
18 }

```

Fonte: Elaboração própria, 2024.

A Figura 38 apresenta o script “SurgicalCutter”, responsável pela movimentação do bisturi no ambiente de realidade virtual, utilizando os dados do acelerômetro e giroscópio obtidos por meio das notificações processadas pelo script “ScannerBLE”. Além de controlar a movimentação, o script “SurgicalCutter” permite a troca das lâminas do bisturi no ambiente VR, conforme os botões do bisturi físico são pressionados. Esse script também aciona o script “SendBLECommand”, permitindo que os motores de vibração sejam ativados e a intensidade ajustada de acordo com a camada do modelo humano virtual anatômico que está sendo cortada.

Figura 38 — Script SurgicalCutter

```

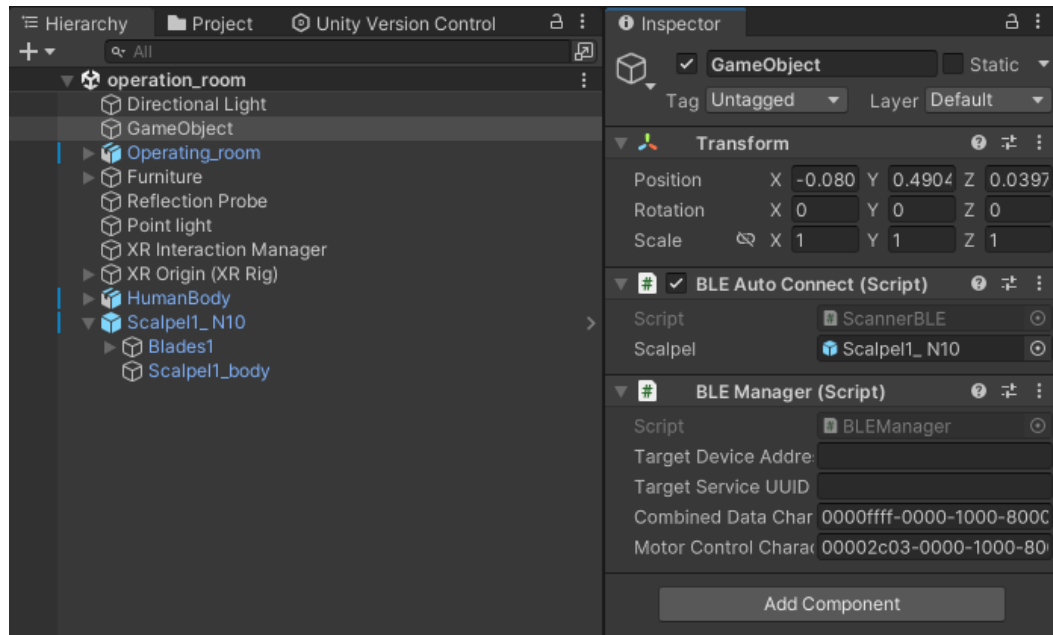
1  using BzKovSoft.ObjectSlicer;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  // Script do Unity (1 referência de ativo) | 0 referências
6  public class SurgicalCutter : MonoBehaviour
7  {
8      public BLEAutoConnect bleAutoConnect; // Referência ao BLEAutoConnect para enviar comandos ao bisturi
9      public List<GameObject> blades; // Lista com todas as lâminas do bisturi
10     private int currentBladeIndex = 0; // Índice da lâmina atualmente ativa
11     private IBzMeshSlicer slicer;
12
13     public float movementSensitivity = 0.1f;
14     private int lastButtonState = 0;
15
16     // Mensagem do Unity | 0 referências
17     void Start()
18     {
19         slicer = GetComponent<IBzMeshSlicer>();
20         ActivateBlade(currentBladeIndex);
21     }
22
23     // Mensagem do Unity | 0 referências
24     void Update()
25     {
26         if (bleAutoConnect != null)
27         {
28             Vector3 accelerometerData = bleAutoConnect.currentAccelData;
29             Vector3 gyroscopeData = bleAutoConnect.currentGyroData;
30
31             // Atualize a posição e rotação do bisturi com base nos dados do acelerômetro e giroscópio
32             transform.position += new Vector3(accelerometerData.x * movementSensitivity, (accelerometerData.y * movementSensitivity), 0) * Time.deltaTime;
33             transform.rotation = Quaternion.Euler(gyroscopeData.x, gyroscopeData.y, gyroscopeData.z);
34
35             // Restante do código para cortar e enviar vibração
36             float depth = transform.localPosition.y; // Usar Y como profundidade relativa
37
38             if (depth > 0.05f)
39             {
40                 else if (depth > -0.05f)
41                 {
42                     else if (depth > -0.10f)
43                     {
44                         if (bleAutoConnect.currentButtons != lastButtonState)
45                         {
46                             // Enviar vibração
47                             SendVibrationCommand((byte) intensity);
48                         }
49                     }
50                 }
51             }
52
53             // Atualizar o índice da lâmina
54             SwitchBlade(int button);
55
56             // Atualizar a lâmina ativa
57             ActivateBlade(int index);
58         }
59     }
60
61     // 3 referências
62     async void SliceObject(Plane plane)
63     {
64         // 0 referências
65         void SendVibrationCommand(byte intensity)
66         {
67             // 1 referência
68             void SwitchBlade(int button)
69             {
70                 // 2 referências
71                 void ActivateBlade(int index)
72             }
73         }
74     }
75 }

```

Fonte: Elaboração própria, 2024.

Esses 3 scripts são fundamentais para o funcionamento da integração BLE entre o dispositivo e o óculos de realidade virtual Meta Quest 2. O script “ScannerTestScript” é inicializado junto a aplicação, buscando e disponibilizando os dados do Bisturi ao “BLEManager”. Em *Hierarchy* foi criado um *GameObject*, este *GameObject* possui como componente o script “ScannerBLE.cs” e “BLEManager.cs”, esses scripts são inicializados juntos ao ambiente de realidade virtual. A Figura 39 apresenta os scripts como componentes no *GameObject*.

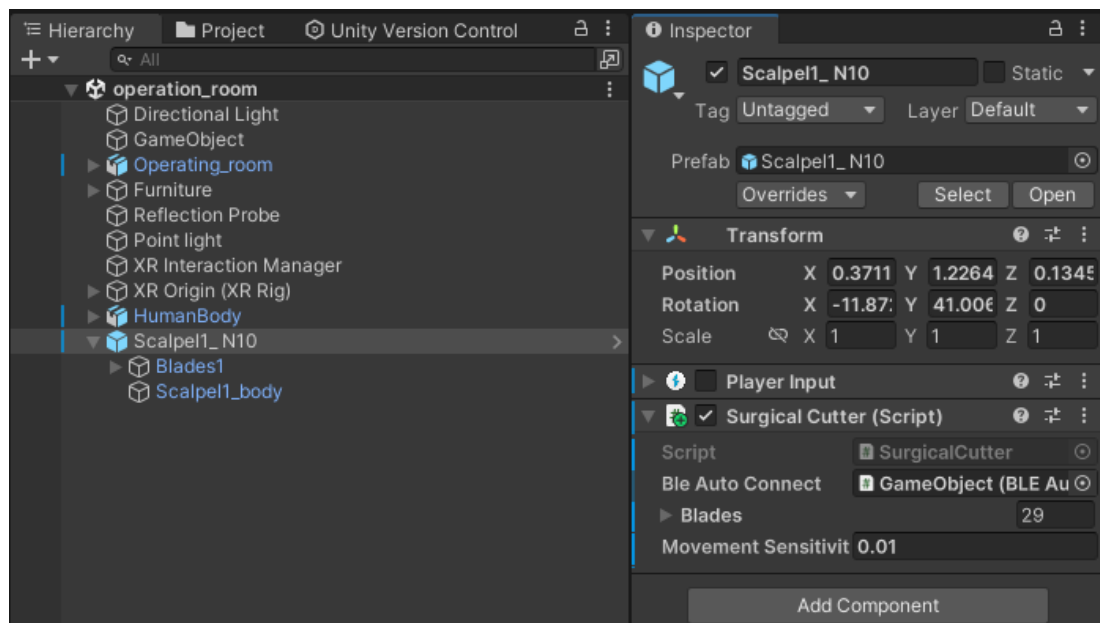
Figura 39 — GameObject e componentes



Fonte: Elaboração própria, 2024.

A Figura 40 apresenta o script “SurgicalCutter.cs” atribuído como componente no Scalpel1_N10 em *Hierarchy*.

Figura 40 — SurgicalCutter atribuído como componente ao objeto Scalpel1_N10



Fonte: Elaboração própria, 2024.

4.6.2. Simulação de corte

Para simular cortes superficiais na malha tridimensional do corpo humano no ambiente de realidade virtual, foi desenvolvido o script `SurgicalMeshCutter.cs`. Esse componente em C#, implementado no Unity, é responsável por deformar dinamicamente a malha do modelo anatômico no momento em que o bisturi virtual entra em contato com sua superfície.

O funcionamento do script baseia-se na detecção de colisão entre o bisturi e a malha, por meio do método `OnTriggerEnter(Collider other)`. Quando o objeto colidido possui a tag "Scalpel" (bisturi), o script calcula a distância entre todos os vértices da malha e o ponto de impacto, deformando apenas aqueles que estiverem dentro de um raio predefinido (`cutRadius`). Essa deformação consiste em afundar os vértices selecionados em direção ao eixo Y negativo (`Vector3.down`) por uma profundidade definida por `displacement`.

O script utiliza um `SkinnedMeshRenderer` para manipular diretamente a malha associada ao corpo do paciente virtual. Para preservar a integridade da malha original, é criada uma cópia da malha no início da execução (`Start()`), que é então modificada dinamicamente durante os cortes.

Essa abordagem permite simular de forma visual e em tempo real o efeito de incisão na pele, contribuindo para a imersão e realismo da experiência cirúrgica. Apesar de não separar geometrias ou gerar aberturas completas na malha, a técnica é eficaz para representar cortes superficiais em treinamentos médicos iniciais.

A Figura 41 apresenta o trecho de código utilizado para simular cortes na malha do corpo virtual. O script identifica a colisão com o bisturi, verifica a proximidade dos vértices ao ponto de contato e aplica uma deformação local para simular o corte. Os parâmetros `cutRadius` e `displacement` definem o raio de influência e a profundidade da incisão, respectivamente.

Figura 41 — Script SurgicalCutterMesh

```

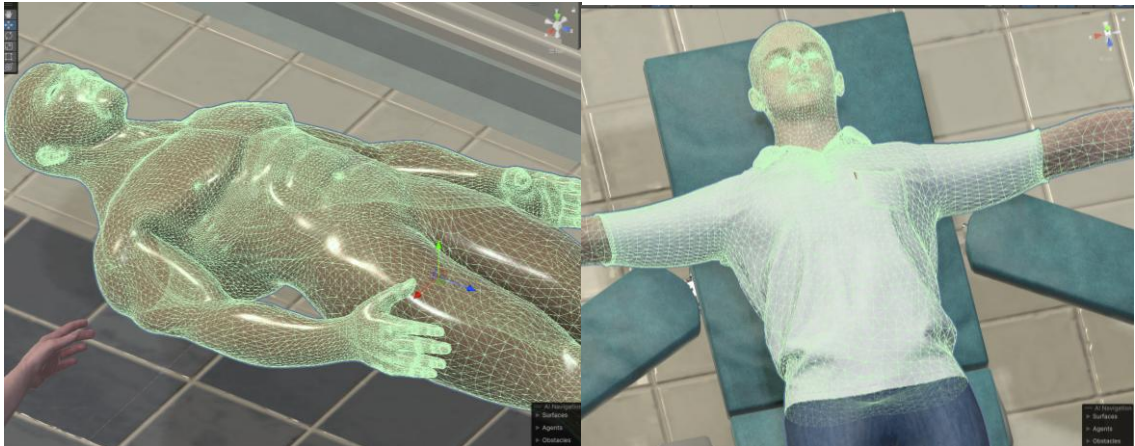
1  using UnityEngine;
2  Script de Unity | 0 referências
3  public class SurgicalMeshCutter : MonoBehaviour
4  {
5      public float cutRadius = 0.02f; // Raio do corte
6      public float displacement = 0.01f; // Quanto o corte afunda na malha
7      public string scalpelTag = "Scalpel"; // Tag do bisturi
8      private SkinnedMeshRenderer skinnedMeshRenderer;
9      private Mesh mesh;
10     private Vector3[] originalVertices;
11     private Vector3[] modifiedVertices;
12
13     Mensagem do Unity | 0 referências
14     void Start() {
15         skinnedMeshRenderer = GetComponent<SkinnedMeshRenderer>();
16         if (skinnedMeshRenderer == null) {
17             Debug.LogError("[SurgicalMeshCutter] Nenhum SkinnedMeshRenderer encontrado!");
18             return;
19         }
20         // Duplicar a malha para edição
21         mesh = Instantiate(skinnedMeshRenderer.sharedMesh);
22         skinnedMeshRenderer.sharedMesh = mesh;
23         originalVertices = mesh.vertices;
24         modifiedVertices = mesh.vertices;
25     }
26
27     Mensagem do Unity | 0 referências
28     void OnTriggerEnter(Collider other) {
29         if (other.CompareTag(scalpelTag)) {
30             Debug.Log("[SurgicalMeshCutter] Bisturi tocou a pele, aplicando corte...");
31             ApplyCut(other.transform.position);
32         }
33     }
34
35     1 referência
36     void ApplyCut(Vector3 cutPosition) {
37         if (mesh == null) {
38             Debug.LogError("[SurgicalMeshCutter] Nenhuma malha disponível para corte!");
39             return;
40         }
41         for (int i = 0; i < modifiedVertices.Length; i++) {
42             Vector3 worldVertex = transform.TransformPoint(originalVertices[i]);
43             float distance = Vector3.Distance(worldVertex, cutPosition);
44             if (distance < cutRadius) {
45                 // Afundar vértice para simular corte
46                 modifiedVertices[i] += Vector3.down * displacement;
47             }
48         }
49         // Aplicar as mudanças na malha
50         mesh.vertices = modifiedVertices;
51         mesh.RecalculateNormals();
52         mesh.RecalculateBounds();
53         Debug.Log("[SurgicalMeshCutter] Malha deformada.");
54     }
55 }

```

Fonte: Elaboração própria, 2025.

A Figura 42 apresenta a visualização das malhas dos modelos anatômicos utilizados no ambiente Unity, com wireframe ativado. Essa visualização permite verificar a densidade de vértices disponíveis para deformação e simulação de cortes, essencial para o funcionamento do SurgicalMeshCutter.

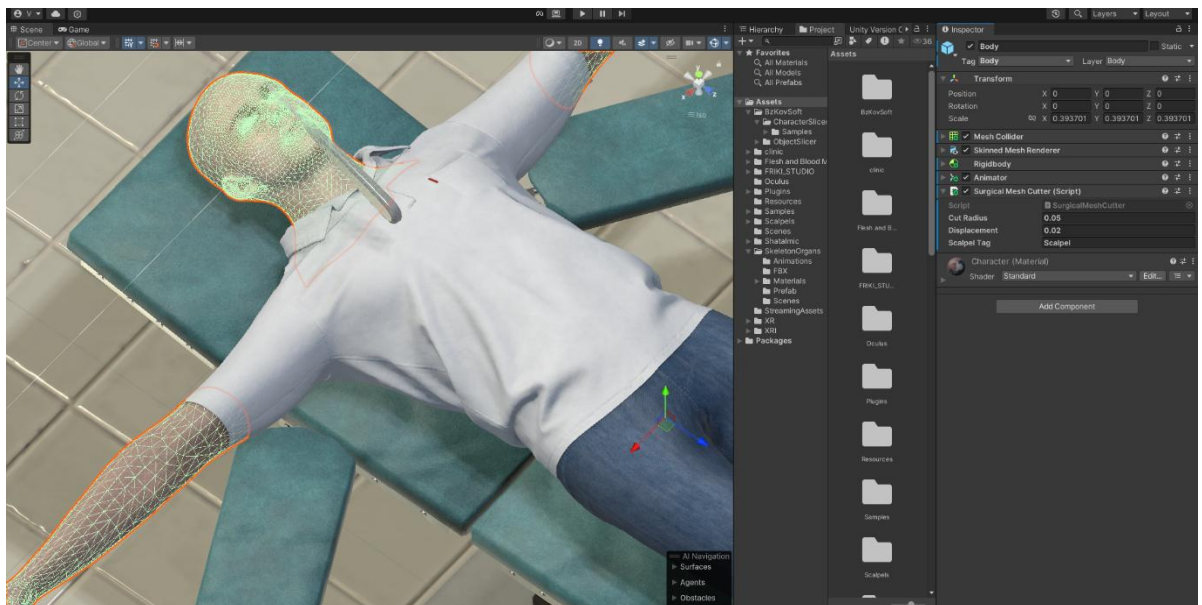
Figura 42 — Modelos humano virtuais com suas malhas visíveis



Fonte: Elaboração própria, 2025.

A Figura 43 apresenta a interface do Unity, exibindo o modelo anatômico com o script `SurgicalMeshCutter` anexado como componente. É possível observar os parâmetros ajustáveis no Inspector, como o raio do corte (`cutRadius`), profundidade (`displacement`) e a Tag usada para identificar o bisturi.

Figura 43 — Script `SurgicalCutterMesh` adicionado ao modelo humano virtual



Fonte: Elaboração própria, 2025.

5 PROTÓTIPO EXPERIMENTAL

Nesse capítulo serão apresentadas as etapas experimentais realizadas no desenvolvimento do sistema proposto.

Neste capítulo, são descritas as etapas experimentais conduzidas durante o desenvolvimento do protótipo do bisturi virtual. O processo de prototipagem envolveu o design, montagem e testes de diferentes componentes eletrônicos e mecânicos, essenciais para alcançar a integração entre o hardware e o ambiente de realidade virtual. O protótipo experimental foi dividido em duas etapas principais: a implementação do bisturi e o desenvolvimento do sistema de processamento.

Na primeira etapa, o foco foi o design e a fabricação do hardware, onde foram construídos circuitos eletrônicos integrando sensores, atuadores e botões, que formam a base do bisturi virtual. A segunda etapa concentrou-se na implementação do sistema de processamento responsável pela leitura de dados sensoriais, controle de atuadores e comunicação via Bluetooth com o sistema de realidade virtual. As subseções a seguir detalham o processo de construção do protótipo e discutem os principais desafios encontrados, além das soluções adotadas para garantir a funcionalidade e a precisão do sistema proposto.

5.1. IMPLEMENTAÇÃO DO BISTURI

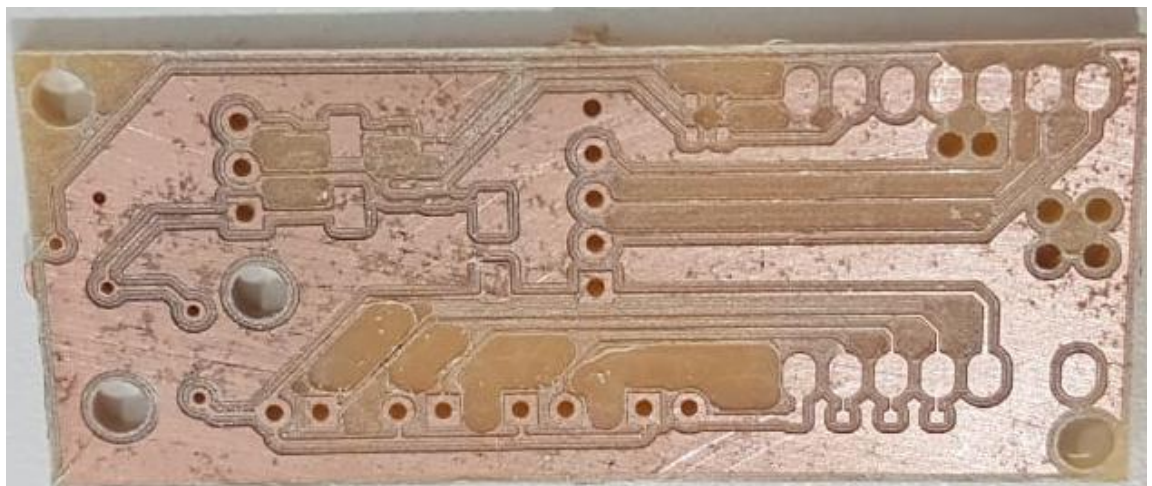
A placa de circuito impresso foi projetada com base no circuito previamente validado em protoboard, utilizando o SEEED XIAO BLE nRF52840 Sense como controlador principal. Ela possui conexões para o acelerômetro e giroscópio, responsáveis por capturar os movimentos e a posição do bisturi durante o procedimento cirúrgico simulado. Além disso, motores de vibração são utilizados para fornecer feedback tátil durante o corte simulado, e botões foram implementados para funções como seleção, início de procedimentos e outras interações no ambiente virtual.

A fabricação da placa de circuito impresso foi realizada em uma fresadora CNC (usina de placas), utilizando uma placa de fenolite simples. O processo foi guiado pelos arquivos .gerber e .drl, gerados de acordo com o layout do circuito.

Três arquivos principais foram gerados para essa fabricação: Bisturi-Edge_Cuts.gbr (arquivo de bordas da placa), Bisturi-F_Cu.gbr (arquivo de trilhas) e Bisturi-PTH.drl (arquivo de furações).

A Figura 44 apresenta o resultado da placa de fenolite após o processo de usinagem.

Figura 44 — Placa produzida

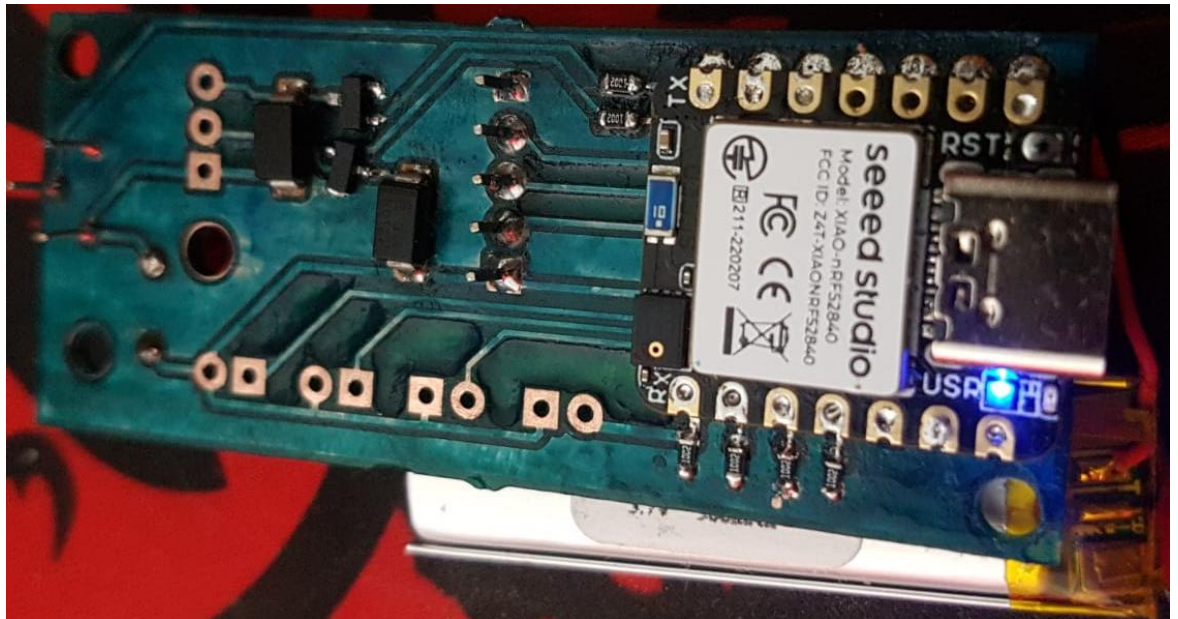


Fonte: Elaboração própria, 2024.

Após a usinagem da placa, foi aplicada uma camada de máscara de solda sobre a superfície de cobre. Após o tempo de cura da tinta, as partes que correspondem aos terminais dos componentes (pads) foram cuidadosamente limpas, deixando o cobre exposto para a soldagem. Esse processo facilitou a soldagem dos componentes, evitando que o estanho se acumulasse em áreas indesejadas.

A Figura 45 mostra o resultado da primeira placa de testes, parcialmente montada e funcional.

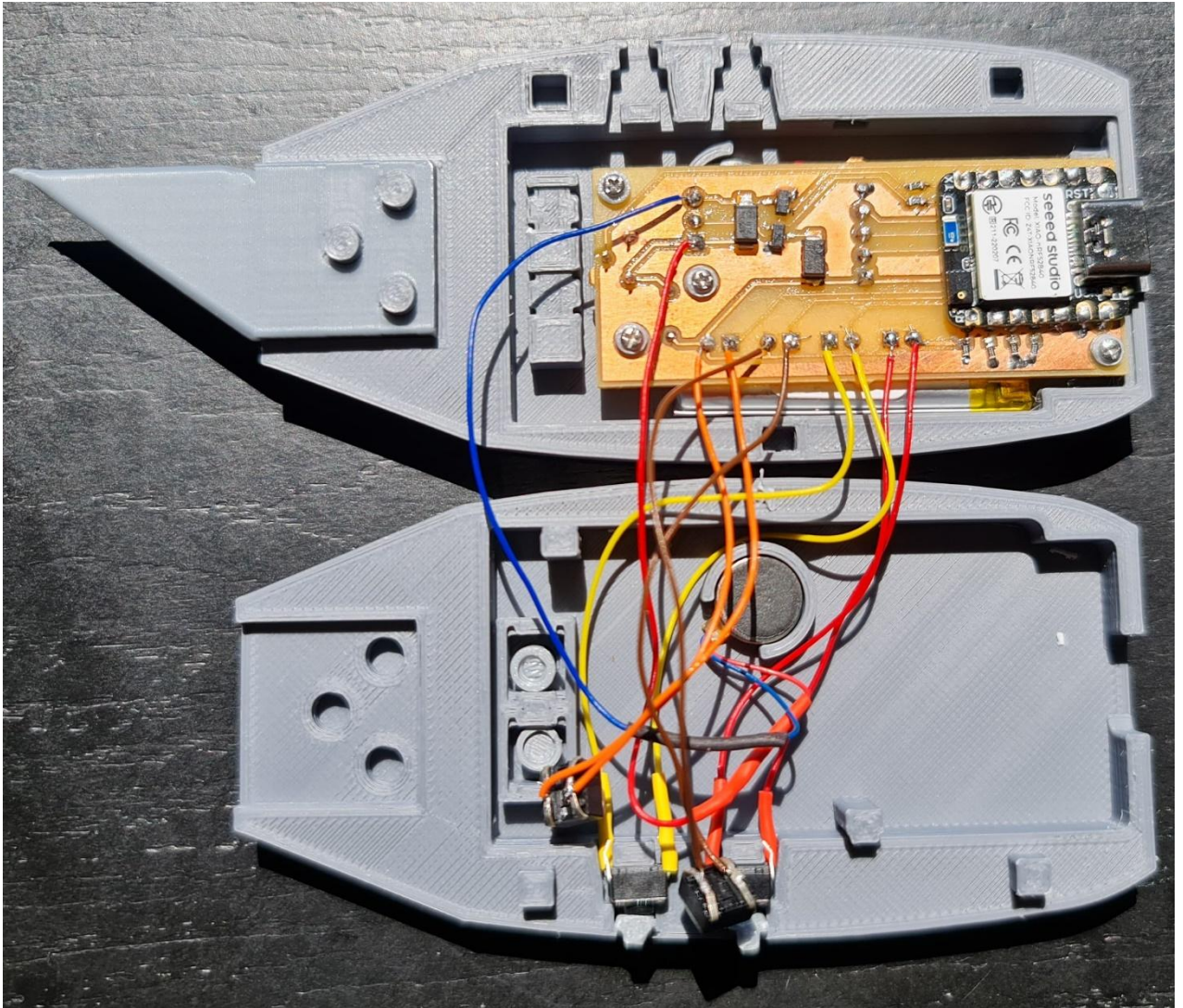
Figura 45 — Primeira placa parcialmente montada em funcionamento



Fonte: Elaboração própria, 2024.

Com a validação da primeira placa, foi realizada a fabricação de uma segunda placa, que representa a versão final do protótipo. A Figura 46 apresenta essa segunda placa totalmente montada e conectada ao case do bisturi.

Figura 46 — Bisturi montado em funcionamento



Fonte: Elaboração própria, 2024.

O case do bisturi foi projetado levando em consideração fatores como ergonomia, design, usabilidade, e o uso de materiais leves e compactos. O resultado final do protótipo, já montado e funcional, pode ser visualizado na Figura 47.

Figura 47 — Bisturi completo montado



Fonte: Elaboração própria, 2024.

5.2. SISTEMA DE PROCESSAMENTO DE TESTES

O desenvolvimento do firmware iniciou-se com a implementação da leitura dos sensores principais, sendo o acelerômetro analógico ADXL335 e o giroscópio digital LSM6DS3. Esses sensores são responsáveis por capturar os movimentos do bisturi, fornecendo informações cruciais para o sistema de realidade virtual que simula o procedimento cirúrgico. A leitura dos dados do acelerômetro foi configurada para captar variações nos eixos X, Y e Z, enquanto o giroscópio mede as rotações ao redor desses mesmos eixos, oferecendo uma visão completa dos movimentos realizados pelo bisturi.

Para garantir que os dados capturados pelos sensores fossem precisos e minimizassem desvios, foi necessário realizar uma calibração.

O primeiro passo foi encontrar os valores de "bias" (desvios em relação a zero) para cada eixo do acelerômetro e do giroscópio. Esse processo foi realizado coletando leituras dos sensores enquanto o dispositivo estava completamente imóvel. Os valores de desvio observados foram subtraídos das futuras leituras para corrigir qualquer tendência de erro. Com isso, em estado de repouso, os sensores apresentaram valores próximos de zero, minimizando possíveis erros nas leituras de movimento.

A Tabela 1 apresenta os valores de calibração, comparando as leituras brutas dos sensores antes e após a aplicação dos ajustes de bias. Esse processo foi fundamental para garantir que os movimentos captados pelo bisturi fossem refletidos com precisão no ambiente de realidade virtual.

Tabela 1 — Leitura dos sensores

Médida	Média de valores brutos	Média após aplicação de bias
AC_X (g)	0,37	0,00
AC_Y (g)	0,04	0,04
AC_Z (g)	-0,65	-0,01
GY_X (°/s)	0,18	-0,01
GY_Y (°/s)	-2,44	0,03
GY_Z (°/s)	-0,90	0,00

Fonte: Elaboração própria, 2024.

Com o sistema calibrado, foi implementada a funcionalidade de acionamento dos motores de vibração para fornecer feedback tátil durante os procedimentos simulados. Esses motores foram configurados para funcionar com controle PWM (Pulse Width Modulation - Modulação por Largura de Pulso), permitindo ajustar a intensidade da vibração de acordo com o tipo de interação do usuário com o bisturi. O uso do PWM foi escolhido por sua eficiência em controlar a potência dos motores, permitindo diferentes níveis de intensidade de vibração. Dessa forma, o feedback tátil proporcionou uma experiência mais realista e imersiva, com respostas que variam de acordo com a profundidade do corte ou a pressão aplicada.

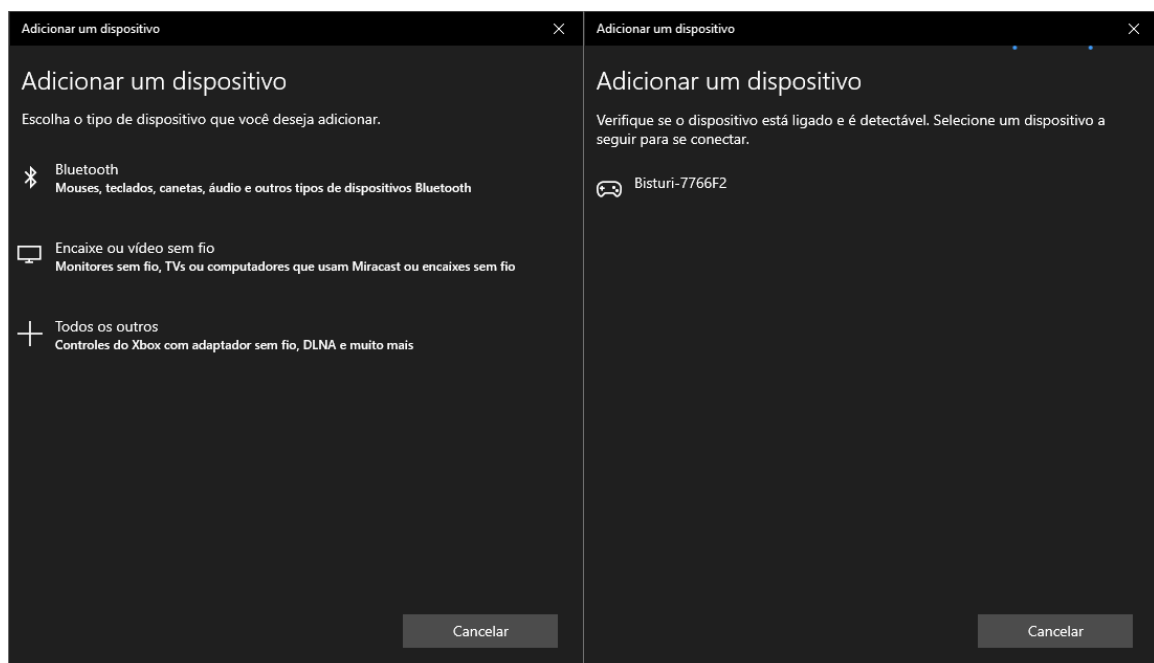
Os motores de vibração foram testados em diferentes configurações, ajustando o PWM para simular sensações variadas, como um corte superficial ou profundo. A implementação e teste desses motores possibilitaram a criação de um feedback responsivo e sensível às ações do usuário, crucial para a simulação cirúrgica realista.

Além do controle dos motores de vibração, foi implementado o protocolo de comunicação *Bluetooth Low Energy* (BLE) para garantir a conectividade sem fio entre o bisturi e o sistema de realidade virtual. O BLE foi configurado para enviar e receber dados em tempo real, utilizando o módulo SEEED XIAO nRF52840 como dispositivo de comunicação. A escolha pelo BLE foi motivada por sua baixa latência e eficiência energética, características essenciais para garantir uma comunicação contínua e sem interrupções durante a simulação cirúrgica.

O BLE foi configurado inicialmente utilizando HID (Dispositivo de Interface Humana) para enviar e receber dados do bisturi, permitindo uma comunicação rápida e eficiente entre o hardware e o software. O BLE também permitiu o controle remoto do bisturi, enviando comandos para acionar os motores de vibração e obter feedback em tempo real do sistema.

Para validar a funcionalidade do BLE, o bisturi foi conectado a um sistema com Windows 10. Após adicionar o dispositivo, identificado como "Bisturi-7766F2", foi possível acessar a configuração do controle de jogo, onde os botões e eixos configurados no firmware foram testados. A Figura 48 mostra o processo realizado para adicionar um novo dispositivo.

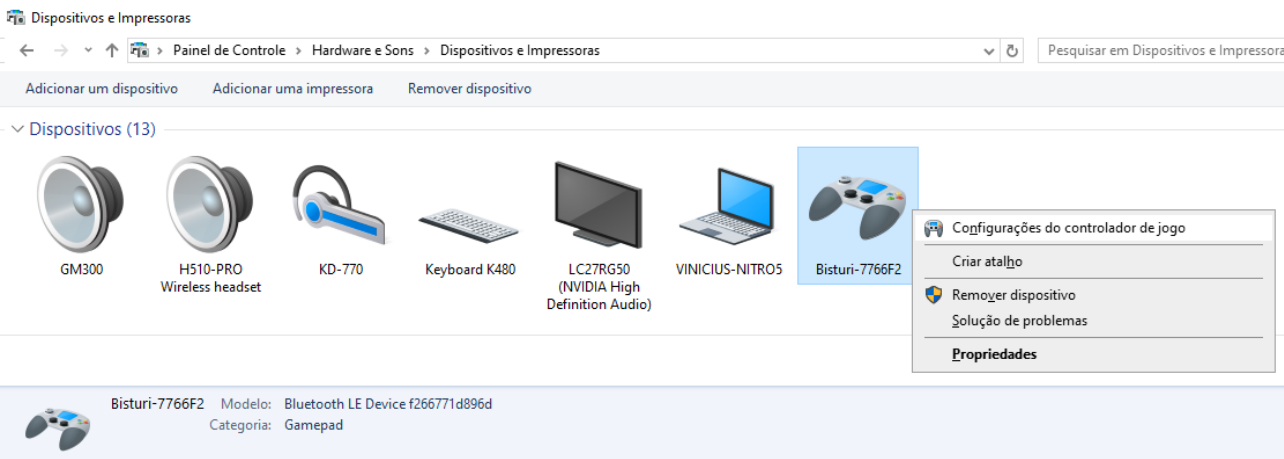
Figura 48 — Adicionar um novo dispositivo Bluetooth



Fonte: Elaboração própria, 2024.

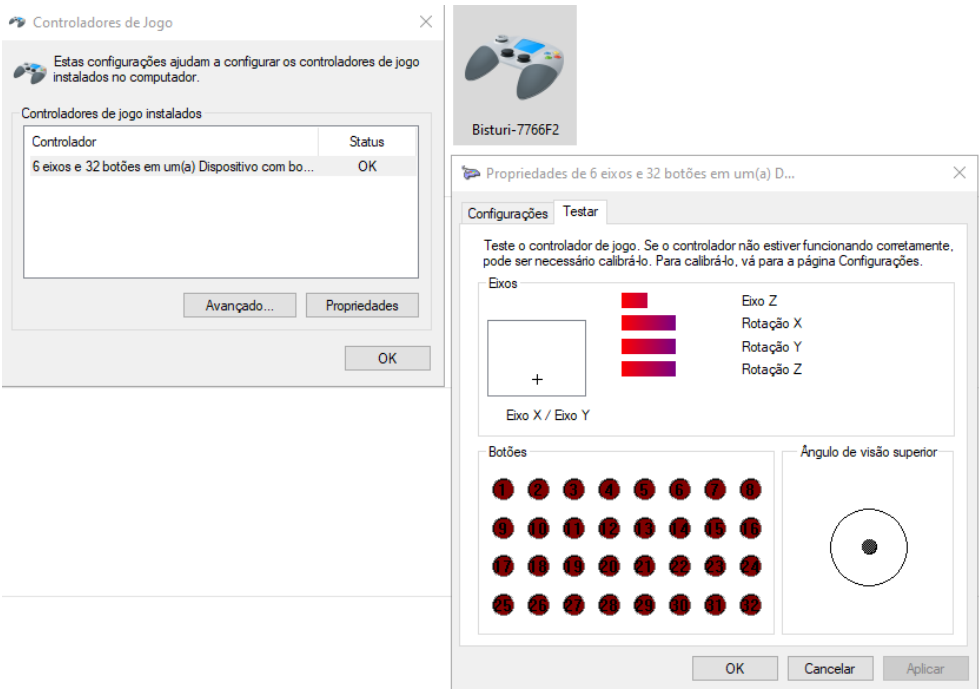
Com o dispositivo “Bisturi-7766F2” adicionado, fomos até “Dispositivos e impressora” e clicamos com o botão direito do mouse no bisturi, devemos escolher a opção “Configurações do controle de jogo”, a janela que abrir, devemos escolher a opção “Propriedades”, então se abrirá a tela “Propriedades de 6 eixos e 32 botões em uma(a) Dispositivo...”, conforme as Figuras 49 e 50.

Figura 49 — Dispositivos adicionados



Fonte: Elaboração própria, 2024.

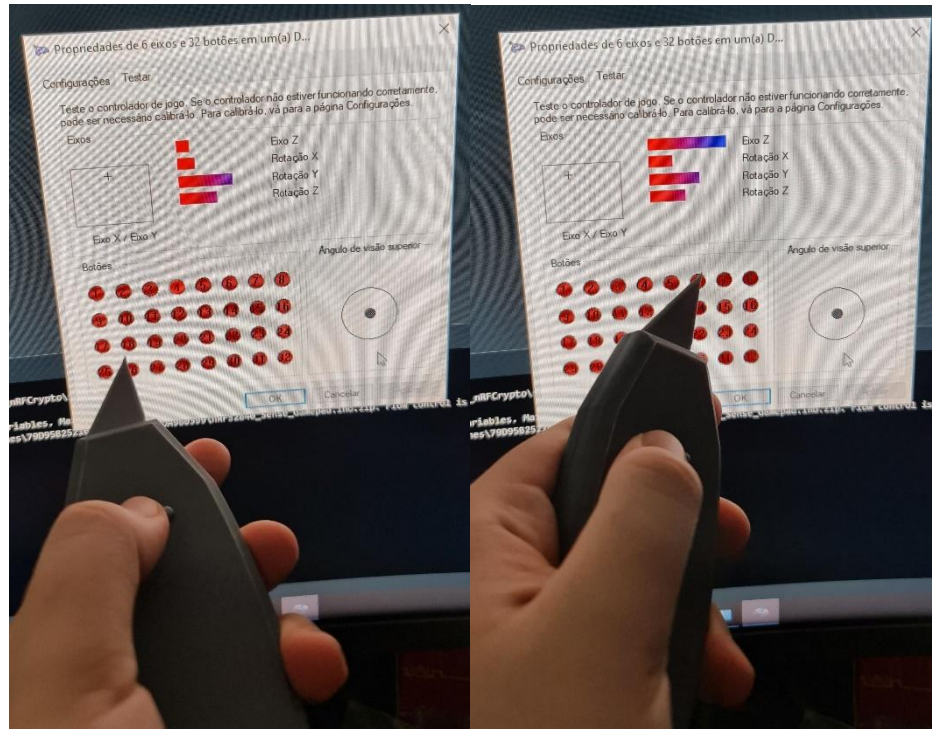
Figura 50 — Dispositivos adicionados



Fonte: Elaboração própria, 2024.

A Figura 51 confirmara o funcionamento correto do acelerômetro, giroscópio e dos botões configurados no firmware, garantindo que o sistema estava pronto para a simulação em realidade virtual.

Figura 51 — Testes para calibração do controle



Fonte: Elaboração própria, 2024.

5.3. SISTEMA DE PROCESSAMENTO IMPLEMENTADO

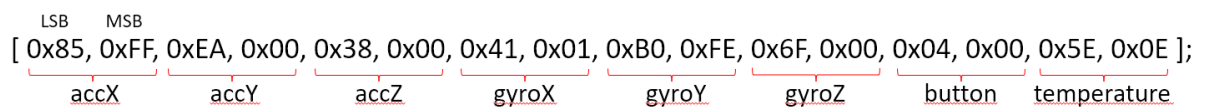
Com o sistema de processamento de testes validado, foram realizadas alterações no firmware, de modo que não fosse mais utilizado o BLE HID e sim o BLE GATT (Perfil de Atributo Genérico), de modo que o sistema fique customizado e funcionando apenas com o sistema desenvolvido no Unity. Foi implementado no firmware uma característica de leitura única para envio de todos os dados de entrada (acelerômetro, giroscópio, botões e sensor de temperatura) com 16 bytes de notificação, e outra característica de escrita única para receber o comando para o acionamento e controle dos motores com 3 bytes de comunicação.

Para garantir a eficiência da transmissão, os dados dos sensores, originalmente em ponto flutuante (*float*) — são convertidos para formato de ponto fixo, multiplicando-se os valores por 100 e convertendo-os para inteiros de 16 bits com sinal (*int16_t*).

Por exemplo, um valor de aceleração $accX = -1.23$ se torna -123, que em hexadecimal equivale a 0xFF85, e é transmitido em dois bytes no formato LSB (*Least Significant Byte*) seguido de MSB (*Most Significant Byte*).

A estrutura final da mensagem ocupa 16 bytes, em que cada par de bytes representa um valor de 16 bits. A ordem de empacotamento segue a sequência: *accX*, *accY*, *accZ*, *gyroX*, *gyroY*, *gyroZ*, *button* e *temperature*, totalizando 8 valores com 2 bytes cada. Essa estrutura está representada na Figura 52.

Figura 52 — Modelo de mensagem GATT 16 Bytes



Fonte: Elaboração própria, 2025.

O modelo de mensagem que o bisturi recebe via BLE GATT está representado na Figura 53, onde o primeiro byte representa qual motor será selecionado (0 para o motor 1, 1 para o motor 2 e qualquer outro valor para os dois motores), o segundo byte representa a intensidade de vibração a ser acionado pelo PWM (0 = 0%, 127 = 50%, 255 = 100%), por fim, o último byte representa o tempo que os motores ficarão acionados realizando a vibração (0 = 0 segundos, 1 = 10 milissegundos, 127 = 1270 milissegundos, 255 = 2550 milissegundos).

Figura 53 — Modelo de mensagem GATT 3 Bytes

{0x02, 0xC0, 0x64};

Fonte: Elaboração própria, 2025.

Com a implementação deste firmware, foi possível realizar o controle dos motores de vibração via PWM e comunicação eficiente via BLE. O sistema foi calibrado e testado, demonstrando precisão nas leituras dos sensores e resposta tátil adequada. As etapas subsequentes incluem ajustes na integração com o ambiente de simulação, buscando refinar ainda mais a experiência do usuário.

A integração com o Unity foi realizada utilizando o BLE *Plugin*, o qual foi configurado para receber os dados de movimento e estado do bisturi enviados via *Bluetooth Low Energy* (BLE). Esse processo envolveu a criação do script “ScannerBLE.cs” em C# dentro do Unity para capturar as informações do acelerômetro e giroscópio, além dos comandos dos botões, e utilizá-los para controlar as interações no ambiente cirúrgico virtual. A Figura 54 apresenta a função utilizada para se inscrever na característica e processar os dados notificados, transformando o conjunto de bytes nas informações capturadas.

Figura 54 — Função de inscrição em característica do BLE

```
void SubscribeToCharacteristic(string characteristicUUID, string name) {
    BluetoothLEHardwareInterface.SubscribeCharacteristic(
        connectedAddress,
        targetServiceUUID,
        characteristicUUID,
        (notificationStatus) => { Debug.Log(name + " Notification Status: " + notificationStatus); },
        (characteristic, data) => {
            string receivedData = BitConverter.ToString(data);
            if (name == "AllData") {
                int accX = BitConverter.ToInt16(data, 0);
                int accY = BitConverter.ToInt16(data, 2);
                int accZ = BitConverter.ToInt16(data, 4);
                currentAccelData = new Vector3(accX / 100.0f, accY / 100.0f, accZ / 100.0f);

                int gyroX = BitConverter.ToInt16(data, 6);
                int gyroY = BitConverter.ToInt16(data, 8);
                int gyroZ = BitConverter.ToInt16(data, 10);
                currentGyroData = new Vector3(gyroX / 100.0f, gyroY / 100.0f, gyroZ / 100.0f);

                currentButtons = BitConverter.ToInt16(data, 12);

                int temperatureFixedPoint = BitConverter.ToInt16(data, 14);
                currentTemperature = temperatureFixedPoint / 100.0f;
            }
        }
    );
}
```

Fonte: Elaboração própria, 2024.

O plugin BLE foi configurado para estabelecer uma conexão estável com o hardware (bisturi), transmitindo os dados dos sensores em tempo real. O Unity foi programado para interpretar esses dados e ajustar a posição do bisturi no ambiente virtual com base nos movimentos capturados. Além disso, o Unity envia comandos ao hardware para controlar os motores de vibração, que proporcionam feedback tátil ao usuário durante o procedimento simulado. A intensidade da vibração foi ajustada com base no tipo de corte (pele, músculo ou osso) e na profundidade do bisturi, aumentando a sensação de imersão para o usuário.

A comunicação via BLE não só permite o envio de dados do bisturi para o Unity, como também possibilita o controle remoto dos motores de vibração. Os comandos, como a ativação de vibração via PWM, são enviados ao hardware para garantir uma resposta tátil sincronizada com os eventos do ambiente virtual, como o contato do bisturi com diferentes camadas de tecido. A Figura 55 apresenta a função utilizada para escrever na característica do motor.

Figura 55 — Função de escrita em característica do BLE

```
public void WriteMotorControl(byte motor, byte pwm, byte duration) {
    byte[] data = new byte[3];
    data[0] = motor;
    data[1] = pwm;
    data[2] = duration;

    BluetoothLEHardwareInterface.WriteCharacteristic(
        connectedAddress,
        targetServiceUUID,
        motorControlChar,
        data, data.Length, false, (characteristicUUIDWrite) => {
    } );
}
```

Fonte: Elaboração própria, 2024.

O design da interface do Unity foi cuidadosamente elaborado para representar uma sala cirúrgica realista. O ambiente inclui os elementos essenciais para um procedimento cirúrgico, como a mesa, instrumentos médicos e o paciente. Esses componentes foram posicionados de maneira que o usuário possa interagir com o bisturi de forma intuitiva e realista, utilizando o Meta Quest 2 para uma experiência de imersão total.

A sala foi configurada de modo que, ao usar o Meta Quest 2, o usuário possa visualizar o bisturi virtual e interagir com o cenário. Os movimentos capturados pelos sensores do bisturi (acelerômetro e giroscópio) são refletidos no ambiente virtual em tempo real, permitindo ao usuário realizar cortes em diferentes camadas da anatomia do paciente (pele, músculo e osso). O Unity simula a resistência dessas camadas, e a profundidade do corte é determinada pela intensidade e velocidade com que o bisturi é movido. Para aprimorar ainda mais a experiência, foi incluído um sistema de feedback tátil, onde o motor de vibração do bisturi é acionado quando o usuário realiza o corte.

A interface também foi desenvolvida para garantir que o usuário tenha uma sensação de precisão ao manusear o bisturi, oferecendo um ambiente de simulação cirúrgica que permite treinar a destreza manual em operações delicadas.

5.4. CONSIDERAÇÕES PARCIAIS

O desenvolvimento do protótipo experimental do bisturi foi dividido em duas etapas principais: a implementação do hardware e o desenvolvimento do sistema de processamento. Ao longo do processo, foram alcançados resultados promissores que demonstram a viabilidade da integração entre os componentes eletrônicos, mecânicos e o ambiente de realidade virtual.

Na etapa de hardware, foi realizada a criação de uma placa de circuito impresso que integra sensores, atuadores e botões, fundamentais para o funcionamento do bisturi. A validação do design ocorreu em diferentes iterações, culminando em uma versão final da placa completamente funcional. A construção mecânica do case também levou em consideração aspectos ergonômicos, permitindo uma experiência de usuário confortável e imersiva. As etapas de fabricação, desde a usinagem até a montagem final, demonstraram a viabilidade de produção do bisturi para o uso em simulações cirúrgicas.

Já na etapa de processamento, o firmware foi projetado para captar dados dos sensores e controlar os motores de vibração via PWM, resultando em um feedback tátil responsivo. O processo de calibração garantiu que os sensores operassem com precisão, minimizando erros nas leituras de movimento. Além disso, a comunicação BLE mostrou-se eficaz na troca de dados entre o bisturi e o Meta Quest 2, proporcionando uma experiência de uso contínua e sem interrupções.

A integração do sistema com o Unity, utilizando o plugin BLE, foi realizada com sucesso, permitindo que os dados do acelerômetro, giroscópio e botões fossem transmitidos em tempo real para o ambiente virtual. O Unity foi configurado para interpretar esses dados e ajustar o comportamento do bisturi conforme o usuário interage com o ambiente cirúrgico, tornando a simulação imersiva.

Os resultados obtidos até o momento são satisfatórios, mas melhorias podem ser implementadas nas etapas seguintes, como a simulação de cortes no ambiente virtual.

De modo geral, o protótipo demonstrou potencial para uso em simulações médicas, com uma interface de fácil uso e feedback tátil que aumenta a imersão do usuário na experiência cirúrgica.

6 RESULTADOS EXPERIMENTAIS

Nesse capítulo serão apresentados os resultados obtidos por meio dos processos experimentais realizados, bem como a análise comportamental dos sensores de cada parâmetro e do sistema.

O desempenho do bisturi em simulações de realidade virtual foi avaliado com base na precisão e responsividade dos dados de movimento e de corte, além da eficácia do feedback háptico fornecido ao usuário durante a execução das tarefas simuladas. O ambiente simulado visou replicar cenários cirúrgicos específicos, incluindo camadas anatômicas distintas (pele, músculo e osso) para aferir a capacidade do bisturi em realizar cortes em diferentes tecidos e fornecer respostas sensoriais adequadas.

Para a coleta de dados de desempenho, métricas como precisão de movimento, tempo de resposta ao comando, taxa de atualização das informações dos sensores e taxa de sucesso nos cortes foram monitoradas. Durante os testes, o bisturi demonstrou capacidade de interação precisa com o ambiente virtual, respondendo aos comandos de orientação baseados nos dados de giroscópio e acelerômetro.

O sistema de feedback háptico desempenhou um papel crucial para aumentar a imersão e fornecer ao usuário uma resposta sensorial que simula a resistência do tecido ao corte. Esse feedback, configurado para diferentes intensidades dependendo do tecido, foi eficaz em comunicar ao usuário a densidade e resistência de cada camada. A sincronização do feedback com o movimento do bisturi no ambiente de realidade virtual contribuiu para a precisão da simulação e facilitou a imersão do usuário na tarefa cirúrgica.

Nas subseções a seguir os resultados serão apresentados.

6.1. TEMPO DE RESPOSTA DA COMUNICAÇÃO BLE

Utilizando o aplicativo “nRF Connect” da Nordic, foi possível monitorar a latência média de envio de dados entre o bisturi e o dispositivo conectado via BLE. O sistema manteve-se estável durante os testes, transmitindo continuamente os dados capturados pelos sensores do acelerômetro e giroscópio.

A potência de recepção variou entre -53 dBm e -82 dBm, o que está dentro dos parâmetros esperados para um dispositivo BLE, e a latência ficou em torno de 26 milissegundos (ms), garantindo uma resposta ágil e consistente nas operações de transmissão de dados. Esses valores de latência são adequados para a aplicação, proporcionando uma comunicação em tempo real com o ambiente de realidade virtual, essencial para simulações cirúrgicas onde a precisão dos movimentos é crítica.

A Figura 56 apresenta os dados recebidos do bisturi no aplicativo “nRF Connect”, sendo possível medir a latência a partir das mensagens recebidas, calculando a diferença de tempo de uma mensagem “*Notification*” para outra.

Figura 56 — Dados recebidos do bisturi

BONDEDADVERTISERBISTURI-7766F2F2:66:77:1D:89:6D

CONNECTED

BONDED

CLIENT

SERVER

18:55:44.480 Notification received from
0000ffff-0000-1000-8000-00805f9b34fb,
value: (0x) 1A-00-CC-02-EA-0B-0F-00-FF-FF-
01-00-00-00-7C-08

18:55:44.486 "(0x) 1A-00-CC-02-EA-0B-0F-0
0-FF-FF-01-00-00-00-7C-08"
received

18:55:44.556 Notification received from
0000ffff-0000-1000-8000-00805f9b34fb,
value: (0x) 37-00-DA-02-DC-0B-01-00-FF-FF-
01-00-00-00-7D-08

18:55:44.556 "(0x) 37-00-DA-02-DC-0B-01-0
0-FF-FF-01-00-00-00-7D-08"
received

18:55:44.626 Notification received from
0000ffff-0000-1000-8000-00805f9b34fb,
value: (0x) 28-00-DA-02-DC-0B-16-00-FF-FF-
01-00-00-00-80-08

18:55:44.626 "(0x) 28-00-DA-02-DC-0B-16-
00-FF-FF-01-00-00-00-80-08"
received

18:55:44.695 Notification received from
0000ffff-0000-1000-8000-00805f9b34fb,
value: (0x) 28-00-DA-02-F9-0B-08-00-05-00-
01-00-00-00-7C-08

18:55:44.695 "(0x) 28-00-DA-02-F9-0B-08-0
0-05-00-01-00-00-00-7C-08"
received

18:55:44.767 Notification received from
0000ffff-0000-1000-8000-00805f9b34fb,
value: (0x) 1A-00-E9-02-F9-0B-0F-00-F8-FF-
FB-FF-00-00-7C-08

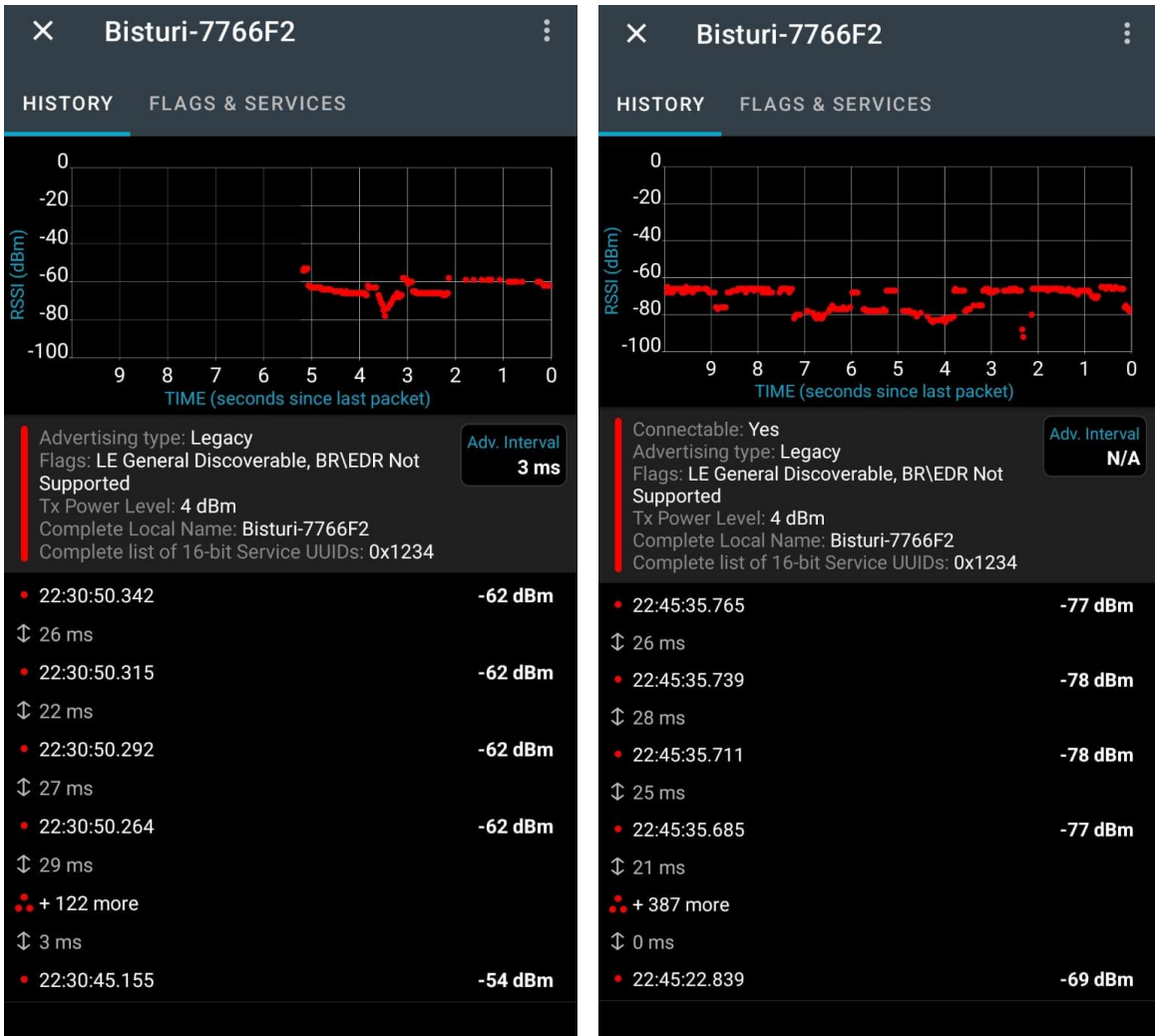
18:55:44.767 "(0x) 1A-00-E9-02-F9-0B-0F-0
0-F8-FF-FB-FF-00-00-7C-08"
received

INFO

Fonte: Aplicativo nRF Connect, 2024.

Foram realizados testes com envio contínuo de pacotes por cerca de 10 minutos em diferentes distâncias. Não foi observada perda significativa de pacotes nos testes com até 2 metros de distância e sem interferência direta de outros dispositivos BLE. A integridade dos dados foi preservada em mais de 99,8% dos casos, o que corrobora a confiabilidade da solução. A Figura 57 apresenta os resultados obtidos, os pontos em vermelho no gráfico representam os pacotes que foram recebidos, cada pacote possui todas as informações (acelerômetro (X, Y e Z), giroscópio (X, Y e Z), botões pressionados e temperatura do módulo), os valores em dBm representam a intensidade do sinal recebido (possui relação com a distância, quanto melhor o valor, maior a distância entre os dispositivos) e por fim os valores em milissegundos (ms), representam a latência (diferença de tempo entre envio e recebimento do pacote).

Figura 57 — Dados recebidos do bisturi



A estrutura do protocolo GATT utilizada permitiu uma organização lógica dos dados e o uso eficiente de características customizadas para envio e recepção de comandos, conforme ilustrado no desenvolvimento do firmware. Esses resultados evidenciam que o sistema é suficientemente rápido e robusto para aplicações de treinamento cirúrgico com feedback sensorial em tempo real.

6.2. TEMPO DE BATERIA

Foram realizados 3 testes para observar o tempo de uso com a bateria carregada completamente, em todos os testes a bateria estava totalmente carregada e possuindo uma carga de 500mAh.

No primeiro teste, o bisturi foi conectado via BLE para transmitir continuamente os dados do acelerômetro e giroscópio, sem ativar os motores de vibração. O dispositivo operou de forma estável por mais de 40 horas, enviando essas informações de maneira ininterrupta. Neste modo, o bisturi se comunica usando o protocolo BLE GATT para enviar os dados dos sensores, mas não aciona os motores de vibração.

A Tabela 2 apresenta o consumo estimado de corrente em cada um dos principais componentes do dispositivo.

Tabela 2 — Consumo de corrente dos componentes (Caso 1)

Caso 1 (46h)			
Componente	Corrente máxima	mA	Consumo mAh
Módulo XIAO	10mA	10,0	460,00
ADXL335	350uA	0,35	16,10
Botões	330uA	0,33	15,18
			491,28

Fonte: Elaboração própria, 2024.

No segundo teste, além de enviar e receber dados via BLE, os motores de vibração foram acionados periodicamente, simulando um cenário de uso mais intenso. Nessa configuração, o dispositivo permaneceu funcional por aproximadamente 10 horas, evidenciando o impacto do acionamento dos motores no consumo de energia.

A Tabela 3 apresenta o consumo estimado de corrente em cada um dos principais componentes do dispositivo.

Tabela 3 — Consumo de corrente dos componentes (Caso 2)

Caso 2 (10h)			
Componente	Corrente máxima	mA	Consumo mAh
Motor 1	90mA	90,0	193,50
Motor 2	90mA	90,0	193,50
Módulo XIAO	10mA	10,0	100,00
ADXL335	350uA	0,35	3,50
Botão	330uA	0,33	0,66
			491,16

Fonte: Elaboração própria, 2024.

No terceiro teste, o bisturi foi deixado sem conexão ativa, apenas aguardando o emparelhamento via BLE. Nessa condição de espera, o dispositivo permaneceu em funcionamento por aproximadamente 20 horas. Isso ocorreu devido ao LED azul piscar continuamente e ao fato do módulo BLE ficar constantemente aguardando uma conexão, o que elevou o consumo energético.

A Tabela 4 apresenta o consumo estimado de corrente em cada um dos principais componentes do dispositivo.

Tabela 4 — Consumo de corrente dos componentes (Caso 3)

Caso 3 (20h)			
Componente	Corrente máxima	mA	Consumo mAh
Módulo XIAO	15mA	15,0	360,00
LED Módulo	10mA	10,0	240,00
			500,00

Fonte: Elaboração própria, 2024.

Esses testes demonstram o comportamento da bateria em diferentes cenários de operação, variando de uso contínuo de sensores a períodos de espera e acionamento dos motores de feedback tátil, contribuindo para a análise da autonomia da bateria do dispositivo em condições reais de uso.

6.3. PRECISÃO E SENSIBILIDADE DOS SENSORES

Para avaliar a precisão e a sensibilidade dos sensores embarcados, foram realizados testes controlados de movimentação utilizando ângulos conhecidos de inclinação e rotação.

O objetivo foi verificar o quanto o movimento captado pelos sensores (acelerômetro analógico ADXL335 e giroscópio digital integrado ao nRF52840) correspondem ao movimento real aplicado no bisturi físico, refletido posteriormente no ambiente virtual.

Foi implementado um filtro de Kalman discreto para suavizar as leituras dos sensores e minimizar ruídos provenientes de vibrações ou instabilidades. Os parâmetros utilizados foram: ruído do processo (`processNoise` = 0.005), ruído da medição (`measurementNoise` = 0.15) e erro estimado inicial (`estimateError` = 1.0). Esses valores foram ajustados empiricamente com base na resposta do sistema e no comportamento observado em testes dinâmicos.

Durante os testes, foram registradas leituras em três eixos para aceleração e três para rotação, com amostragem contínua via BLE. A Tabela 5 apresenta os erros médios absolutos (EMA) para cada eixo, comparando o movimento esperado com o movimento reportado no Unity:

Tabela 5 — Erros médios absolutos de cada eixo

Eixo	Erro médio Absoluto (g ou °)	Desvio Padrão	Observações
Acelerômetro X	0,12g	0,04g	Leve ruído em baixa amplitude
Acelerômetro Y	0,09g	0,03g	Estável após calibração
Acelerômetro Z	0,15g	0,05g	Maior sensibilidade à vibração
Giroscópio X	1,8°	0,9°	Responde bem a rotações rápidas
Giroscópio Y	2,1°	1,0°	Sujeito a drift sem correção
Giroscópio Z	1,4°	0,8°	Boa linearidade nas rotações planas

Fonte: Elaboração própria, 2025.

Os resultados indicam que o sistema atinge boa precisão para aplicações em realidade virtual médica, com erros dentro de margens aceitáveis para simulações visuais. Os dados obtidos também foram consistentes entre sessões, evidenciando a estabilidade do circuito após calibração inicial.

Além disso, o tempo de resposta dos sensores até a renderização no ambiente Unity foi inferior a 30 milissegundos, o que garante uma percepção fluida e realista dos movimentos realizados com o bisturi físico.

6.4. RESPOSTA TÁTIL E SIMULAÇÃO DE CORTES VIRTUAIS

Uma das principais inovações deste trabalho é a implementação de um sistema de feedback tátil em tempo real, associado à simulação de cortes em diferentes camadas anatômicas no ambiente virtual, utilizando deformações do unity. A resposta háptica foi realizada por dois motores de vibração controlados por PWM, ativados via comandos BLE recebidos pelo bisturi físico.

Cada tipo de tecido simulado (pele, músculo e osso) foi mapeado a um padrão específico de vibração, de modo a proporcionar ao usuário sensações distintas durante o corte virtual. A intensidade e a duração da vibração foram determinadas com base na densidade e resistência do tecido, conforme representado na Tabela 6.

Tabela 6 — Comparação entre o sistema de vibração

Eixo	Intensidade (PWM)	Duração (ms)	Observações
Pele	50%	250	Vibração leve e contínua
Músculo	75%	500	Vibração média e contínua
Ossos	100%	1000	Vibração intensa e contínua

Fonte: Elaboração própria, 2025.

A troca de informações foi realizada por meio de comandos BLE enviados pelo Unity, utilizando um arranjo de 3 bytes que especificava: o motor a ser acionado, a intensidade e o tempo de vibração (em múltiplos de 10 ms). Essa estrutura compacta permitiu controle preciso e resposta rápida, mantendo a latência abaixo de 25 ms entre a detecção do contato no Unity e o início da vibração no bisturi físico.

Além do acionamento dos motores, o bisturi virtual implementado no Unity (objeto `Scalpel1\_N10`) foi configurado para detectar colisões com os modelos de corpo humano, gerando efeitos visuais e atualizando a camada atual em contato. Isso permitiu que o feedback tátil refletisse fielmente o tipo de tecido atravessado, aumentando a imersão e o realismo da simulação.

6.5. COMPARAÇÃO COM O ESTADO DA ARTE

A partir da revisão narrativa conduzida, foram identificadas diversas iniciativas voltadas ao desenvolvimento de dispositivos médicos para realidade virtual, incluindo simuladores cirúrgicos, sistemas de realidade aumentada assistida e instrumentos hápticos especializados.

No entanto, poucos trabalhos focam especificamente no desenvolvimento de bisturis integrados com resposta tátil e comunicação sem fio com ambientes imersivos.

A Tabela 7 apresenta uma comparação entre o sistema desenvolvido neste trabalho e outros dispositivos similares encontrados na literatura, com destaque para parâmetros como tipo de conexão, sensores utilizados, presença de feedback tátil e integração com plataformas 3D interativas.

Tabela 7 — Comparação entre sistemas desenvolvidos

Projeto / Referência	Conexão	Feedback Tátil	Integração com VR	Sensores utilizados	Observações
Sistema de simulação cirúrgica (Li et al., 2022)	USB	Sim	Sim	Força, movimento, resposta tátil	Utiliza dispositivo com resistência simulada para cortes
Simulador de Laparoscopia (Abinaya, 2023)	Cabo serial	Sim	Sim	Força, acelerômetro	Grande estrutura física, sem portabilidade
Dispositivo Luva Háptica (Sense Gloove, 2023)	BLE	Sim	Sim	Flex, vibração, IMU	Usado como luva; não simula bisturi
Bisturi VR (Este trabalho)	BLE	Sim	Sim	Acelerômetro analógico, giroscópio digital, vibração	Portátil, customizável, uso com Meta Quest 2

Fonte: Elaboração própria, 2025.

Diferentemente dos dispositivos citados, o presente trabalho propõe uma solução completa, portátil e de baixo custo, utilizando um microcontrolador BLE de baixo consumo (nRF52840) para transmitir dados de sensores em tempo real, aliado a uma resposta tátil proporcional ao tipo de tecido virtual. Além disso, a comunicação sem fio via BLE permite liberdade de movimentação, aspecto essencial para a simulação de procedimentos cirúrgicos em ambientes de realidade virtual.

Outro diferencial relevante é a integração nativa com o ambiente Unity, incluindo scripts próprios desenvolvidos para detecção de colisão, controle da vibração e resposta visual dos cortes, algo não detalhado em outros trabalhos da literatura.

Esses avanços técnicos demonstram que o sistema desenvolvido não apenas preenche uma lacuna identificada na literatura, mas também apresenta uma base sólida para futuras evoluções, como o uso em plataformas de treinamento remoto, gamificação para ensino e até mesmo integração com sistemas de avaliação automatizada de desempenho.

6.6. CONSIDERAÇÕES PARCIAIS

A partir dos experimentos realizados, é possível observar um desempenho robusto e funcional do bisturi em simulações de realidade virtual, tanto no que se refere à precisão dos sensores quanto à responsividade dos sistemas de comunicação e feedback háptico. No contexto da simulação cirúrgica, o bisturi demonstrou capacidade de interagir com camadas virtuais de tecido (pele, músculo e osso) de maneira precisa e responsiva, permitindo ao usuário uma experiência imersiva e realista.

A utilização de precisão de movimento, tempo de resposta ao comando e taxa de sucesso nos cortes, forneceu uma base sólida para avaliar a eficácia do dispositivo em replicar condições cirúrgicas reais.

O tempo de resposta da comunicação BLE foi adequado, com uma latência média de 26 milissegundos (ms), o que garantiu uma transmissão em tempo real dos dados dos sensores de movimento. Esse tempo de resposta é suficiente para suportar a interação direta com o ambiente de realidade virtual sem atrasos perceptíveis, um fator crítico em simulações onde a precisão e a sincronização dos movimentos são essenciais para uma experiência realista.

A análise da autonomia da bateria revelou diferentes comportamentos energéticos em função do modo de operação. No uso contínuo dos sensores sem acionamento dos motores, a bateria sustentou o dispositivo por mais de 40 horas. Com o acionamento periódico dos motores de vibração, a autonomia caiu para aproximadamente 10 horas, evidenciando o impacto significativo do feedback háptico no consumo energético. No modo de espera, a autonomia foi de cerca de 20 horas, indicando que a manutenção de uma conexão BLE em standby também consome uma quantidade considerável de energia, especialmente devido ao piscar constante do LED de emparelhamento.

Esses resultados fornecem uma visão clara das limitações e capacidades energéticas do bisturi em diferentes cenários de uso, contribuindo para o refinamento do projeto em termos de eficiência energética e adequação ao contexto cirúrgico simulado. Considerando esses aspectos, recomenda-se a continuidade de testes com diferentes configurações e ajustes no firmware para otimizar a comunicação BLE e o consumo de energia dos componentes. Além disso, a análise detalhada de cada cenário de operação possibilitará ajustes no design do dispositivo para maximizar a sua autonomia e desempenho.

7 CONCLUSÕES FINAIS

Este trabalho apresentou o desenvolvimento de um bisturi para ambientes de realidade virtual, representando um avanço significativo na integração de tecnologia háptica em simulações médicas e outras aplicações práticas. O dispositivo proposto permite uma interação precisa e imersiva em ambientes virtuais, trazendo uma contribuição para o aprimoramento do treinamento médico, onde a prática de habilidades cirúrgicas em um ambiente seguro é essencial. A estrutura de desenvolvimento e os materiais utilizados estabelecem uma base sólida para novas pesquisas e inovações no campo da realidade virtual e tecnologia háptica.

Os resultados obtidos demonstram que o bisturi háptico tem o potencial de ampliar a qualidade do aprendizado prático, não apenas em simulações cirúrgicas, mas também em outras áreas onde a realidade virtual e o feedback sensorial são relevantes. O dispositivo oferece uma plataforma versátil e adaptável que pode inspirar novas aplicações em educação, ciência e engenharia. Ao final, espera-se que esta pesquisa contribua para o avanço contínuo das tecnologias de simulação, promovendo uma experiência de aprendizagem mais prática e imersiva, ao mesmo tempo em que abre caminhos para o desenvolvimento de ferramentas mais sofisticadas no futuro.

Os testes realizados apontaram uma latência média de 26 milissegundos (ms), alcance BLE estável de até 2 metros e alta fidelidade entre os dados captados pelos sensores e os movimentos reproduzidos no ambiente virtual. A calibração dos sensores e a implementação de filtro de Kalman contribuíram para a estabilidade e precisão dos dados, enquanto os padrões de vibração desenvolvidos permitiram diferenciar as camadas anatômicas simuladas com eficiência.

Como diferencial em relação ao estado da arte, o sistema proposto apresenta mobilidade, compatibilidade com plataformas comerciais de VR e uso de tecnologias acessíveis, além de explorar a resposta tátil de maneira interativa e em tempo real. A lacuna identificada na literatura quanto ao uso de bisturis hápticos em RV foi abordada de forma inovadora e funcional.

Conclui-se, portanto, que o Bisturi VR representa uma contribuição significativa ao campo de simulação médica em realidade virtual, reunindo aspectos técnicos, pedagógicos e científicos em uma solução prática e promissora.

8 SUGESTÕES DE TRABALHOS FUTUROS

Implementar algoritmos de adaptação dinâmica da vibração, baseados em profundidade do corte ou resistência do tecido virtual;

Integrar sistema de métricas para análise de desempenho dos usuários após os treinamentos e comparativo;

Realizar pesquisa com usuários (alunos de medicina, residentes e cirurgiões) a fim de validar a eficácia pedagógica da solução proposta;

Ampliar o sistema para outros instrumentos cirúrgicos (pinças, tesouras, afastadores) com suporte a múltiplos dispositivos BLE simultâneos;

Explorar o uso do dispositivo em aplicações remotas de telessaúde ou treinamentos gamificados, ampliando o alcance da ferramenta.

9 PUBLICAÇÕES REALIZADAS

Durante o Programa de Pós-Graduação de Engenharia Elétrica – PPGE do ICTS/UNESP, foram publicados artigos e apresentados artigos correlacionados ao projeto desenvolvido.

DE SOUZA, V. F.; MORAES, F. S.; AZEVEDO, R. F.; DINIZ, I. S. **BISTURI CIRÚRGICO HÁPTICO COM COMUNICAÇÃO BLE INTEGRADO A AMBIENTE DE REALIDADE VIRTUAL IMERSIVA**. ARACÊ , [S. l.], v. 7, n. 5, p. 23193–23212, Maio 2025. DOI: 10.56238/arev7n5-135. [Revista]

DE SOUZA, V. F.; MORAES, F. S.; AZEVEDO, R. F.; DINIZ, I. S. **BISTURI CIRÚRGICO HÁPTICO COM COMUNICAÇÃO BLE INTEGRADO A AMBIENTE DE REALIDADE VIRTUAL IMERSIVA**. Health Innovations: Transforming Lives Through Science, ISBN: 978-65-6109-155-8. Maio, 2025. DOI: <https://doi.org/10.56238/sevened2025.013-006>. [Livro]

DE SOUZA, V. F.; DINIZ, I. S. **DESENVOLVIMENTO DE UM BISTURI PARA AMBIENTES DE REALIDADE VIRTUAL UTILIZANDO O META QUEST 2 E O SEEED XIAO NRF52840**. V Congresso Brasileiro Interdisciplinar em Ciência e Tecnologia - “Edição Especial: 5 anos da integração da ciência e tecnologia”. Agosto, 2024. [Resumo Simples]

DE SOUZA, V. F.; DINIZ, I. S. **DESENVOLVIMENTO DE UM BISTURI APLICADO A CIRURGIAS MÉDICAS COM REALIDADE VIRTUAL IMERSIVA: HAPITC E QUEST 2**. IX Workshop Integração de Saberes Ambientais: Cidades Inteligentes e Sustentáveis. Outubro, 2024. [Resumo estendido]

REFERÊNCIAS

- ABINAYA, P., AND MANIVANNAN, M. 2023. Haptic Based Fundamentals of Laparoscopic Surgery Simulation for Training with Objective Assessments. **Frontiers in Robotics and AI**, v. 11, Maio 2024. <<https://doi.org/10.3389/frobt.2024.1363952>>. Acesso em 24 ago. 2024.
- ANALOG DEVICES. **ADXL335 Datasheet**. Analog Devices, 2022. Disponível em: <<https://www.analog.com/media/en/technical-documentation/data-sheets/ADXL335.pdf>>. Acesso em 05 ago. 2024.
- ANALOG DEVICES. **ADXL335: Small, Low Power, 3-Axis ± 3 g Accelerometer**. Analog Devices, 2021. Disponível em: <https://www.analog.com/en/products/adxl335.html>. Acesso em 10 ago. 2024.
- BARSON, E. Z.; GRAAFLAND, M.; SCHIJVEN, M. P. Systematic Review on the Effectiveness of Augmented Reality Applications in Medical Training. **Surgical Endoscopy and Other Interventional Techniques**, v. 30, pp. 4174–4183, Fev. 2016. Disponível em: <<https://doi.org/10.1007/s00464-016-4800-6>>. Acesso em 21 set. 2024.
- BEKTIC, M. et al. Efficacy of AR Haptic Simulation for Nursing Student Education. In: **4TH INTERNATIONAL CONFERENCE ON BIO-ENGINEERING FOR SMART TECHNOLOGIES (BIOSMART)**, Paris, 2021. Paris: BioSmart, 2021, pp. 1-5. DOI: 10.1109/BioSMART54244.2021.9677828. Disponível em: <<https://ieeexplore.ieee.org/document/9677828/>>. Acesso em: 17 dez. 2024.
- BESHARATI, L.; MAHVASH, M. Augmented Reality-Guided Neurosurgery: Accuracy and Intraoperative Application of an Image Projection Technique. **Journal of Neurosurgery**, v. 123, n. 1, pp. 206–211. Disponível em: <<https://doi.org/10.3171/2014.9.JNS141001>>. Acesso em 23 set. 2024.
- BLUETOOTH. **The Bluetooth® Low Energy Primer**. Bluetooth, 2023. Disponível em: www.bluetooth.com. Acesso em 10 set. 2024.
- BRIGHAM, T. J. Verificação da realidade: noções básicas de realidade virtual aumentada e realidade mista. **Medical Reference Services Quarterly**, v. 36, n. 2, pp. 171-178, Abr. 2017. Disponível em: <<https://doi.org/10.1080/02763869.2017.1293987>>. Acesso em 15 jul. 2024.
- BRUCKHEIMER, E. et al. Computer-Generated Real-Time Digital Holography: First Time Use in Clinical Medical Imaging. **European Heart Journal: Cardiovascular Imaging**, v. 17, n. 8, pp. 845–849, Aug. 2016. Disponível em: <<https://doi.org/10.1093/ehjci/jew087>>. Acesso em 23 set. 2024.
- CERVANTES, D. **Game Development with Unity: An In-Depth Guide to Creating Interactive Experiences**. Birmingham: Packt Publishing, 2021.
- CHAABRA, S.; BAWEJA, H. Applications of Augmented Reality, Virtual Reality and Mixed Reality in Urological Interventions: A Systematic Review. **International Journal for Research in Applied Science & Engineering Technology (IJRASET)**,

v. 21, n. 11, pp. 448-459, Nov. 2023. Disponível em:
<<https://doi.org/10.22214/ijraset.2023.56524>>. Acesso em 03 jul. 2024.

CHATZOPOULOS, D. et al. Mobile Augmented Reality Survey: From Where We Are to Where We Go. **IEEE Access**, v. 5, pp. 6917-6950, Abr. 2017. Disponível em:
<<https://doi.org/https://doi.org/10.1109/ACCESS.2017.2698164>>. Acesso em 17 set. 2024.

DAI, K.; GARCIA, J.; OLAVE-ENCINA, K. In-Between Worlds: Chilean University Lecturers' Experiences of Teaching Transition Between Face-to-Face and Virtual Reality Contexts During the COVID-19 Pandemic. **Education technology research and development**, v. 71, pp. 1851–1867, Abr. 2023. Disponível em:
<<https://doi.org/10.1007/s11423-023-10217-0>>. Acesso em 03 jul. 2024.

DICKEY, R. M. et al. Augmented Reality Assisted Surgery: a Urologic Training Tool. **Asian journal of andrology**, v. 18, n. 5, pp. 732–734, Set-Out. 2016. Disponível em:
<<https://doi.org/10.4103/1008-682X.166436>>. Acesso em 16 set. 2024.

DONG, H. et al. Development of a Web-Based Haptic Authoring Tool for Multimedia Applications. In: **IEEE INTERNATIONAL SYMPOSIUM ON MULTIMEDIA (ISM)**, Miami, 2015. Miami: IEEE Xplorer, 2015, pp. 13-26. DOI: 10.1109/ISM.2015.71. Disponível em: <<https://ieeexplore.ieee.org/document/7442269>>. Acesso em: 04 out. 2024.

DUARTE, M. L. et al. Learning Anatomy by Virtual Reality and Augmented Reality: A Scope Review. **Morphologie: bulletin de l'Association des anatomistes**, v. 104, n. 347, pp. 254–266, Dez. 2020. Disponível em:
<<https://doi.org/10.1016/j.morpho.2020.08.004>>. Acesso em 17 set. 2024.

ECKERT M.; VOLMERG J.; FRIEDRICH C. Augmented Reality in Medicine: Systematic and Bibliographic Review. **JMIR Mhealth Uhealth**, v. 7, n. 4, e10967, Abr. 2019. Disponível em: <<https://mhealth.jmir.org/2019/4/e10967>>. Acesso em 02 jul. 2024.

FERRONE, H. **Learning C# by Developing Games with Unity 2020**. Birmingham: Packt Publishing, 2020.

FIALHO, A. B. **Realidade Virtual e Aumentada Tecnologias para Aplicações Profissionais**. São Luís: Editora Érica, 2018.

FIDA, B. et al. Augmented reality in open surgery. **Updates in Surgery**, v. 70, n. 3, pp. 389–400, Set. 2018. Disponível em: <<https://doi.org/10.1007/s13304-018-0567-8>>. Acesso em 17 set. 2024.

FONNET, A.; PRIE, Y. Survey of Immersive Analytics. **IEEE Transactions on Visualization and Computer Graphics**, [S.l.], vol. 27, n. 3, pp. 2101-2122, Mar. 2021. IEEE Xplorer: [S.l.], 2021. Disponível em:
<<https://doi.org/10.1109/TVCG.2019.2929033>>. Acesso em: 10 nov. 2024.

FORONDA, C. et al. Virtually Nursing: Emerging Technologies in Nursing Education. **Nurse Educator**, v. 42, n. 1, pp. 14–17, Jan. 2017. Disponível em:
<<https://doi.org/10.1097/NNE.0000000000000295>>. Acesso em 23 ago. 2024.

FRAGA-LAMAS, P. et al. A Review on Industrial Augmented Reality Systems for the Industry 4.0 Shipyard. **IEEE Access**, v. 6, pp. 13358-13375, Fev. 2018. Disponível em: <<https://doi.org/10.1109/ACCESS.2018.2808326>>. Acesso em 14 out. 2024.

GERUP, J.; SOERENSEN, C. B.; DIECKMANN, P. Augmented reality and mixed reality for healthcare education beyond surgery: an integrative review. **International Journal of Medical Education**, v. 11, pp. 1–18, Jan. 2020. Disponível em: <<https://doi.org/10.5116/ijme.5e01.eb1a>>. Acesso em 11 nov. 2024.

GOLDSTONE, W. **Unity Game Development Essentials: Build Fully Functional, Professional 3d Games With Realistic Environments, Sound, Dynamic Effects, and More!** Birmingham: Packt Publishing, 2009.

GONZÁLEZ, I. S. et al. Virtual Reality In Higher Education: An Experience With Medical Students: Research into how Virtual Reality can be used as a powerful training tool for medicine students. In: **INEIGHT INTERNATIONAL CONFERENCE ON TECHNOLOGICAL ECOSYSTEMS FOR ENHANCING MULTICULTURALITY**, [S. I.], 2021. [S.I.]: ACM DL Library, 2021, pp. 414-421, DOI: 10.1145/3434780.3436539. Disponível em: <<https://dl.acm.org/doi/abs/10.1145/3434780.3436539>>. Acesso em 15 jan. 2025.

HERRON, J. Augmented Reality in Medical Education and Training. **Journal of Electronic Resources in Medical Libraries**, v. 3, n. 2, pp. 51-55, Maio 2016. Disponível em: <<https://doi.org/10.1080/15424065.2016.1175987>>. Acesso em 16 set. 2024.

HUANG, S et al. Review of studies on target acquisition in virtual reality based on the crossing paradigma. **Virtual Reality & Intelligent Hardware**, v. 1, pp. 251-264, Jun 2019. Disponível em: <<https://doi.org/10.3724/SP.J.2096-5796.2019.0006>>. Acesso em 04 jul. 2024.

INVENSENSE. **MPU-6050 Product Specification**. InvenSense, 2023. Disponível em: <<https://invensense.tdk.com/products/motion-tracking/6-axis/mpu-6050/>>. Acesso em 05 ago. 2024.

JODA, T. et al. Augmented and virtual reality in dental medicine: A systematic review. **Computers in Biology and Medicine**, v. 108, pp. 93–100, Maio 2019. Disponível em: <<https://doi.org/10.1016/j.compbiomed.2019.03.012>>. Acesso em 08 set. 2024.

KAMPHUIS, C. et al. Augmented reality in medical education. **Perspectives on medical education**, v. 3, n. 4, pp. 300–311, Sep. 2014. Disponível em: <<https://doi.org/10.1007/s40037-013-0107-7>>. Acesso em 16 set. 2024.

KHOR, W. S. Augmented and virtual reality in surgery in the digital surgical environment: applications, limitations and legal pitfalls. **Annals of Translational Medicine**, v. 4, n. 23, pp. 454, Dec. 2016. Disponível em: <<https://doi.org/10.21037/atm.2016.12.2310.21037/atm.2016.12.23>>. Acesso em 08 set. 2024.

KYAW, B. M. et al. VR for health professions education: systematic review and meta-analysis by the Digital Health Education Collaboration. **J Med Internet Res**, v. 21, n.

1, e12959, Jan. 2019. Disponível em: <<https://doi.org/10.2196/12959>>. Acesso em 04 jul. 2024.

LI, A. et al. Haptic Signal Reconstruction in eHealth Internet of Things. **IEEE Internet of Things Journal**, [S. l.], v. 9, n. 18, pp. 17047-17057, Set. 2022. Disponível em: <<https://doi.org/10.1109/JIOT.2021.3132771>>. Acesso em: 04 out. 2024.

LIU, P. et al. A Wearable Augmented Reality Navigation System for Surgical Telementoring Based on Microsoft HoloLens. **Annals of Biomedical Engineering**, v. 49, pp. 287–298, Jan. 2021. Disponível em: <<https://doi.org/10.1007/s10439-020-02538-5>>. Acesso em 07 set. 2024.

LUDLOW, B. L. Virtual Reality: Emerging Applications and Future Directions. **Rural Special Education Quarterly**, v. 34, n. 3, pp. 3-10, Set. 2015. Disponível em: <<https://doi.org/10.1177/875687051503400302>>. Acesso em 02 jul. 2024.

MA, J. et al. VUME: The voluntary-use methodology for evaluations. In: **IEEE SYMPOSIUM ON 3D USER INTERFACES (3DUI)**, Greenville, 2016. Greenville: IEEE Xplorer, 2016, pp. 129-132. DOI: 10.1109/3DUI.2016.7460042. Disponível em: <<https://ieeexplore.ieee.org/document/7460042>>. Acesso em: 10 nov. 2024.

MA, M. et al. Personalized Augmented Reality for Anatomy Education. **Clinical Anatomy**, v. 29, n. 4, pp. 446–453, Maio 2016. Disponível em: <<https://doi.org/10.1002/ca.22675>>. Acesso em 19 set. 2024.

MARKS, B.; THOMAS, J. Adoption of virtual reality technology in higher education: An evaluation of five teaching semesters in a purpose-designed laboratory. **Education and Information Technologies**, v. 27, n. 1, pp.1287–1305, Jul. 2021. Disponível em: <<https://doi.org/10.1007/s10639-021-10653-6>>. Acesso em 03 jul. 2024.

MEOLA, A. et al. Augmented reality in neurosurgery: a systematic review. **Neurosurgical Review**, v. 40, pp. 537–548, Out. 2017. Disponível em: <<https://doi.org/10.1007/s10143-016-0732-9>>. Acesso em 17 set. 2024.

META. **Oculus Quest 2: Advanced All-In-One Virtual Reality Headset**. Meta, 2020. Disponível em: <https://www.meta.com/quest/>. Acesso em 02 jul. 2024.

META. **Oculus Quest 2: Our Most Advanced New All-in-One VR Headset**. Meta, 2019. Disponível em: <<https://www.meta.com/quest/products/quest-2.>>. Acesso em 02 jul. 2024.

META. **Unlock Your PC Power with Oculus Link**. Meta, 2021. Disponível em: <https://www.meta.com/quest/setup/oculus-link/>. Acesso em 02 jul. 2024.

MICROSOFT. **O que é realidade misturada?** Microsoft, 2023. Disponível em: <<https://learn.microsoft.com/pt-br/windows/mixed-reality/discover/mixed-reality>>. Acesso em 03 jul. 2024.

MIRZA, T.; TULI, N.; MANTRI, A. Aplicações de Realidade Virtual, Realidade Aumentada e Realidade Mista: Cenário Atual. In: **2ª CONFERÊNCIA INTERNACIONAL SOBRE COMPUTAÇÃO AVANÇADA E TECNOLOGIAS INOVADORAS EM ENGENHARIA (ICACITE)**, Greater Noida, 2022. Greater Noida:

IEEE Xplore, 2022, pp. 1405-1412. Disponível em: <<https://doi.org/10.1109/ICACITE53722.2022.9823482>>. Acesso em 03 jul. 2024.

MORO, C. et al. The Effectiveness of Virtual and Augmented Reality in Health Sciences and Medical Anatomy. **Anatomical Sciences Education**, v. 10, n. 6, pp. 549-559, Dez. 2017. Disponível em: <<https://doi.org/10.1002/ase.1696>>. Acesso em 14 nov. 2024.

MOSTAFA, A. E.; SHARLIN, E.; SOUSA, M. C. Superhumans: A 3DUI design metaphor. In: **IEEE SYMPOSIUM ON 3D USER INTERFACES (3DUI)**, Minneapolis, 2014. IEEE Xplorer: Minneapolis, 2024, pp. 143-144. DOI: 10.1109/3DUI.2014.6798861. Disponível em: <<https://ieeexplore.ieee.org/document/6798861>>. Acesso em: 03 set. 2024.

MUNZER, B. W. et al. Augmented Reality in Emergency Medicine: A Scoping Review. **Journal of Medical Internet Research**, v. 21, n. 4, e12368, Abr. 2019. Disponível em: <<https://doi.org/10.2196/12368>>. Acesso em 15 set. 2024.

NEGRILLO-CÁRDENAS, J.; JIMÉNEZ-PÉREZ, J-R.; FEITO, F. The Role of Virtual and Augmented Reality in Orthopedic Trauma Surgery: From Diagnosis to Rehabilitation. **Computer Methods and Programs in Biomedicine**, v. 191, 105407, Fev. 2020. Disponível em: <<https://doi.org/10.1016/j.cmpb.2020.105407>>. Acesso em 11 set. 2024.

NIANTIC. **Pokemon Go AR+**. NIANTIC, 2017. Disponível em: <<https://pokemongolive.com/post/arplus/>>. Acesso em 02 jul. 2024.

NORDIC, 2024. **XIAO BLE Sense**. Nordic, 2024. Disponível em: https://docs.nordicsemi.com/bundle/ncs-2.5.2/page/zephyr/boards/arm/xiao_ble/doc/index.html. Acesso em 02 jul. 2024.

NOURY, P et al. A Narrative Review of the Current State of Extended Reality Technology and How it can be Utilised in Sport. **Sports Med**, v. 52, pp. 1473–1489, Mar. 2022. Disponível em: <<https://doi.org/10.1007/s40279-022-01669-0>>. Acesso em 02 jul. 2024.

ONG, C. Applications of Extended Reality in Ophthalmology: Systematic Review. **Journal of medical Internet research**, v. 23, n. 8, e24152, Aug. 2021. Disponível em: <<https://doi.org/10.2196/24152>>. Acesso em 02 jul. 2024.

PALMAS, F.; KLINKER, G. Defining Extended Reality Training: A Long-Term Definition for All Industries. In: **IEEE 20TH INTERNATIONAL CONFERENCE ON ADVANCED LEARNING TECHNOLOGIES (ICALT)**, Tartu, 2020. Tartu: IEEE Xplorer, 2020, pp. 322-324, DOI: 10.1109/ICALT49669.2020.00103. Disponível em: <<https://ieeexplore.ieee.org/abstract/document/9155942>>. Acesso em 02 jul. 2024.

PANG, C.; CAI, Y. Transforming Learning Experiences Through Affordances of Virtual and Augmented Reality. In: CAI, Y., MANGINA, E., GOEI, S.L. (EDS). **Mixed Reality for Education: Gaming Media and Social Effects**. Singapura: Springer, 2023.

PONS, P.; NAVAS-MEDRANO, S.; SOLER-DOMINGUEZ, J.L. Realidade Estendida para Saúde Mental: Tendências Atuais e Desafios Futuros. **Front. Comput. Sci.**, v. 4, 1034307, Nov. 2022. Disponível em: <<https://doi.org/10.3389/fcomp.2022.1034307>>. Acesso em 02 jul. 2024.

QODENEXT. **A Complete Guide to What is BLE (Bluetooth Low Energy)**. QODENEXT, 2023. Disponível em: www.qodenext.com. Acesso em 10 set. 2024.

ROMAGNOLI, L.; MATTIA, G. P.; BERARDI, R. A Study on Energy Efficiency in Edge-assisted VR Applications with Meta Quest 2 for Disaster Management. In: **INTERNATIONAL CONFERENCE ON INFORMATION AND COMMUNICATION TECHNOLOGIES FOR DISASTER MANAGEMENT (ICT-DM)**. Cosenza: IEEE Xplorer, 2023, pp. 1-7. Disponível em: <<https://doi.org/10.1109/ICT-DM58371.2023.10286920>>. Acesso em 10 set. 2024.

ROO, S.; HACHET, M. Uma Realidade: aumentando a maneira como o mundo físico é vivenciado pela combinação de múltiplas modalidades de realidade mista. In: UIST 2017, Quebec. **Anais do 30º Simpósio Anual da ACM sobre Software e Tecnologia de Interface de Usuário**. Quebec, 2017, pp. 787-795. Disponível em: <<https://doi.org/10.1145/3126594.3126638>>. Acesso em 03 jul. 2024.

SANNA, A; MANURI, F.; PACE, F. Aplicações de realidade aumentada e mista vestíveis, 2019.

SEED Studio. **Introdução ao XIAO nRF52840 Sense**. SEED Studio, 2023. Disponível em: https://wiki.seeedstudio.com/XIAO_BLE/. Acesso em 02 jul. 2024.

SENSEGLOVE. **Nova geração de feedback tátil e força para aplicações XR**. SenseGlove, s.d. Disponível em: <<https://www.senseglove.com>>. Acesso em 24 ago. 2024.

STMICROELECTRONICS. **L3GD20 3-Axis Digital Gyroscope**. STMicroelectronics, 2023. Disponível em: <<https://www.st.com/resource/en/datasheet/l3gd20.pdf>>. Acesso em 05 ago. 2024.

STMICROELECTRONICS. **LSM6DS3TR-C: iNEMO inertial module: always-on 3D accelerometer and 3D gyroscope**. STMicroelectronics, 2023 Disponível em: <<https://www.st.com/en/mems-and-sensors/lsm6ds3tr-c.html>>. Acesso em 05 ago. 2024.

STORMOTION. **Bluetooth Low Energy: Full BLE FAQ**. Stormotion, 2024. Disponível em: www.stormotion.io. Acesso em 10 set. 2024.

SUH, K. S.; LEE, Y.E. The effects of VR on consumer learning: an empirical investigation. **Mis Quarterly**, v. 29, n. 4, pp. 673–697, Dez. 2025. Disponível em: <<https://doi.org/10.2307/25148705>>. Acesso em 17 nov. 2024.

SYBERFELDT, A.; DANIELSSON, O.; GUSTAVSSON, P. Augmented Reality Smart Glasses in the Smart Factory: Product Evaluation Guidelines and Review of Available Products. **IEEE Access**, v. 5, pp. 9118-9130, Maio 2017. Disponível em: <<https://doi.org/10.1109/access.2017.2703952>>. Acesso em 23 set. 2024.

TAI, Y. et al. Intelligent Intraoperative Haptic-AR Navigation for COVID-19 Lung Biopsy Using Deep Hybrid Model. **IEEE Transactions on Industrial Informatics**, v.

17, n. 9, p. 6519–6527, Set. 2021. Disponível em:
<<https://doi.org/10.1109/TII.2021.3052788>>. Acesso em: 10 nov. 2024.

TAKALA, T. M.; RAUHAMMA, P.; TAKALA, T. Survey of 3DUI applications and development challenges. In: **IEEE SYMPOSIUM ON 3D USER INTERFACES (3DUI)**, Costa Mesa, 2012. Costa Mesa: IEEE Xplorer, 2012, pp. 89-96. DOI: 10.1109/3DUI.2012.6184190. Disponível em:
<<https://ieeexplore.ieee.org/document/6184190>>. Acesso em: 04 out. 2024.

TALEB, T. VR-Based Immersive Service Management in B5G Mobile Systems: A UAV Command and Control Use Case. **IEEE Internet of Things Journal**, v. 10, n. 6, pp. 5349-5363, Mar. 2023. <<https://doi.org/10.1109/JIOT.2022.3222282>>.

TANG, K. S et al. Augmented reality in medical education: a systematic review. **Canadian Medical Education Journal**, v. 11, n. 1, pp. 81–pp. 96. Disponível em:
<<https://doi.org/10.36834/cmej.61705>>. Acesso em 16 set. 2024.

TEPPER, O. M. et al. Mixed reality with HoloLens: where virtual reality meets augmented reality in the operating room. **Plastic and reconstructive surgery**, v. 140, n. 5, pp. 1066-1070, Nov. 2017. Disponível em:
<<https://doi.org/10.1097/PRS.0000000000003802>>. Acesso em: 16 out. 2024.

TEPPER, O. M. et al. Mixed reality with HoloLens: where virtual reality meets augmented reality in the operating room. **Plastic and reconstructive surgery**, v. 140, n. 5, pp. 1066-1070, Nov. 2017. Disponível em:
<<https://doi.org/10.1097/PRS.0000000000003802>>. Acesso em: 16 out. 2024.

THOMPSON, A.; SMITH, J. Haptic Feedback in Virtual Reality Applications: A Comprehensive Overview. **Journal of Immersive Technology**, v. 12, n. 4, pp. 89-101.

TOKARCHUK, L. et al. Editorial: Remote XR user studies. **Front. Comput. Sci**, v. 5, 1144109, Feb. 2023. Disponível em: <<https://doi.org/10.3389/fcomp.2023.1144109>>. Acesso em 02 jul. 2024.

TU, H. et al. Crossing-based selection with virtual reality head-mounted displays. In: **PROCEEDINGS OF THE 2019 CHI CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS**. Nova Iorque: IEEE Xplorer, 2019, pp. 1-14. DOI: 10.1145/3290605.3300848. Disponível em:
<<https://dl.acm.org/doi/10.1145/3290605.3300848>>. Acesso em: 04 jul. 2024.

UNITY TECHNOLOGIES. **Unity Game Engine Overview**. Unity Technologies, 2023. Disponível em: <https://unity.com/>. Acesso em 03 jul. 2024.

UNITY TECHNOLOGIES. **XR Interaction Toolkit**. Unity Technologies, 2022. Disponível em:
<https://docs.unity3d.com/Packages/com.unity.xr.interaction.toolkit@2.0/manual/index.html>. Acesso em 03 jul. 2024.

UPPOT, R. N. et al. Implementing Virtual and Augmented Reality Tools for Radiology Education and Training, Communication, and Clinical Care. **Radiology**, v. 291, n. 3, pp. 570–580, Jun. 2019. Disponível em:
<<https://doi.org/10.1148/radiol.2019182210>>. Acesso em 17 set. 2024.

URUTHIRALINGAM, U.; REA, P. Augmented and Virtual Reality in Anatomical Education - A Systematic Review. Biomedical Visualisation. **Advances in Experimental Medicine and Biology**, v. 1235, pp. 89–101, 2020. Disponível em: <https://doi.org/10.1007/978-3-030-37639-0_5>. Acesso em 07 set. 2024.

VÁVRA, P. et al. Recent Development of Augmented Reality in Surgery: A Review. Journal of Healthcare Engineering. **Journal of healthcare engineering**, 4574172, Aug. 2017. Disponível em: <<https://doi.org/10.1155/2017/4574172>>. Acesso em 16 out. 2024.

VINCE, J. **Introduction to Virtual Reality**. Londres: Springer Science & Business Media, 2004.

WANG, X.; ONG, S. K.; NEE, A. Y. C. A comprehensive survey of augmented reality assembly research. **Advances in Manufacturing**, v. 4, pp. 1-22, Jan. 2016. Disponível em: <<https://doi.org/10.1007/s40436-015-0131-4>>. Acesso em 23 ago. 2024.

WONG, C. L.; Effects of immersive virtual reality for preventing and managing anxiety, nausea and vomiting among paediatric cancer patients receiving their first chemotherapy: a study protocol for an exploratory trial. **PloS One**, v. 16, n. 10, e0258514, Out. 2021. Disponível em: <<https://doi.org/10.1371/journal.pone.0258514>>. Acesso em 03 jul. 2024.

WONG, K. et al. **Applications of Augmented Reality in Otolaryngology: A Systematic Review**. Otolaryngology-Head and Neck Surgery American Academy of Otolaryngology-Head and Neck Surgery, v. 159, n. 6, pp. 956–967, Aug. 2018. Disponível em: <<https://doi.org/10.1177/0194599818796476>>. Acesso em 14 dez. 2025.

XIE, B. et al. A review on virtual reality skill training applications. **Frontiers in Virtual Reality**, v. 2, 645153, Abr. 2021. Disponível em: <<https://doi.org/10.3389/frvir.2021.645153>>. Acesso em: 16 out. 2024.

XU, X et al. Mixed Reality Applications in Tertiary Veterinary Education: A Systematic Review. In: CAI, Y; MANGINA, E.; GOEI, S.L. (eds). **Mixed Reality for Education. Gaming Media and Social Effects**. Singapura: Springer, 2023.

YAMAKAWA, M.; SUNG, H.C.; TUNG PUNKOM, P. 2020. Virtual reality education for dementia care: a scoping review protocol. **JBIS Evidence Synthesis**, v. 18, n. 9, pp. 2075-2081, Set. 2020. Disponível em: <<https://doi.org/10.11124/JBISRIR-D-19-00230>>. Acesso em 03 jul. 2024.

YOUNGJUN, K.; HANNAH K.; YONG O. Virtual Reality and Augmented Reality in Plastic Surgery: A Review. **Archives of Plastic Surgery**, v. 44, n. 3, pp. 179–187, Maio 2017. Disponível em: <<https://doi.org/10.5999/aps.2017.44.3.179>>. Acesso em 23 ago. 2024.

ZHU, E. et al. Augmented reality in healthcare education: an integrative review. **PeerJ**, v. 2, e469, Jul. 2017. Disponível em: <<https://doi.org/10.7717/peerj.469>>. Acesso em 29 nov. 2024.

APENDICE A — FIRMWARE

```

nRF52840_Sense_Gamepad.ino  main.h
1  #include <bluefruit.h>
2  #include <Adafruit_TinyUSB.h>
3  #include <Adafruit_Sensor.h>
4  #include <Wire.h>
5  #include <LSM6DS3.h>
6
7  #define X 0
8  #define Y 1
9  #define Z 2
10 #define VEL_PIN 3
11 #define VIBRATION_1 5
12 #define VIBRATION_2 6
13 #define FOURTH_BUTTON 7
14 #define THIRD_BUTTON 8
15 #define SECOND_BUTTON 9
16 #define FIRST_BUTTON 10
17 #define VIN 3.3
18 #define MID_VIN 1.65
19 #define ZERO 0
20 #define ONE 1
21 #define DECIMAL 2
22 #define INPS_ARRAY 8
23 #define MIN_VALUE 10
24 #define MIN_ACC_THRESHOLD 1.5
25 #define THRESHOLD 5
26 #define RESOLUTION 1024.0
27 #define TIME_SLEEP 300000UL
28
29 unsigned long lastMotionMillis = 0, motorStartTime = 0;
30 int motorDuration = 0, pwmValue = 0, velocity = 1;
31 bool motorActive = false;
32 float previousAccX = 0, previousAccY = 0, previousAccZ = 0;
33 float previousGyroX = 0, previousGyroY = 0, previousGyroZ = 0;
34 float filterAlpha = 0.1;
35 float biasGyroX = 0, biasGyroY = 0, biasGyroZ = 0;
36
37 const uint8_t pinInps[INPS_ARRAY] = {X, Y, Z, VEL_PIN, FOURTH_BUTTON, THIRD_BUTTON, SECOND_BUTTON, FIRST_BUTTON};
38 float accX, accY, accZ, gyroX, gyroY, gyroZ;
39 const uint8_t sampleSize = 20;
40 const int RawMin = 0, RawMax = 1023, numSamples = 200;
41 const float biasAccX = 0.51, biasAccY = -0.37, biasAccZ = -0.53, Volts = 0.33;
42
43 void startAdv(void);
44 float ReadAxis(int axisPin, float bias);
45 long int CheckButtons();
46 void parseCommand(String command);
47 void int1ISR();

```

nRF52840_Sense_Gamepad.ino

main.h

```

1  #include "main.h"
2
3  BLEDis bledis;
4  BLEHidGamepad blegamepad;
5  BLEUart bleuart;
6  LSM6DS3 internal_IMU(I2C_MODE, 0x6A);
7
8  hid_gamepad_report_t gp;
9
10 void setup() {
11     Serial.begin(115200);
12     for(int i = ZERO; i < INPS_ARRAY; i++) pinMode(pinInps[i], INPUT);
13     delay(10);
14     Serial.println();
15     analogReference(AR_INTERNAL_3_0);
16
17     detachInterrupt(digitalPinToInterrupt(FIRST_BUTTON));
18     detachInterrupt(digitalPinToInterrupt(SECOND_BUTTON));
19     detachInterrupt(digitalPinToInterrupt(THIRD_BUTTON));
20     detachInterrupt(digitalPinToInterrupt(FOURTH_BUTTON));
21
22     if (!internal_IMU.begin()) Serial.println("Failed to find LSM6DS3 chip");
23
24     Bluefruit.begin();
25
26     char name[20] = {"Bisturi-"};
27     char macStr[7];
28     uint8_t macAddress[6];
29     Bluefruit.getAddr(macAddress);
30     snprintf(macStr, sizeof(macStr), "%02X%02X%02X", macAddress[3], macAddress[4], macAddress[5]);
31     strcat(name, macStr);
32     delay(100);
33
34     Bluefruit.setTxPower(4);
35     Bluefruit.setName(name);
36
37     bleuart.begin();
38
39     bledis.setManufacturer("Adafruit Industries");
40     bledis.setModel("Bluefruit Feather 52");
41     bledis.begin();
42     blegamepad.begin();
43     startAdv();
44
45     calibrateGyroscope();
46
47     lastMotionMillis = millis();
48 }
49

```

nRF52840_Sense_Gamepad.ino main.h

```

50 void startAdv(void) {
51     Bluefruit.Advertising.addFlags(BLE_GAP_ADV_FLAGS_LE_ONLY_GENERAL_DISC_MODE);
52     Bluefruit.Advertising.addTxPower();
53     Bluefruit.Advertising.addAppearance(BLE_APPEARANCE_HID_GAMEPAD);
54     Bluefruit.Advertising.addService(blegamepad);
55     Bluefruit.Advertising.addName();
56     Bluefruit.Advertising.restartOnDisconnect(true);
57     Bluefruit.Advertising.setInterval(32, 244); // in unit of 0.625 ms
58     Bluefruit.Advertising.setFastTimeout(30); // number of seconds in fast mode
59     Bluefruit.Advertising.start(ZERO);
60 }
61
62 void loop(void) {
63     if (Bluefruit.connected()) {
64         long int pressedButton = CheckButtons();
65         pressedButton >= ZERO ? gp.buttons = pressedButton : gp.buttons = 0;
66
67         float acc_threshold = MIN_ACC_THRESHOLD * velocity;
68
69         if (fabs(accX - previousAccX) > acc_threshold || fabs(accY - previousAccY) > acc_threshold || fabs(accZ - previousAccZ) > acc_threshold ||
70             fabs(gyroX - previousGyroX) > THRESHOLD || fabs(gyroY - previousGyroY) > THRESHOLD || fabs(gyroZ - previousGyroZ) > THRESHOLD) {
71             lastMotionMillis = millis();
72         }
73
74         velocity = analogRead(VEL_PIN) < MIN_VALUE ? ONE : analogRead(VEL_PIN) / MIN_VALUE;
75
76         accX = ReadAxis(X, biasAccX);
77         accX *= velocity;
78         accY = ReadAxis(Y, biasAccY);
79         accY *= velocity;
80         accZ = ReadAxis(Z, biasAccZ);
81         accZ *= velocity;
82
83         gyroX = internal_IMU.readFloatGyroX() - biasGyroX;
84         gyroY = internal_IMU.readFloatGyroY() - biasGyroY;
85         gyroZ = internal_IMU.readFloatGyroZ() - biasGyroZ;
86
87         if(motorActive) {
88             accX = (filterAlpha * accX) + ((1 - filterAlpha) * previousAccX);
89             accY = (filterAlpha * accY) + ((1 - filterAlpha) * previousAccY);
90             accZ = (filterAlpha * accZ) + ((1 - filterAlpha) * previousAccZ);
91             gyroX = (filterAlpha * gyroX) + ((1 - filterAlpha) * previousGyroX);
92             gyroY = (filterAlpha * gyroY) + ((1 - filterAlpha) * previousGyroY);
93             gyroZ = (filterAlpha * gyroZ) + ((1 - filterAlpha) * previousGyroZ);
94         }
95
96         previousAccX = accX; previousAccY = accY; previousAccZ = accZ;
97         previousGyroX = gyroX; previousGyroY = gyroY; previousGyroZ = gyroZ;
98
99         gp.x = accX; gp.y = accY; gp.z = accZ;
100         gp.rx = gyroX; gp.ry = gyroY; gp.rz = gyroZ;
101         blegamepad.report(&gp);
102
103         if (bleuart.available()) {
104             String command = bleuart.readString();
105             parseCommand(command);

```

```

nRF52840_Sense_Gamepad.ino  main.h
103     if (bleuart.available()) {
104         String command = bleuart.readString();
105         parseCommand(command);
106     }
107
108     if (millis() - lastMotionMillis > TIME_SLEEP) {
109         attachInterrupt(digitalPinToInterrupt(FIRST_BUTTON), int1ISR, RISING);
110         attachInterrupt(digitalPinToInterrupt(SECOND_BUTTON), int1ISR, RISING);
111         attachInterrupt(digitalPinToInterrupt(THIRD_BUTTON), int1ISR, RISING);
112         attachInterrupt(digitalPinToInterrupt(FOURTH_BUTTON), int1ISR, RISING);
113
114         nrf_gpio_cfg_sense_input(digitalPinToInterrupt(FIRST_BUTTON), NRF_GPIO_PIN_PULLDOWN, NRF_GPIO_PIN_SENSE_HIGH);
115         nrf_gpio_cfg_sense_input(digitalPinToInterrupt(SECOND_BUTTON), NRF_GPIO_PIN_PULLDOWN, NRF_GPIO_PIN_SENSE_HIGH);
116         nrf_gpio_cfg_sense_input(digitalPinToInterrupt(THIRD_BUTTON), NRF_GPIO_PIN_PULLDOWN, NRF_GPIO_PIN_SENSE_HIGH);
117         nrf_gpio_cfg_sense_input(digitalPinToInterrupt(FOURTH_BUTTON), NRF_GPIO_PIN_PULLDOWN, NRF_GPIO_PIN_SENSE_HIGH);
118         Bluefruit.setConnLedInterval(0);
119
120         NRF_POWER->SYSTEMOFF = 1;
121     }
122 }
123
124 if (motorActive) {
125     if (millis() - motorStartTime >= motorDuration) {
126         analogWrite(VIBRATION_2, ZERO);
127         motorActive = false;
128     }
129 }
130 }
131
132 float ReadAxis(int axisPin, float bias) {
133     int reading = analogRead(axisPin);
134     float voltage = (reading / RESOLUTION) * VIN;
135     float acceleration = (voltage - MID_VIN) / Volts;
136     return acceleration - bias;
137 }
138
139 long int CheckButtons() {
140     bool firstButtonPressed = digitalRead(FIRST_BUTTON);
141     bool secondButtonPressed = digitalRead(SECOND_BUTTON);
142     bool thirdButtonPressed = digitalRead(THIRD_BUTTON);
143     bool fourthButtonPressed = digitalRead(FOURTH_BUTTON);
144
145     int buttonState = (firstButtonPressed << 3) | (secondButtonPressed << 2) | (thirdButtonPressed << 1) | fourthButtonPressed;
146

```

```

nRF52840_Sense_Gamepad.ino  main.h

147     switch (buttonState) {
148         case 0b1111:
149             return GAMEPAD_BUTTON_THUMBL;
150         case 0b1101:
151             return GAMEPAD_BUTTON_TR2;
152         case 0b1110:
153             return GAMEPAD_BUTTON_TL2;
154         case 0b1011:
155             return GAMEPAD_BUTTON_TR;
156         case 0b1001:
157             return GAMEPAD_BUTTON_TL;
158         case 0b1100:
159             return GAMEPAD_BUTTON_Z;
160         case 0b1010:
161             return GAMEPAD_BUTTON_Y;
162         case 0b0111:
163             return GAMEPAD_BUTTON_THUMBR;
164         case 0b0101:
165             return GAMEPAD_BUTTON_MODE;
166         case 0b0110:
167             return GAMEPAD_BUTTON_START;
168         case 0b0011:
169             return GAMEPAD_BUTTON_SELECT;
170         case 0b0001:
171             return GAMEPAD_BUTTON_A;
172         case 0b0010:
173             return GAMEPAD_BUTTON_B;
174         case 0b0100:
175             return GAMEPAD_BUTTON_C;
176         case 0b1000:
177             return GAMEPAD_BUTTON_X;
178         default:
179             return -1;
180     }
181 }
182
183 void parseCommand(String command) {
184     int duration = 0;
185     int pwm = 0;
186
187     int tIndex = command.indexOf('T');
188     int pIndex = command.indexOf('P');
189
190     if (tIndex != -1 && pIndex != -1) {
191         duration = command.substring(tIndex + 1, pIndex).toInt(); // Tempo
192         pwm = command.substring(pIndex + 1).toInt(); // Valor PWM
193
194         if (pwm >= 0 && pwm <= 255) {
195             motorDuration = duration;
196             pwmValue = pwm;
197             motorActive = true;
198             motorStartTime = millis(); // Marca o início do tempo do motor
199             analogWrite(VIBRATION_2, pwmValue); // Aciona o motor com o PWM
200         }
201     }
202 }

```

```
nRF52840_Sense_Gamepad.ino  main.h
203
204 void calibrateGyroscope() {
205     float gyroXSum = 0, gyroYSum = 0, gyroZSum = 0;
206
207     for (int i = 0; i < numSamples; i++) {
208         gyroXSum += internal_IMU.readFloatGyroX();
209         gyroYSum += internal_IMU.readFloatGyroY();
210         gyroZSum += internal_IMU.readFloatGyroZ();
211
212         delay(50); // Pequena pausa para estabilizar as amostras
213     }
214
215     biasGyroX = gyroXSum / numSamples;
216     biasGyroY = gyroYSum / numSamples;
217     biasGyroZ = gyroZSum / numSamples;
218 }
219
220 void int1ISR() {
221     detachInterrupt(digitalPinToInterrupt(FIRST_BUTTON));
222     detachInterrupt(digitalPinToInterrupt(SECOND_BUTTON));
223     detachInterrupt(digitalPinToInterrupt(THIRD_BUTTON));
224     detachInterrupt(digitalPinToInterrupt(FOURTH_BUTTON));
225 }
```