



**UNIVERSIDADE FEDERAL DO CEARÁ**

**INSTITUTO DE ARQUITETURA , URBANISMO E DESIGN**

**CURSO BACHARELADO EM DESIGN**

**GABRIEL EDUARDO OLIVEIRA DA SILVA**

**SISTEMAS DE INTERFACES ALTERNATIVAS PARA MOBILIDADE REDUZIDA**

**FORTALEZA-CE**

**2026**

GABRIEL EDUARDO OLIVEIRA DA SILVA

**SISTEMAS DE INTERFACES ALTERNATIVAS PARA MOBILIDADE REDUZIDA**

Trabalho de Conclusão de Curso apresentado ao Curso de Design da Universidade Federal do Ceará, como requisito parcial à obtenção do título de Bacharel em Design.

Orientador: Prof. Dr. Roberto Cesar Cavalcante Vieira

**FORTALEZA-CE**

**2026**

GABRIEL EDUARDO OLIVEIRA DA SILVA

**SISTEMAS DE INTERFACES ALTERNATIVAS PARA MOBILIDADE REDUZIDA**

Trabalho de Conclusão de Curso apresentado ao Curso de Design da Universidade Federal do Ceará, como requisito parcial à obtenção do título de Bacharel em Design.

Aprovada em: \_\_\_\_/\_\_\_\_/\_\_\_\_

BANCA EXAMINADORA

---

Prof. Dr. Roberto Cesar Cavalcante Vieira.(Orientador)

Universidade Federal do Ceará (UFC)

---

Prof. Dr.Diego Éneas Ricca

Universidade Federal do Ceará (UFC)

Profa. Dr. Allyneanhy Gade Nunes Alves Oliveira

Universidade Federal do Ceará (UFC)

Prof. Dr. Edilson Porfirio Filho

Universidade Federal do Ceará (UFC)

## **AGRADECIMENTOS**

Agradeço antes de tudo a Deus, por me conceder força de vontade, dedicação e segurança.

À minha família, pelo amor incondicional, paciência e apoio em todos os momentos. Obrigado pai, mãe, irmã e minha madrinha, por todo o suporte emocional e financeiro, minha tia Silvia por ter mostrado a força e a importância de ser autêntico. Agradeço com carinho e respeito aos meus guias espirituais, de todas as falanges, linhas e caminhos de luz. Obrigado pela proteção, pelos sinais, pelas intuições e pela energia que me sustentou nos momentos de incerteza. Sei que cada passo foi amparado pelo invisível que me guia com amor.

Ao professor Roberto Vieira, meu orientador, agradeço sinceramente pela oportunidade de desenvolver este projeto, pelos conselhos valiosos, pela paciência e pela orientação dedicada. Sua contribuição foi fundamental para a realização deste trabalho. Aos demais que, direta ou indiretamente, contribuíram com esse projeto, deixo meu sincero reconhecimento e gratidão. E aos meus amigos Luciano, Jerdeth, Baracho, Cássio, Diego, Elisson, Thiago, Jadson, Tia Eunice, Mirian, Laureane e Suelen, meu muito obrigado pelos conselhos, pela amizade leal e por compartilharem essa caminhada comigo. Em cada conversa, incentivo e gesto, encontrei força para seguir em frente. E por último agradeço a mim, pelo esforço e a força, por ter lutado até o final e não ter desistido.

A todos vocês, meu eterno agradecimento.

## RESUMO

O presente trabalho aborda o desenvolvimento de tecnologias assistivas voltadas a pessoas com mobilidade reduzida severa , com o objetivo de ampliar sua autonomia na interação com sistemas digitais e dispositivos físicos. Diante das limitações impostas por interfaces tradicionais baseada em toque ou movimento , propõe-se o desenvolvimento do Sistema Gestus, uma interface alternativa baseada predominantemente em *eye tracking* e detecção de piscadas.

A metodologia adotada fundamenta-se nos princípios do Design Centrado no Usuário e da Interação Humano-Computador, priorizando acessibilidade, usabilidade e baixo custo. O desenvolvimento do sistema utilizou tecnologias acessíveis e de código aberto, como Arduino, OpenCV e MediaPipe, organizadas em uma arquitetura modular composta por blocos de interação independentes.

Como desenvolvimento prático, o sistema foi aplicado ao controle de uma interface gráfica com dispositivos físicos, representado por uma garra robótica assistiva. Interfaces cérebro-computador baseadas em EEG foram analisadas conceitualmente como possibilidade futura de expansão do sistema, não sendo implementadas nesta etapa.

Os resultados obtidos indicam que a arquitetura modular do Sistema Gestus favorece a adaptação a diferentes níveis de comprometimento motor, apresentando boa aceitação em testes de usabilidade. Conclui-se que o sistema representa um avanço relevante no desenvolvimento de tecnologias acessíveis, humanas e socialmente inclusivas.

**Palavras-chave:** acessibilidade; rastreamento ocular; mobilidade reduzida, EEG, TECNOLOGIAS ASSISTIVAS.

## ABSTRACT

"This study addresses the development of assistive technologies aimed at individuals with severe reduced mobility, with the goal of expanding their autonomy in interaction with digital systems and physical devices. Given the limitations imposed by traditional interfaces based on touch or movement, the development of the Gestus System is proposed—an alternative interface predominantly based on eye tracking and blink detection. The adopted methodology is grounded in the principles of User-Centered Design and Human-Computer Interaction, prioritizing accessibility, usability, and low cost. The system's development utilized accessible and open-source technologies, such as Arduino, OpenCV, and MediaPipe, organized in a modular architecture composed of independent interaction blocks.

As a practical development, the system was applied to the control of a graphical interface integrated with physical devices, represented by an assistive robotic gripper. Brain-computer interfaces based on EEG were conceptually analyzed as a future possibility for system expansion, though they were not implemented at this stage.

The results obtained indicate that the modular architecture of the Gestus System facilitates adaptation to different levels of motor impairment, showing good acceptance in usability tests. It is concluded that the system represents a significant advancement in the development of accessible, human-centered, and socially inclusive technologies."

**Palavras-chave:** accessibility; eye tracking; reduced mobility; EEG; assistive technologies.

## LISTA DE FIGURAS

Figura 1- Juliano Pinto utiliza o exoesqueleto de Miguel Nicolelis para o chute inaugural da Copa do Mundo. ....	18
Figura 2 - Ilustração do funcionamento básico de um eye tracker (A) e os padrões de reflexão da luz infravermelha na córnea em diferentes direcionamentos do olhar (B). .....	30
Figura 3 – Diagrama de um sistema de interface cérebro- computador (BCI) baseado em EEG para controle por meio da imaginação motora (MI) .....	34
Figura 4-Diagrama de um Pipeline de um sistema BCI .....	38
Figura 5 - Dispositivo Tobii Eyex Eye Tracker utilizado em estudos de rastreamento ocular assistivo, conforme exemplificado nas imagens (módulo acoplado ao monitor e aparelho isolado) .....	43
Figura 6 - Pranchas de Comunicação Alternativa para o ambiente hospitalar.....	44
Figura 7- protótipo de garra robótica modular.....	46
Figura 8 - diagrama de Ishikawa .....	40
Figura 9 - Diagrama de desenvolvimento de um sistema assistivo.....	41
Figura 10 - Interface inicial de testes em setembro, evidenciando a instabilidade do rastreamento ocular e a necessidade de inclinação da cabeça do usuário .....	51
Figura 11 - Representação visual dos dados brutos vs. dados filtrados pelo algoritmo de Kalman no rastreamento ocular de outubro.....	52
Figura 12 - Interface final do Sistema Gestus em novembro, apresentando a dinâmica de cards e o indicador visual de tempo de seleção (dwell time). ....	53

Figura 13 - Teste de integração final: comando ocular via Python resultando no movimento físico da garra robótica .....	54
Figura 14 - interface gráfica do sistema Gestus .....	58
Figura 15 - Interface do sistema Gestus e exemplo de usuário utilizando em tempo real .....	60
Figura 16 - Fluxo de interação biblioteca OpenCV .....	67
Figura 17 - MediaPipe , captação de pontos faciais .....	68
Figura 18 - Filtro de Kalman .....	69
Figura 19 - print da tela do bloco de interação Comunicador Ocular utilizando tabelas das necessidades.....	70
Figura 20 - Gráfico de análise de suavização .....	72
Figura 21 - fluxo de movimentação dos cards .....	73
Figura 22 - Sistema dwell time para ativação dos cards .....	75
Figura 23 - Fluxo de interação com algoritmo base.....	78
Figura 24- print da tela do bloco de interação Mouse Ocular81Figura 25 - print da tela do bloco de interação Comunicador Ocular .....	83
Figura 26 - Fluxo de ativação com feedback auditivo.....	84
Figura 27 - Fluxo de ativação com garra robótica .....	86
Figura 28 - Teste com usuário utilizando o sistema Gestus em tempo .....	89
Figura 29 - print dos feedbacks dos usuários em relação ao primeiro bloco de interação o Comunicador Ocular.....	92

Figura 30- print dos feedbacks dos usuários em relação ao primeiro bloco de interação o Mouse Ocular .....	93
Figura 31 - print dos feedbacks dos usuários em relação ao primeiro bloco de interação o Olhar smart.....	95
Figura 32 - print dos feedbacks dos usuários em relação ao primeiro bloco de interação garra robótica .....	96

## **LISTA DE TABELAS**

Tabela 01- Análise de similares resumida .....37

Tabela 02- Dados orçamentários dos materiais ..... 44

## LISTA DE ABREVIações E SIGLAS

ANN	Redes Neurais Artificiais
BCI	Interface cérebro computador
CAT	Comitê de Ajudas Técnicas
CNNS	Redes Neurais Convolucionais
DCU	Design Centrado no Usuário
ELA	Esclerose Lateral Amiotrófica
EEG	Eletroencefalograma
EMG	Eletromiografia
IHC	Interface Humano – Computador
IMUS	Sensores Inerciais
IBGE	Instituto Brasileiro de Geografia e Estatística
MI	Imaginação Motora
SSVEP	Potenciais Evocados Visuais de Estado Estável
SVM	Máquinas de Vetores de Suporte
TA	Tecnologias Assistivas
KNN	K-Nearest Neighbors
EAR	<i>Eye Aspect Ratio</i>

## LISTA DE SÍMBOLOS

\$ Reais

% Porcentagem

® Marca Registrada

## SUMÁRIO

<b>1. INTRODUÇÃO .....</b>	<b>17</b>
1.1 Problema da Pesquisa .....	20
1.2 Justificativa .....	22
1.3 Pergunta De Pesquisa .....	23
<b>2. OBJETIVOS .....</b>	<b>23</b>
2.1 Objetivo Geral.....	23
2.2 Objetivos Específicos .....	24
<b>3. FUNDAMENTAÇÃO TEÓRICA.....</b>	<b>24</b>
3.1 Tecnologia Assistiva.....	25
3.2 Interação Humano-Computador e Usabilidade.....	27
3.3 <i>Eye Tracking</i> como Interface Assistiva .....	29
3.4 Utilização de EEG (BCI) em Sistemas Assistivos.....	31
3.5 Possibilidades Futuras com Sensores de Movimento .....	34
3.6 Algoritmos de Controle.....	37
3.7 Usabilidade em Sistemas Assistivos.....	39
<b>4. ANÁLISES DE SIMILARES.....</b>	<b>41</b>
4.1 Interface cérebro-máquina de baixo custo – Grandini (2017) .....	41
4.2 EEG-Controlled Wheelchair – IEEE (2020) .....	42
4.3 Sistema de rastreamento ocular como alternativa de controle – Dusik (2018).....	42
4.4 Projeto Santos Vieira (2020) .....	43
4.5 Análise crítica e contribuições.....	44

<b>5. METODOLOGIA .....</b>	<b>38</b>
5.1 Abordagem Geral .....	38
5.2 Design Centrado no Usuário (DCU).....	39
5.3 Diagrama de Ishikawa .....	40
5.4 Ciclo de Desenvolvimento .....	40
5.5 Avaliação e Técnicas de Validação .....	42
5.6 Design de interfaces e Design modular .....	43
5.7. Diretrizes projetuais.....	45
5.8 Aspectos Éticos .....	50
5.9 Desenvolvimento Iterativo: Ciclos de Prototipagem e Refinamento do Sistema....	50
<b>6. SISTEMA GESTUS – CONCEPÇÃO E DESENVOLVIMENTO .....</b>	<b>55</b>
<b>7. FUNCIONAMENTO DO SISTEMA GESTUS .....</b>	<b>66</b>
7.1 Camada de Aquisição de Dados Visuais .....	66
7.2 Camada de Análise Facial e Rastreamento Ocular .....	67
7.3 Processamento Matemático e Inferência de Piscadas .....	68
7.5 Camada de Interface e Feedback Multissensorial .....	70
<b>8. LÓGICA DE INTERAÇÃO POR PISCADAS E NAVEGAÇÃO NO SISTEMA GESTUS .....</b>	<b>71</b>
8.1 Piscadas como Evento de Entrada Intencional .....	72
8.2 Diferenciação Direcional: Piscada Direita e Piscada Esquerda .....	73
8.3 Navegação Linear e Controle Cognitivo .....	73
8.4 Tempo de Permanência (Dwell Time) como Confirmação de Ação .....	74
8.5 Feedback Visual e Auditivo com Reforço da Interação .....	75
8.6. Relação entre Lógica de Interação e Design Modular.....	76
<b>9. BLOCOS DE INTERAÇÃO DO SISTEMA GESTUS .....</b>	<b>78</b>
9.1 Bloco Mouse ocular .....	79

9.2 Bloco Comunicador Ocular .....	82
9.3 Bloco Olhar Smart .....	85
<b>10. TESTES COM USUÁRIOS E VALIDAÇÃO DO SISTEMA .....</b>	<b>88</b>
10.1 Procedimento do teste .....	89
10.2 Instrumentos de avaliação .....	90
10.3 Avaliação por blocos de interação .....	91
10.4 Síntese dos resultados .....	97
<b>CONCLUSÃO .....</b>	<b>99</b>
<b>APÊNDICE A .....</b>	<b>101</b>
<b>APÊNDICE B .....</b>	<b>105</b>
<b>APÊNDICE C .....</b>	<b>111</b>
<b>APÊNDICE D .....</b>	<b>115</b>
<b>REFERÊNCIAS .....</b>	<b>117</b>

## 1. INTRODUÇÃO

A acessibilidade é consagrada como um direito humano fundamental nas leis brasileiras e nos tratados internacionais de direitos humanos. No entanto, milhões de pessoas enfrentam diariamente barreiras físicas, tecnológicas e comunicativas que limitam sua plena participação na sociedade. Entre essas barreiras, a mobilidade é um fator restritivo significativo. Isso pode ocorrer de forma temporária ou a longo prazo devido a amputações, paralisias, doenças neuromusculares, lesões físicas ou problemas congênitos.

Mais de 14 milhões de pessoas no país têm algum tipo de deficiência motora, de acordo com o Instituto Brasileiro de Geografia e Estatística (IBGE) em 2022. Além disso, fatores relacionados à idade e um aumento nos acidentes continuam a aumentar esse número. Isso serve como um lembrete de que tecnologias que garantam autonomia e independência funcional são necessárias mais do que nunca (TEIXEIRA et al., 2018). Nesse contexto, Damásio (1996, p.216) destaca que:

“Nossos sentimentos parecem depender de um complexo sistema dotado de diversos elementos associados à regulação biológica. A razão, por sua vez, dependeria de sistemas cerebrais particulares, sendo que alguns deles processam sentimentos. Isso faz com que possa existir um elo entre sentimentos e razão, e entre eles e o corpo, do ponto de vista anatômico e funcional.”

Ao compreender a relação entre a emoção, razão e corpo é fundamental pensar em soluções eficazes. Já que a construção segura da autodeterminação está ligada à capacidade do indivíduo de tomar decisões. Por esse motivo, tanto a emoção quanto a razão são envolvidas nesse processo.

Um exemplo emblemático dessa integração entre processamento cerebral, reconquista de autonomia e mobilidade física restaurada é o uso do exoesqueleto robótico controlado por EEG, que foi apresentado durante a cerimônia de abertura da Copa do Mundo de 2014, no Brasil. Desenvolvido pelo Projeto Andar de Novo, liderado pelo neurocientista Miguel Nicolelis, esse exoesqueleto permitiu que um

voluntário paraplégico realizasse um chute simbólico em campo, utilizando sinais cerebrais não invasivos para controlar o traje robótico. Como representado na Figura 1, apesar da demonstração na TV ter durado apenas alguns segundos, ela simbolizou o forte impacto emocional e social gerado pelo restabelecimento, mesmo que parcial e simbólico, da mobilidade de um indivíduo que vivia em uma condição de paralisia permanente.

Esse caso ilustra perfeitamente um salto de como as tecnologias assistivas, como rastreamento ocular, adotado como interface principal do sistema, e as interfaces cérebro-computador baseadas em EEG, analisadas como alternativa conceitual e possibilidade de expansão futura. A pessoa voluntária toma a decisão consciente, depois ela experimenta a emoção significativa da autonomia e o envolvimento social, e vê isso refletido em um movimento pelo seu próprio corpo (ação física), tudo mediado por uma tecnologia assistiva.

Figura 1- Juliano Pinto utiliza o exoesqueleto de Miguel Nicolelis para o chute inaugural da Copa do Mundo.



Fonte – G1 (2014)

Portanto, tecnologias verdadeiramente assistivas não podem negar as emoções das pessoas durante a usabilidade, bem como as percepções de ter controle sobre seus ambientes e vidas. Nesse contexto, a Tecnologia Assistiva (TA) surge para atender a essa necessidade. Ela abrange dispositivos, serviços, procedimentos e

abordagens desenvolvidos para pessoas com deficiência, a fim de melhorar suas habilidades funcionais e integrá-las à sociedade. Como vê Bersch (2017), a TA não apenas compensa uma deficiência, mas também abre possibilidades que reintegram a pessoa ao seu ambiente e comunidade.

Nesse contexto, o presente trabalho propõe o desenvolvimento de um comunicador ocular, caracterizado como um sistema de interface alternativas voltado a pessoas com mobilidade reduzida. O sistema utiliza predominantemente o rastreamento ocular (*eye tracking*) como meio de entrada, permitindo que o usuário envie comandos por meio do movimento dos olhos. A interação será demonstrada por meio de um controle de uma garra robótica, representando a tradução da intenção do usuário em uma ação física e também a demonstração de blocos de interação que integram todo o sistema da interface criada.

Durante o processo de concepção do projeto, foram analisadas diferentes possibilidades de interação assistiva, incluindo o uso de interfaces cérebro-computador baseadas em EEG (eletroencefalograma) — exame neurofisiológico não invasivo que registra a atividade elétrica espontânea do cérebro —. Embora o EEG apresente potencial significativo, o *eye tracking* foi definido como interface principal devido à facilidade de acesso aos equipamentos necessários, menor complexidade de uso e viabilidade prática no contexto proposto, mantendo-se o EEG como uma possibilidade futura dentro de uma arquitetura modular do sistema.

Assim, este estudo insere-se no campo da acessibilidade técnica e do design de interfaces assistivas, adotando o Design Centrado no Usuário como abordagem metodológica (BAXTER,2011). Fundamentado nos princípios da interação Humano-Computador, o trabalho prioriza a usabilidade, acesso e a experiência do usuário (BARBOSA et al. ,2021), visando o desenvolvimento de um sistema inclusivo, funcional e socialmente relevante. Durante o desenvolvimento desta pesquisa, foi estabelecida uma parceria interdisciplinar com o curso de Engenharia de Computação da Universidade Federal do Ceará, por meio da colaboração com Professor Edilson e bolsista Luís Carlos Rodrigues graduando em engenharia da computação. Essa cooperação surgiu ao longo do processo projetual e teve como objetivo fortalecer o

desenvolvimento de um sistema de comunicação assistiva, integrando conhecimentos das áreas de design, computação e engenharia.

A proposta foi conduzida de forma colaborativa entre os envolvidos, permitindo a troca constante de experiências técnicas para construção de uma solução mais robusta e alinhada às reais necessidades dos usuários com mobilidade reduzida. O desenvolvimento deste trabalho também está vinculado ao Programa de Extensão PROTECS — Programa de tecnologias, inovação e inclusão da Universidade federal do Ceará — iniciativa que tem como propósito integrar tecnologia, inovação e acessibilidade para a promoção da inclusão social e a transformação da vida de pessoas com deficiência. A participação no PROTECS possibilitou a ampliação do escopo do projeto, promovendo um ambiente colaborativo entre docentes, discentes e pesquisadores de diversas áreas do conhecimento. Foi por meio dele que o sistema que será comentado nos próximos capítulos pôde evoluir conceitualmente e tecnicamente, fortalecendo sua proposta como uma solução assistiva de caráter interdisciplinar, social e tecnológico. Embora interfaces cérebro-computador baseadas em sinais cerebrais (EEG) sejam discutidas ao longo do trabalho, estas não foram implementadas nesta etapa, sendo abordadas apenas como possibilidades futuras de expansão dos sistemas. A relevância deste estudo reside em sua abordagem interdisciplinar, integrando design, computação e tecnologia assistiva, além de sua preocupação com acessibilidade, reprodutibilidade e sustentabilidade tecnológica.

## **1.1 Problema da Pesquisa**

A mobilidade reduzida afeta uma ampla gama de pessoas, incluindo amputados, idosos ou até mesmo indivíduos com paralisia e doenças neurodegenerativas; isso afeta seriamente sua capacidade de interagir com o mundo ao seu redor. Um grande número de tecnologias assistivas existentes depende do controle motor — especialmente dos membros superiores —, e isso efetivamente leva à exclusão de uma grande parte da população doente que possui limitações físicas severas ou tem o controle motor comprometido.

Além disso, as interfaces projetadas carecem de personalização e adaptabilidade; a maioria dos sistemas assistivos baseia-se em uma padronização funcional, sem considerar as particularidades subjetivas de cada usuário. Há uma falha

não apenas tecnológica, mas também moralista, simbólica e política, porque, por esse meio, essas pessoas não conseguem participar plenamente da sociedade.

“A privação dos direitos humanos acontece quando uma pessoa não apenas deixa de pertencer a uma comunidade política, mas perde o espaço onde suas ações e palavras têm relevância. Ela não é mais considerada alguém com uma história a ser ouvida, com uma identidade reconhecida ou com direitos que possam ser exigidos. Perder o lugar no mundo é perder a condição de ser visível para os outros como alguém que importa.” (ARENDDT, 2009, p. 376)

Essa invisibilidade gerada pela exclusão tecnológica é o que torna urgente a reformulação do design de sistemas de interação. O desafio está em criar soluções em rastreamento ocular, capazes de ampliar autonomia de pessoas com comprometimento motor severo considerando tecnologias como EEG enquanto possibilidade futura de integração. A Tecnologia Assistiva, segundo o Comitê de Ajudas Técnicas (CAT), deve ser compreendida como:

“[...] uma área do conhecimento, de característica interdisciplinar, que engloba produtos, recursos, metodologias, estratégias, práticas e serviços que objetivam promover a funcionalidade, relacionada à atividade e participação de pessoas com deficiência, incapacidades ou mobilidade reduzida, visando sua autonomia, independência, qualidade de vida e inclusão social.” (BRASIL/CAT, 2009)

O mesmo fluxo é encontrado nos comentários de DUSIK (2013), que a tecnologia assistiva, devidamente projetada, pode reconstruir a autoimagem e ser uma contribuição tremenda para o usuário, de modo que, uma vez mais na sociedade, ele se sinta ativo, produtivo e pleno.

## 1.2 Justificativa

Apesar da rápida evolução das tecnologias assistivas nas últimas décadas, muitas soluções ainda dependem fortemente do movimento corporal ou do uso de mãos e braços para interação. Para indivíduos com comprometimento motor severo - como pessoas com tetraplegia, amputações bilaterais ou doenças neuromusculares degenerativas, como o ELA (ESCLEROSE LATERAL AMIOTRÓFICA) - tais tecnologias tornam-se inadequadas ou completamente inviáveis.

Diante desse cenário, evidencia-se a existência de uma lacuna crítica: a necessidade de sistemas verdadeiramente inclusivos, capazes de adaptar seus funcionamentos às condições específicas de cada usuário, e não o contrário. Conforme apontam Dusik (2018) e Grandini (2017), a efetiva inclusão só ocorre quando a tecnologia se molda às necessidades individuais, respeitando os diferentes graus de limitação funcional.

Nesse contexto, o uso do rastreamento ocular apresenta-se como uma alternativa viável e acessível, uma vez que mesmo indivíduos com mobilidade reduzida frequentemente mantêm o controle do movimento dos olhos. Adicionalmente, a exploração de sinais cerebrais por meio de tecnologias baseadas em EEG amplia ainda mais o espectro de acessibilidade, possibilitando a interação mesmo em casos de mobilidade extremamente limitada ou inexistente. Embora o EEG represente uma abordagem promissora, sua aplicação prática ainda enfrenta desafios relacionados à complexidade técnica, à calibração e ao custo dos equipamentos, o que limita sua adoção em soluções acessíveis e de ampla disseminação.

Outro aspecto relevante que justifica este estudo, está relacionado à viabilidade técnica e econômica dessas tecnologias. Os custos de dispositivos de rastreamento ocular e sensores EEG, apesar de estarem diminuindo significativamente, ainda são relativamente altos, quando falamos de custo benefício e acessibilidade para a maior parte da população que delas necessita. Assim, torna-se fundamental investigar alternativas que conciliam eficiência funcional, baixo custo e possibilidade de reprodução em diferentes contextos.

Assim, este trabalho justifica-se tanto pela urgência técnica de desenvolver sistemas de interação mais acessíveis quanto por sua relevância social, ao propor uma solução que valoriza a autonomia, a dignidade humana e a inclusão. A pesquisa fundamenta-se na concepção de Tecnologia Assistiva definida pelo Comitê de Ajudas Técnicas (CAT), entendido como:

“[...] uma área do conhecimento, de característica interdisciplinar, que engloba produtos, recursos, metodologias, estratégias, práticas e serviços que objetivam promover a funcionalidade, relacionada à atividade e participação de pessoas com deficiência, incapacidades ou mobilidade reduzida, visando à sua autonomia, independência, qualidade de vida e inclusão social.”  
(BRASIL/CAT,2009)

Além disso, conforme destacado por Dusik (2013), tecnologias assistivas adequadamente projetadas podem contribuir para reconstrução da autoimagem do usuário, favorecendo sua reinserção social e o sentimento de pertencimento, produtividade e plenitude.

### **1.3 Pergunta De Pesquisa**

Assim, este trabalho pergunta: Como desenvolver um sistema assistivos modular, de baixo custo, baseado em rastreamento ocular e detecção de piscadas, capaz de ampliar a autonomia de pessoas com diferentes níveis de comprometimento motor severo por meio da comunicação, do controle digital e da interação com dispositivos físicos?

## **2. OBJETIVOS**

### **2.1 Objetivo Geral**

Desenvolver um sistema modular de interação assistiva baseado em rastreamento ocular e detecção de piscadas, voltada à ampliação da autonomia de pessoas com mobilidade reduzida, analisando tecnologias alternativas, como EEG, enquanto possibilidades futuras de expansão do sistema.

## **2.2 Objetivos Específicos**

1. Investigar tecnologias assistivas voltada à interação humano-computador, incluindo rastreamento ocular, comunicação alternativa interfaces cérebro-computador em EEG, identificando suas potencialidades e limitações.
2. Analisar critérios de viabilidade, custos, acessibilidade e reprodutibilidade das tecnologias estudadas, justificando a escolha do rastreamento ocular e da detecção de piscadas como abordagem principal do projeto.
3. Aplicar princípios de design modular, design de interfaces e design centrado no usuário na concepção de um sistema assistivo organizado em blocos de interação de interação independentes e integráveis.
4. Realizar testes com usuários para avaliar a usabilidade, eficiência e aceitabilidade do sistema.
5. Avaliar o sistema quanto aos princípios de baixo custo, sustentabilidade e reprodutibilidade, com base nos recursos tecnológicos utilizados.

## **3. FUNDAMENTAÇÃO TEÓRICA**

Este trabalho tem como objetivo apresentar os principais conceitos, abordagens e referenciais que sustentam o desenvolvimento do Sistema GESTUS, situando-o no campo das tecnologias assistivas, da interação humano-computador e do design de interfaces acessíveis. Este capítulo busca contextualizar o projeto a partir de uma revisão do estado da arte, estabelecendo uma base conceitual que permita compreender as escolhas metodológicas, tecnológicas e projetuais adotadas ao longo do trabalho.

Inicialmente, são discutidos os conceitos de tecnologia assistiva e acessibilidade, com ênfase em soluções voltadas à ampliação da autonomia de

‘

peças com comprometimento motor severo. Em seguida, são abordados os princípios da interação humano-computador e da usabilidade, fundamentais para a concepção de interfaces centradas no usuário e adequadas a contextos de uso assistivo.

Na sequência, o capítulo aprofunda-se no rastreamento ocular (eye tracking) como alternativa de interface, destacando seu potencial como meio de interação não invasivo, de baixo custo e tecnicamente viável. Essa abordagem constitui o eixo central do Sistema GESTUS, sendo analisada tanto sob a perspectiva técnica quanto projetual.

Por fim, são apresentadas as interfaces cérebro-computador baseadas em sinais cerebrais (EEG), tratadas neste trabalho em caráter conceitual e exploratório. A inclusão dessa abordagem tem como finalidade ampliar a compreensão do estado da arte em tecnologias assistivas, permitindo uma análise comparativa entre diferentes formas de interação e apontando possibilidades de expansão futura do sistema, sem que haja implementação prática de EEG nesta etapa do projeto.

### **3.1 Tecnologia Assistiva**

O desenvolvimento de tecnologias voltadas à inclusão de pessoas com deficiência, mobilidade reduzida ou limitações funcionais está diretamente relacionada ao campo da Tecnologia Assistivas (TA). Trata-se de uma área interdisciplinar que integra conhecimentos provenientes de diversas disciplinas, como engenharia, ciência da computação, design, psicologia, fisioterapia e educação, com o objetivo de promover autonomia, funcionalidade e participação social.

De acordo com o Comitê de Ajudas Técnicas (CAT), instalado por uma Portaria nº 142/2006, da Secretaria Especial de Direitos Humanos da Presidência da República, a Tecnologia Assistiva pode ser definida como:

“uma área do conhecimento, de característica interdisciplinar, que engloba produtos, recursos, metodologias, estratégias, práticas e serviços que objetivam promover a funcionalidade, relacionada à atividade e participação de pessoas com deficiência, incapacidades ou mobilidade reduzida, visando sua

autonomia, independência, qualidade de vida e inclusão social” (BRASIL, 2007 apud TEIXEIRA et al., 2018)

A TA abrange um amplo espectro de soluções, que vão desde tecnologias de baixa complexidade — como talheres adaptados, pranchas de comunicação e andadores — até tecnologias de maior complexidade, incluindo softwares de rastreamento ocular, interfaces cérebro-computador, próteses inteligentes e dispositivos robóticos avançados (BERSCH, 2017). Nesse contexto, a tecnologia atua como meio de equalização social, conforme sintetizado por Radabaugh (1993 apud BERSCH, 2017), ao afirmar que, enquanto para pessoas sem deficiência a tecnologia facilita tarefas, para pessoas com deficiência ela torna ações possíveis.

Cook e Hussesy (1995) destacam que os recursos assistivos não devem ser compreendidos apenas como artefatos físicos isolados, mas como parte de um sistema de apoio que envolve treinamento, adaptação, manutenção e acompanhamento contínuo do usuário. Dessa forma, a Tecnologia Assistiva não se configura como um produto estático, mas como um processo dinâmico, que deve se ajustar continuamente às necessidades, capacidades e contextos de uso de cada indivíduo.

Sob a perspectiva da Interação Humano-Computador (IHC), Barbosa et al. (2021) apontam que o sucesso de uma solução assistiva depende diretamente de sua capacidade de adaptação às limitações, preferências e características cognitivas, emocionais e físicas dos usuários. Assim, aspectos como acessibilidade, usabilidade e experiência do usuário tornam-se centrais no desenvolvimento de sistemas assistivos eficazes.

O presente projeto insere-se nesse campo ao propor um sistema assistivo baseado principalmente em rastreamento ocular, voltado à ampliação das possibilidades de interação de pessoas com comprometimento motor severo. A escolha dessa abordagem fundamenta-se em seu caráter não invasivo, em sua viabilidade técnica e em seu potencial de aplicação prática em contextos reais de uso.

Interfaces cérebro-computador baseadas em sinais cerebrais (EEG) são consideradas neste trabalho exclusivamente em nível conceitual, compondo a revisão do estado da arte em tecnologias assistivas. Sua análise contribui para a compreensão das potencialidades e limitações dessa abordagem, sem que haja aplicação prática ou implementação experimental no desenvolvimento do Sistema GESTUS.

Destaca-se ainda que o desenvolvimento de soluções no campo da Tecnologia Assistiva deve considerar princípios do design universal e do design centrado no usuário. Embora o design universal busque soluções acessíveis ao maior número possível de pessoas, reconhece-se que, em muitos casos, é necessária a personalização das interfaces de acordo com necessidades clínicas, cognitivas e emocionais específicas. Conforme aponta Moreira Júnior (2018), a aplicação da Tecnologia Assistiva baseada em um diagnóstico social e funcional adequado pode atuar como vetor de transformação real na vida das pessoas com deficiência, promovendo não apenas facilidades operacionais, mas também autonomia, identidade e participação social.

Dessa forma, a presente pesquisa situa-se em uma fronteira tecnológica e ética ao propor um sistema que dispensa o uso de habilidades motoras finas ou força muscular, utilizando o controle ocular como principal meio de interação. Tal abordagem amplia o potencial da Tecnologia Assistiva e reforça o papel da tecnologia como instrumento de inclusão e reintegração social.

### **3.2 Interação Humano-Computador e Usabilidade**

A IHC (Interação Humano-Computador) estuda o design, implementação e avaliação de sistemas de computador interativos que colocam a necessidade, problema ou demanda humana como foco. A IHC baseia-se em teorias da ergonomia, ciência da computação e psicologia — todas essas disciplinas contribuem para o design de sistemas de informação. Interseções relacionadas encontram-se em campos multidisciplinares como ciência cognitiva e IA, bem como terapia lúdica e tecnologia.

No contexto da Tecnologia Assistiva, o design centrado no usuário está se tornando mais relevante porque envolve diretamente o design de soluções que devem ser funcionais, acessíveis e significativas para diferentes níveis de pessoas com

deficiência. Além disso, propõe-se que as tecnologias adotem a perspectiva do usuário. Que tipo de pessoa usará a tecnologia? Quais são os aspectos físicos, emocionais, cognitivos e ambientais disso?

Um dos pioneiros da disciplina — Donald Norman (2003) — diz: “Os bons designs são invisíveis: não chamam atenção para si mesmos, mas servem como pontes naturais entre as intenções das pessoas e as ações possíveis” (NORMAN, 2013, p. 11).

Essa concepção é particularmente pertinente no desenvolvimento de sistemas assistivos, nos quais a interface não deve representar uma barreira adicional, mas sim facilitar a interação de forma intuitiva, fluida e compreensível. Em sistemas assistivos, a mediação entre o usuário e o computador torna-se ainda mais desafiadora, sobretudo quando o público-alvo apresenta limitações motoras severas, como a impossibilidade de utilizar dispositivos de entrada tradicionais, a exemplo de teclado, mouse ou telas sensíveis ao toque. Nesses casos, a interação deve minimizar a complexidade cognitiva e física, substituindo comandos convencionais por formas de entrada mais acessíveis e compatíveis com as capacidades remanescentes do usuário.

Nesse sentido, o rastreamento ocular destaca-se como uma alternativa eficaz de interação, por permitir o controle de interfaces digitais sem a necessidade de movimentos corporais finos ou força muscular. Essa abordagem, adotada de forma prática no desenvolvimento do Sistema GESTUS, alinha-se aos princípios da IHC ao priorizar usabilidade, acessibilidade e adequação ao contexto real de uso.

Outras abordagens, como sistemas baseados em interfaces cérebro-computador fundamentadas em sinais cerebrais (EEG), são discutidas neste trabalho apenas em nível teórico, compondo a revisão do estado da arte em interação assistiva. Embora apresentem potencial para usuários com níveis extremos de comprometimento motor, tais tecnologias não foram implementadas no Sistema GESTUS, sendo consideradas como possibilidades de expansão futura da arquitetura proposta.

### 3.3 *Eye Tracking* como Interface Assistiva

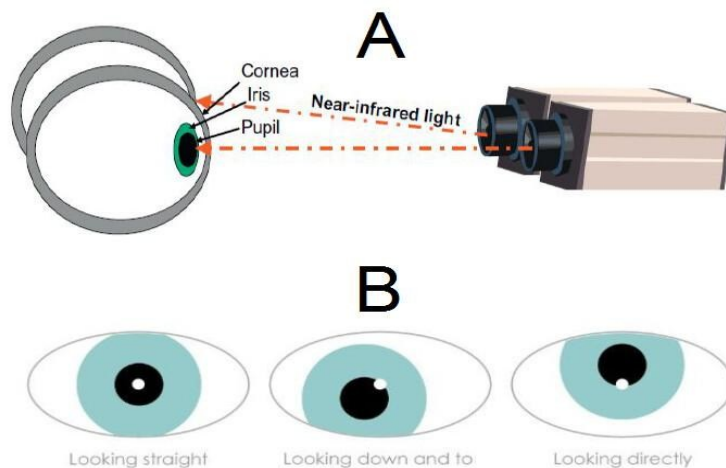
O Rastreamento ocular, ou *eye tracking*, é uma tecnologia para detectar e interpretar os movimentos e fixações oculares de um indivíduo. Originalmente destinada como uma ferramenta de pesquisa em psicologia cognitiva e marketing, essa tecnologia tem encontrado uso crescente em acessibilidade devido à capacidade de interações sem o uso das mãos. Dessa forma, tornou-se uma alternativa fundamental para pessoas com deficiências motoras severas, especialmente aquelas que não conseguem utilizar membros superiores ou a fala como meio de interação.

Segundo Neto (2012, apud Burguer e Knoll, 2018, p. 342), o *eye tracking* “analisa o que é observado pelo usuário por meio de seus movimentos oculares milimétricos, permitindo a formulação de hipóteses e conclusões impossíveis de serem obtidas por outras técnicas”. Essa precisão possibilita que o olhar seja convertido em comandos intencionais, funcionando como um canal direto entre intenção do usuário e a ação do sistema.

De acordo com Barbosa et al. (2021), uma interface assistiva deve adaptar-se às capacidades físicas, cognitivas e sensoriais do usuário, e não exigir que o usuário se adapte à tecnologia. Nesse sentido, o rastreamento ocular apresenta-se como uma solução especialmente promissora, pois explora um canal de comunicação frequentemente preservado mesmo em condições clínicas que comprometem severamente a função motora: o olhar.

O funcionamento básico de um sistema de rastreamento ocular envolve o uso de câmera infravermelhas capazes de captar a reflexão da luz na córnea do usuário. A partir dessas reflexões, é possível mapear com precisão a direção do olhar e identificar pontos de fixação na tela ou no ambiente físico, conforme ilustrado na Figura 2. Essas informações podem ser convertidas em comandos, permitindo o controle de interfaces gráficas, cursores, teclados virtuais ou dispositivos físicos, como explorado neste projeto.

Figura 2 - Ilustração do funcionamento básico de um eye tracker (A) e os padrões de reflexão da luz infravermelha na córnea em diferentes direcionamentos do olhar (B).



Fonte: Brain Support (2023)

Existem várias aplicações bem-sucedidas do rastreamento ocular como recurso assistivo, conforme mostrado na literatura atual. Por exemplo, Dusik (2018) relata o uso de interfaces visuais para escrever texto em teclados virtuais por fixação ocular. Assim, pessoas com Síndrome do Encarceramento – uma condição em que a pessoa está acordada, mas completamente imóvel, exceto pelos olhos – podem se comunicar.

Além de sua alta precisão, o rastreamento ocular também possui outras vantagens importantes: como uma tecnologia não invasiva, é rapidamente calibrado e, com o avanço dos instrumentos comerciais, está se tornando cada vez mais acessível. Empresas como Tobii®, EyeTech®, TheEyeTribe® e Irisbond® oferecem tanto hardware quanto SDKs para esses produtos que permitem tal integração em sistemas de softwares personalizados construídos em linguagens como Python, Javascript ou C++.

Quanto à Tecnologia Assistiva, o rastreamento ocular também é bem aceito pelos usuários. Isso porque, ao contrário de sensores musculares que podem ser sensíveis ao movimento ambiente, botões de pressão e controles de voz, ele requer apenas um mínimo de esforço do usuário. Além disso, pode ser usado em ambientes barulhentos, bem como por pacientes que têm fala reduzida, diz Grandini (2017) a visão

simplesmente se torna um gesto de comando quando todos os outros meios estão bloqueados.

A implementação de sistemas baseados em rastreamento ocular ainda enfrenta muitos problemas. Entre eles estão a necessidade de calibração individualizada, exaustão ocular, sensibilidade à luz ambiente e eventuais atrasos no processamento dos comandos. Além disso, o design da interface visual assume papel central, exigindo cuidados específicos relacionados ao espaçamento dos elementos, à clareza visual e ao tempo de fixação necessário para ativação de comandos.

Diante dessas considerações, o presente trabalho adota o rastreamento ocular como a interface principal de interação, por apresentar maior maturidade tecnológica, viabilidade prática e adequação ao escopo desta pesquisa. Embora as interfaces cérebro-computador baseadas em EEG tenham sido analisadas e discutidas ao longo do trabalho, sua implementação não foi realizada nesta etapa, sendo considerada uma possibilidade para investigações futuras. Assim, o *eye tracking* foi escolhido como solução central por oferecer um equilíbrio entre acessibilidade, confiabilidade técnica e potencial de aplicação imediata.

Desta forma, o rastreamento ocular não se configura apenas como um recurso tecnológico, mas como uma ponte entre intenção e ação, capaz de restaurar a agência, a autonomia e autonomia e a expressão de indivíduos com mobilidade severamente reduzida, fundamentando-se em princípios de Interação Humano-Computador e Design Centrado no Usuário, conforme será aprofundado nos capítulos seguintes.

### **3.4 Utilização de EEG (BCI) em Sistemas Assistivos**

As interfaces cérebro-computador (Brain-Computer Interfaces – BCI) representam uma das abordagens mais relevantes e promissoras no campo das tecnologias assistivas contemporâneas. Essas interfaces possibilitam que pessoas com deficiências motoras severas interajam com sistemas computacionais, dispositivos robóticos ou membros artificiais por meio da captação e interpretação de sinais eletrofisiológicos do cérebro, geralmente obtidos por eletroencefalografia (EEG).

O princípio fundamental das interfaces BCI consiste na tradução de padrões de atividade cerebral em comandos computacionais, buscando inferir as intenções do usuário sem a necessidade de movimentos musculares voluntários. Conforme descrito por Grandini (2017), um sistema BCI é composto por uma arquitetura que realiza a aquisição, o processamento e a classificação de sinais bioelétricos cerebrais, associando-os a ações específicas em sistemas digitais ou físicos.

Essa tecnologia apresenta especial relevância para indivíduos que perderam parcial ou totalmente o controle motor em decorrência de condições como tetraplegia, esclerose lateral amiotrófica (ELA) ou síndrome do encarceramento, nas quais os canais tradicionais de interação se encontram severamente comprometidos. Nesse contexto, o EEG atua como meio alternativo de comunicação e controle.

De modo geral, os sistemas BCI utilizam sensores de EEG posicionados no couro cabeludo para registrar variações de potencial elétrico geradas pela atividade neural. Após a etapa de aquisição, os sinais passam por processos de filtragem, extração de características e reconhecimento de padrões, permitindo sua conversão em comandos, como a movimentação de um cursor, a seleção de elementos em interfaces gráficas ou o acionamento de dispositivos físicos.

Entre os principais paradigmas utilizados em interfaces BCI não invasivas, destacam-se:

- P300 – baseado na detecção de uma resposta cerebral involuntária que ocorre aproximadamente 300 milissegundos após a apresentação de um estímulo relevante. É amplamente empregado em teclados virtuais e sistemas de seleção por atenção visual.
- SSVEP (Steady-State Visually Evoked Potentials) – fundamentado na resposta cerebral a estímulos visuais intermitentes com frequências específicas, permitindo a seleção de comandos a partir do foco visual do usuário. Embora eficaz, pode gerar desconforto visual em usos prolongados.
- Imaginação Motora (MI) – baseada na ativação de áreas cerebrais relacionadas à simulação mental de movimentos, como mover a mão ou

o pé, mesmo sem execução física. Essa abordagem é particularmente adequada para tarefas de controle contínuo, como cadeiras de rodas ou braços robóticos.

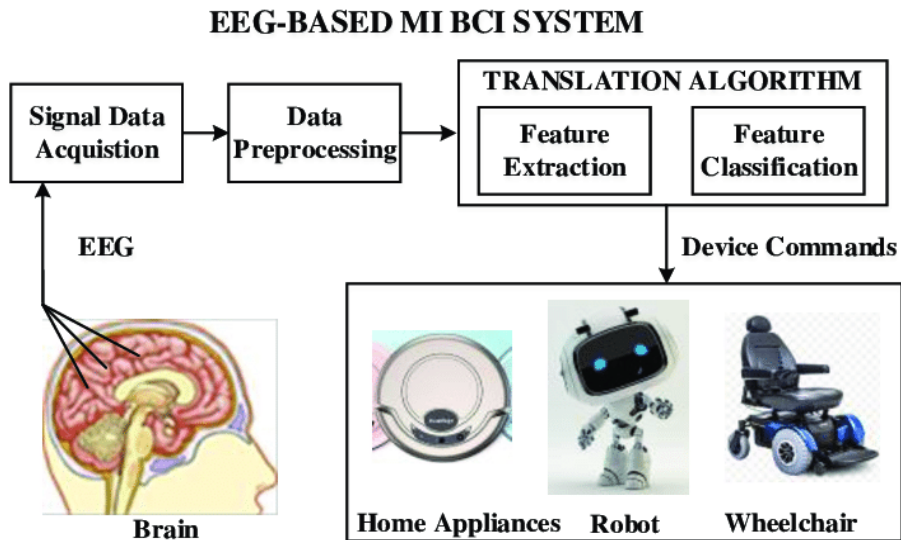
Uma das principais vantagens dessas abordagens reside no fato de serem não invasivas, dispensando implantes cerebrais. Dispositivos comerciais como Emotiv Insight, OpenBCI e NeuroSky MindWave ampliaram o acesso a esse tipo de tecnologia, tornando seu uso mais viável em contextos acadêmicos e experimentais. Grandini (2017), por exemplo, apresenta uma aplicação do Emotiv Insight para operação de um teclado virtual controlado por sinais cerebrais, evidenciando o potencial dessas interfaces para a comunicação assistiva.

Apesar de seu potencial, as interfaces BCI baseadas em EEG enfrentam desafios técnicos e operacionais significativos, como a variabilidade dos sinais cerebrais, a necessidade de treinamento prolongado dos usuários, a fadiga cognitiva e a sensibilidade a ruídos elétricos. Além disso, diferenças fisiológicas individuais impactam diretamente o desempenho e a confiabilidade desses sistemas, restringindo sua aplicação, na maioria dos casos, a ambientes controlados ou experimentais.

Nesse sentido, Nicoletis (2024) ressalta o papel ativo do cérebro na construção da realidade e no planejamento de ações futuras, ao afirmar que “o cérebro cria uma simulação interna do mundo e, a partir dela, age sobre o ambiente e planeja o futuro” (NICOLELIS, 2024). Essa compreensão reforça tanto o potencial quanto a complexidade envolvida no uso de sinais cerebrais como meio de interação.

Do ponto de vista da Tecnologia Assistiva, o uso de interfaces BCI baseadas em EEG representa uma possibilidade relevante de ampliação do acesso à comunicação e ao controle para pessoas com comprometimento motor extremo. Contudo, conforme apontam Teixeira et al. (2018), tais tecnologias devem ser compreendidas não apenas como avanços técnicos, mas também como desafios éticos e antropológicos, exigindo cuidado em sua aplicação.

Figura 3 – Diagrama de um sistema de interface cérebro-computador (BCI) baseado em EEG para controle por meio da imaginação motora (MI)



Fonte – Research Gate (2019)

Diante dessas considerações, o presente trabalho aborda as interfaces cérebro-computador baseadas em EEG exclusivamente como parte do estado da arte em tecnologias assistivas. Essa abordagem é analisada como alternativa complementar e possibilidade de integração futura, não sendo adotada como interface principal no desenvolvimento do Sistema GESTUS. A escolha do rastreamento ocular como eixo central do sistema fundamenta-se em sua maior maturidade tecnológica, confiabilidade, menor complexidade operacional e adequação ao escopo e aos objetivos desta pesquisa.

### 3.5 Possibilidades Futuras com Sensores de Movimento

Esta categoria de dispositivos desempenha um papel importante na área de Tecnologia Assistiva e pode aumentar significativamente a interatividade de sistemas projetados para pessoas com deficiências de mobilidade. Embora suas descobertas não implementem de fato tais sensores neste momento, a abordagem conceitual da pesquisa em relação a eles como um potencial desenvolvimento futuro pode ser

creditado por sua importância estratégica em permitir sistemas acessíveis para necessidades dinâmicas e personalizadas.

Esses sensores funcionam detectando os movimentos residuais do corpo: desde uma leve inclinação da cabeça aqui ou ali, até contrações nos músculos; na melhor das hipóteses, eles também podem lidar com movimentos mais amplos onde há alguma preservação das habilidades motoras. Muitas vezes, pessoas que não conseguem mover os membros superiores ainda podem mover os ombros, a cabeça ou até mesmo expressar comandos através de expressões faciais. Esses sinais podem ser interpretados por várias tecnologias:

- Acelerômetros e Giroscópios (encontrados comumente em smartphones e microcontroladores como Arduino e Raspberry Pi);
- Sensores inerciais (IMUs), que incorporam em três eixos diferentes tipos de movimentos;
- Sensores de proximidade (Infravermelho, Ultrassônico, Capacitivo);
- Eletrodos de eletromiografia (EMG), que captam os músculos eletricamente ativos;
- Kits de visão computacional como o Kinect da Microsoft ou câmeras equipadas com Inteligência Artificial.

Através de sensores de movimento, o sistema adaptativo reconhecerá um conjunto de padrões de movimento personalizados e atribuirá funções específicas para eles, como pressionar teclas em frente a teclados físicos, mover o ponteiro sob controle do usuário ou atuadores mecânicos ativados por movimento físico.

Em situações mais avançadas, começando apenas com a entrada de um único usuário que é fisicamente incapacitado e não pode fornecer entrada direta por si só, mas impacta um ponto de dados específico para si mesmo individualmente, uma rede neural artificial aprenderá cada gesto ou sinal individual da pessoa, resultando em sistemas que respondem de forma mais fluida e são mais inclusivos.

Por exemplo, se sensores de movimento fossem integrados ao sistema proposto aqui, pessoas que ainda possuem algum movimento no pescoço poderiam inclinar suas cabeças e o aperto do robô se moveria. Ou em outro caso: aproveitar pequenas contrações musculares ao redor da boca ou nos ombros poderia significar comandos reais. Todas essas coisas funcionam juntas em um ambiente de acessibilidade holística — onde o rastreamento ocular se combina com EEG em todos os lugares para aqueles que precisam, além de sensores físicos para aqueles que os utilizam.

Barbosa et al. (2021) defendem, que em todos os sistemas interativos acessíveis, o objetivo é que "a interação se adapte aos recursos do usuário enquanto respeita seus limites e maximiza seu potencial residual". Essa flexibilidade é uma das principais forças dos sensores de movimento, que funcionam como interfaces alternativas para usuários com diferentes graus de habilidade motora.

No entanto, para que tal desenvolvimento seja bem-sucedido, vários desafios devem ser considerados:

- Calibração personalizada: cada sensor deve ser ajustado para cada indivíduo e situação;
- Fadiga física: pessoas com alguma mobilidade podem facilmente se cansar de fazer a mesma coisa várias vezes;
- Ruído e interferência: em ambientes com múltiplas fontes de estimulação ou movimento, os sensores podem fazer leituras imprecisas;
- Ergonomia e conforto: os sensores devem ser leves, pequenos em tamanho e fáceis de colocar/retirar ou usar.

Assim, embora esta pesquisa não incorpore inicialmente sensores de movimento em seu sistema, ela reconhece a importância estratégica de tais componentes em desenvolvimentos futuros. Idealmente, as soluções assistivas se moverão em direção a uma interação multimodal, permitindo que vários tipos de entrada (olhar, ondas cerebrais, movimento do corpo) trabalhem juntos - produzindo

interfaces que são mais seguras, mais robustas e menos deixadas para pessoas com deficiências.

### **3.6 Algoritmos de Controle**

A eficiência de um sistema assistivo depende diretamente da inteligência dos algoritmos de controle que interpretam, processam e convertem os sinais capturados em comandos executáveis ou em uma série de ações ordenadas. Esses algoritmos são os intermediários entre a intenção do usuário e a ação realizada no sistema — como mover uma garra robótica, selecionar um botão virtual ou enviar uma mensagem.

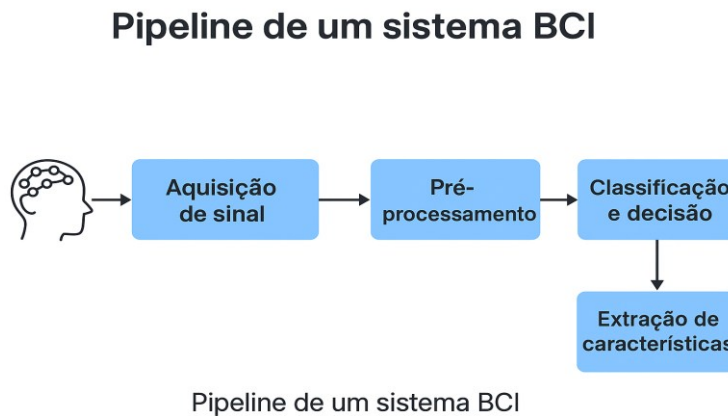
Quando um sistema é baseado em rastreamento ocular, seus algoritmos precisam observar padrões de fixação ocular, movimentos rápidos e súbitos dos olhos e piscadas. Ele utiliza o tempo de fixação — tipicamente entre 500 a 1000 milissegundos — como critério para executar uma função (seleção por olhar e piscadas). Para extrair esses dados, filtros de ruído produzem o sinal, a detecção de eventos seleciona o que é relevante em meio a tudo que está acontecendo, e então temos a lógica de controle que faz a coordenada do olhar realizar algo.

Em sistemas baseados em EEG, os algoritmos de controle são mais complexos — como naqueles que empregam sinais P300, SSVEP ou de imaginação motora. O pipeline típico de um BCI consiste em quatro estágios principais:

1. Aquisição de sinal: realizada por sensores de EEG posicionados de acordo com o sistema 10-20.
2. Pré-processamento: limpeza de ruídos, artefatos musculares e normalização de dados.
3. Extração de características: identificação de tipos significativos de padrões nos sinais, como potência em bandas específicas (alfa, beta, teta).
4. Classificação e decisão: uso de algoritmos de aprendizado de máquina, como Máquinas de Vetores de Suporte (SVM), Redes Neurais Artificiais

(ANN), Floresta Aleatória ou K-Nearest Neighbors (KNN) para atribuir um rótulo (comando) ao padrão identificado.

Figura 4-Diagrama de um Pipeline de um sistema BCI



Nesse ponto, o algoritmo final deve ativar a interface de saída, que pode ser um atuador físico (como um motor de garra robótica) ou um comando digital em uma interface gráfica. Em projetos como Grandini (2017), o algoritmo de controle foi responsável por transformar comandos mentais em digitação de letras em um teclado virtual. Após calibração e treinamento, ele poderia traduzir comandos em texto com uma precisão superior a 70%.

Além das técnicas clássicas, redes neurais convolucionais (CNNs) e aprendizado profundo estão começando a encontrar seu lugar na tecnologia assistiva. Especialmente notável é sua capacidade de lidar com dados não lineares, como os de EEG. No entanto, o último método requer maior poder computacional e uma quantidade mais abundante de dados, o que limita sua aplicação a sistemas embarcados de baixo custo.

Também se pode imaginar uma abordagem de controle contínuo (abrindo e fechando a garra gradualmente com controle proporcional). Em termos da lógica de controle que se aplica a pinças robóticas como a representação física deste trabalho, várias formas são possíveis. É tarefa do usuário especificar essas possibilidades.

Obviamente, o algoritmo aceitará um comando (por olhar ou EEG), validará a entrada, enviará sinais digitais para o microcontrolador (como Arduino ou ESP32), e então terá esse microcontrolador movendo motores de acordo com sua lógica programada.

A unificação de várias fontes de entrada (olhar, cérebro e futuros sensores de movimento) pode ser tratada por um módulo de arbitragem de controle. Este módulo pode decidir qual entrada tem prioridade máxima, ou pode atuar como um sistema redundante: dois ou mais sinais devem confirmar o mesmo comando antes que qualquer coisa seja executada.

Baxter (2011, p. 198) enfatiza: "Para projetos centrados no usuário, a confiabilidade e previsibilidade do sistema de controle são fatores críticos se o usuário deve acreditar ou usar". Um sistema assistivo que age de forma imprevisível ou ineficaz pode facilmente causar frustração, insegurança e abandono tecnológico.

Portanto, os algoritmos de controle nos sistemas assistivos devem possuir alta precisão: mesmo quando o usuário ou o ambiente varia rápido: a interação suave depende da velocidade.

Não apenas alta precisão, mas também adaptabilidade e segurança — capaz de manter operações involuntárias ou perigosas dos usuários ao mínimo. Com base na arquitetura modularizada deste trabalho, a inspeção legislativa e a revisão de algoritmos pelo usuário podem ser continuamente melhoradas, de acordo com os princípios do Design Centrado no Usuário, por etapas, envolvendo testes, *feedback* e aprendizado contínuo.

### **3.7 Usabilidade em Sistemas Assistivos**

A usabilidade é um dos conceitos básicos da IHC e, de acordo com a ISO 9241-11 (1998), "a extensão em que um produto pode ser usado por usuários específicos para alcançar objetivos específicos com eficácia, eficiência e satisfação em um contexto de uso especificado." Em outras palavras, além de um sistema funcionar,

está a sua elegância. Não basta que o sistema funcione; ele deve fazê-lo bem, com o mínimo de esforço físico e cognitivo por parte de seus usuários-alvo.

Para pessoas com mobilidade reduzida, a usabilidade evidente consiste em simplicidade, clareza, tempo de resposta e ausência de carga mental. Usando uma escala como a System Usability Scale (SUS), desenvolvida por Brooke (1996), é possível adaptar testes para as opiniões dos usuários sobre sistemas de rastreamento ocular e EEG. Variáveis a serem consideradas incluem facilidade de uso, previsibilidade de comandos, níveis de conforto visual, bem como a sensação geral de controle do usuário.

Além disso, as considerações dos usuários sobre usabilidade devem abranger erros: nas palavras de Norman (2013), um excelente sistema é aquele que evita erros, mas também fornece rotas fáceis de volta à segurança. Esta é uma característica crítica em sistemas que atendem a grupos minoritários ou até mesmo à segurança pública, como os assistivos, onde operações incorretas podem colocar pessoas em risco ou tornar sua interação com o sistema frustrante.

## **2.8 Experiência do Usuário (UX) e Acessibilidade**

A Experiência do Usuário (UX) inclui não apenas a função, mas também os sentimentos, percepções, expectativas e valores que as pessoas atribuem ao uso de um produto. No caso de sistemas assistivos, isso pode significar mais do que apenas mover braços robóticos; pode significar recuperar um senso de propósito e dignidade a cada movimento. Nas palavras de Barbosa et al. (2021), "Na IHC Inclusiva, a Experiência do Usuário (UX) geral deve basear-se em 3 essenciais:

1. **Significância:** a tecnologia deve cobrir as necessidades pessoais do usuário;
2. **Satisfação:** a interação deve ser agradável e clara;
3. **Empoderamento:** o sistema deve restaurar o controle e o palco para um indivíduo com deficiência."

A acessibilidade, por outro lado, é a parte técnica disso, para que todos tenham uma experiência que possam compartilhar. Isso inclui considerações como contraste adequado para todos, com elementos grandes e bonitos, tolerância a erros (um sistema de rastreamento ocular), suporte para entradas alternativas conforme necessário; tempo de interação ajustável dependendo da necessidade e feedback visual e auditivo claro.

Portanto, no projeto proposto, a IHC, usabilidade e experiência do usuário são integradas com base no princípio do Design Centrado no Usuário (DCU), que é detalhado no Capítulo 4. Ele visa alcançar um objetivo onde a tecnologia não seja apenas acessível, mas também significativa, ética e afetiva, reconhecendo assim as complexidades de projetar para corpos e mentes que rompem com as normas de interação computacional.

#### **4. ANÁLISES DE SIMILARES**

A análise de trabalhos correlatos permite contextualizar o presente projeto dentro do panorama atual da pesquisa em tecnologia assistiva, identificando avanços, limitações e lacunas ainda existentes. Neste sentido, serão apresentados estudos que envolvem o uso de *eye tracking*, EEG (BCI), controle de dispositivos assistivos e integração homem-máquina para pessoas com mobilidade reduzida.

##### **4.1 Interface cérebro-máquina de baixo custo – Grandini (2017)**

O trabalho de Jônatas Grandini apresentou um sistema BCI de baixo custo usando o sensor Emotiv Insight e foi destinado a controlar um teclado virtual através de EEG. O projeto mostrou que, mesmo com dispositivos prontos para uso, é viável projetar e implantar um sistema BCI rudimentar para facilitar a comunicação para usuários com deficiências motoras severas. Nas palavras do autor: "o mérito do trabalho é permitir que pessoas com deficiências motoras tenham uma melhor qualidade de vida, podendo se comunicar e interagir com pessoas e o ambiente" (GRANDINI, 2017, p. 10).

Apesar de não ser baseado em componentes físicos como próteses ou atuadores, estabeleceu uma estrutura para que sinais cerebrais fossem aplicados a

sistemas de controle de dispositivos, o que proporcionou legitimidade aos BCIs para fins assistivos com baixos custos adicionais.

#### **4.2 EEG-Controlled Wheelchair – IEEE (2020)**

Em um artigo apresentado na conferência ICCS (Conferência Internacional sobre Computação e Sistemas), pesquisadores escreveram uma ideia original onde um EEG poderia ser usado para controlar uma cadeira de rodas, sendo a imaginação motora a abordagem principal. Neste estudo, propôs uma abordagem de aprendizado de máquina para realizar a classificação de sinais cerebrais (esquerda, direita, para cima, para baixo) usando o sensor NeuroSky.

A contribuição destacou o uso de feedback auditivo e visual para aumentar a confiança do usuário e seus testes reais com usuários em ambientes internos. No entanto, o tempo de resposta foi destacado como um problema, prescrevendo a necessidade de algoritmos mais rápidos e precisos. Nosso trabalho preenche a lacuna ao demonstrar que sistemas móveis (por exemplo, uma cadeira de rodas) podem ser controlados através de ações cerebrais, sob a condição de personalização e adaptação do usuário.

#### **4.3 Sistema de rastreamento ocular como alternativa de controle – Dusik (2018)**

Dusik examinou o uso de rastreamento ocular para comunicação e controle por pessoas com LIS. O estudo empregou o Tobii Eyex para manipulação do cursor do mouse e acionamento de teclado virtual e apresentou resultados positivos em precisão e usabilidade. A figura 5, ilustra o dispositivo utilizado, evidenciando a interface de controle ocular proposta.

Figura 5 - Dispositivo Tobii Eyex Eye Tracker utilizado em estudos de rastreamento ocular assistivo, conforme exemplificado nas imagens (módulo acoplado ao monitor e aparelho isolado)



Fonte: adaptado de Thankachan (2018) .

O estudo apoia que o ET é econômico, ergonômico, direto em comparação com sistemas de controle por movimento muscular ou voz, particularmente em pacientes que permanecem com OC como a única forma de comunicação no mundo. O autor afirma que: "ao substituir a dependência motora por uma interface baseada no olhar, o esforço físico necessário para controlar o sistema é amplamente reduzido, levando a uma maior autonomia e autoestima por parte do usuário" (DUSIK, 2018, p. 37).

#### **4.4 Projeto Santos Vieira (2020)**

O projeto Santos Vieira configura-se como referência relevante no campo da tecnologia assistiva ao investigar e aplicar sistemas de comunicação alternativa baseadas em cartões (*cards*) no contexto da terapia ocupacional, com ênfase no atendimento às pessoas diagnosticadas com Esclerose Lateral Amiotrófica (ELA). A proposta do projeto parte da compreensão de que a progressão da doença compromete gradualmente as habilidades motoras voluntárias, tornando inadequados ou inviáveis dispositivos que dependam de movimentos finos, força muscular ou manipulação direta.

Nesse cenário, a utilização de *cards* (Figura 6) como unidades mínimas de comunicação apresenta vantagens significativas. Cada *card* representa uma ação, mensagem ou intenção claramente definida, estruturada de forma visual simbólica, o que facilita a compreensão imediata por parte do usuário. Essa organização contribui para a redução da carga cognitiva e favorece processos de seleção baseados em interações simples e repetitivas, amplamente recomendadas em práticas terapêuticas voltadas para à reabilitação e à manutenção da autonomia comunicacional.

Figura 6 - Pranchas de Comunicação Alternativa para o ambiente hospitalar



#### 4.5 Análise crítica e contribuições

Embora os exemplos acima tenham contribuído, observa-se que a maioria dos trabalhos utiliza apenas uma única modalidade de entrada - seja EEG ou rastreamento ocular -, com exceção de abordagens híbridas ou modulares, como o trabalho nesta tese. Além disso, certos trabalhos não lidam com a integração em um dispositivo físico (por exemplo, atuadores ou próteses robóticas), o que diminui o impacto prático em relação às manipulações em ambientes reais.

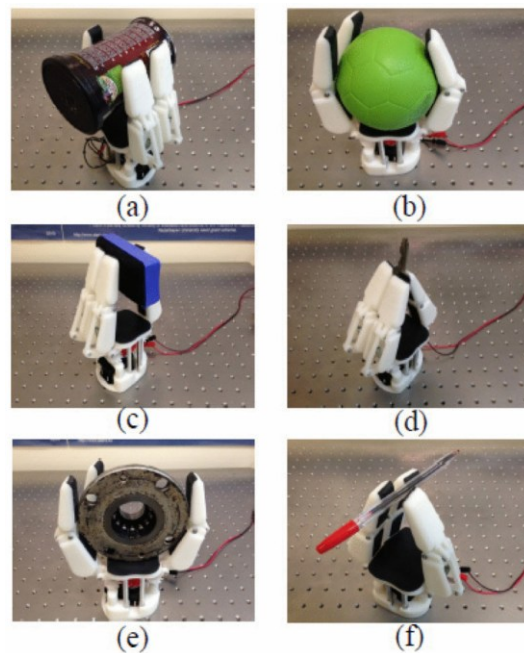
O controle físico completo (garra robótica), com entrada através de rastreamento ocular e a possibilidade de expansão para EEG, pode oferecer uma combinação de acessibilidade, facilidade de uso e um nível razoável de robustez. Com isso, o projeto se aproxima de uma tecnologia assistiva multimodal capaz de lidar com diferentes perfis funcionais de pessoas com mobilidade reduzida.

**TABELA 01 – ANÁLISE DE SIMILARES RESUMIDA**

Autor/ano	Tecnologias	Aplicação	Relevância para este projeto
Dusik (2018)	EEG	Cadeira de rodas	Inspiração para uso de EEG em controle real
Grandini (2017)	EEG	Ações com atenção / relaxamento	Base para uso de EEG de baixo custo
Teixeira (2018)	<i>Eye tracking</i>	Comunicação com olhar	Base de interface visual e feedbacks
Microsoft (2017)	<i>Eye tracking</i>	Comunicação por quadrantes	Uso funcional do olhar em quadros severos
Barbosa (2021)	IHC	Usabilidade e acessibilidade	Base para avaliação centrada no usuário
Projeto Santos Vieira (2020)	<i>Eye tracking</i>	Usabilidade e acessibilidade	Uso funcional do olhar em casos severos

Esta revisão mostra que, embora algum trabalho já tenha sido dedicado ao controle assistivo através de sinais alternativos, há espaço abundante para soluções mais modulares, combináveis e personalizáveis. O sistema proposto aqui visa exatamente combinar e verificar essas tendências com um protótipo físico funcional (garra robótica). A Figura 7, ilustra um exemplo de garra robótica modular e adaptativa, alinhada conceitualmente à proposta desenvolvida, evidenciando seu potencial de aplicação em sistemas assistivos personalizáveis.

Figura 7- protótipo de garra robótica modular



Fonte: Adaptado de ResearchGate (2018); Frontiersin (2023); Hackster.io (2020).

**Descrição:** (a)-(b) modelos adaptativos impressos em 3D com três dedos desmontáveis para diferentes tipos de objetos; (c) mão robótica com atuadores bidirecionais para controle responsivo ao toque; (d) sistema aplicado em robô móvel com garra funcional para manipulação de objetos.

## **5. METODOLOGIA**

A metodologia adotada neste trabalho caracteriza-se como uma pesquisa aplicada, de natureza qualitativa e exploratória, com foco no desenvolvimento e avaliação de uma tecnologia assistiva funcional. O estudo fundamenta-se nos princípios do Design Centrado no Usuário e da Interação Humano-Computador, priorizando usabilidade, acessibilidade e experiência do usuário.

O desenvolvimento do Sistema GESTUS ocorreu em etapas iterativas, envolvendo levantamento teórico, concepção do sistema, prototipação, implementação e testes com usuários. A arquitetura do sistema foi projetada de forma modular, permitindo a integração de diferentes blocos de interação, como comunicação alternativa, controle digital e dispositivos físicos.

Para a implementação do rastreamento ocular e da detecção de piscadas, foram utilizadas bibliotecas de visão computacional e inteligência artificial, como OpenCV e MediaPipe, possibilitando a identificação de padrões oculares em tempo real. A inteligência artificial atuou como suporte na interpretação dos dados visuais e na adaptação do sistema às variações individuais dos usuários.

Os testes de usabilidade foram conduzidos com usuários convidados, considerando critérios como eficiência, facilidade de uso e aceitabilidade. As avaliações foram realizadas por meio de observação direta e aplicação de perguntas estruturadas, permitindo identificar pontos fortes e oportunidades de melhoria no sistema.

### **5.1 Abordagem Geral**

A presente pesquisa caracteriza-se como aplicada de abordagem qualitativa, com ênfase no desenvolvimento e experimentação de um protótipo funcional. A pesquisa aplicada justifica-se pelo objetivo de gerar uma solução prática voltada à ampliação da autonomia de pessoas com comprometimentos motores, enquanto a abordagem qualitativa permite a análise aprofundada dos aspectos

relacionados à experiência de uso, à usabilidade e à adaptação do sistema às necessidades dos usuários.

O desenvolvimento do trabalho priorizou a experimentação prática, contemplando tanto a construção física e digital do sistema quanto a realização de testes com usuários. Esses testes tiveram como finalidade verificar a efetividade do sistema em um nível mínimo de operação, possibilitando ajustes iterativos ao longo do processo de projeto. A avaliação do sistema considerou tanto critérios objetivos — como tempo de resposta, precisão e confiabilidade da interação — quanto critérios subjetivos, relacionados à usabilidade, ao conforto cognitivo e à percepção da experiência de uso.

Essa abordagem metodológica está alinhada a métodos centrados no usuário, os quais reconhecem que a interação com sistemas tecnológicos envolve dimensões cognitivas, sociais e biológicas. Conforme apresentado por Zaltman (2003, apud Burguer e Knoll, 2018, p. 341), “a atividade mental decorre da interação entre os processos sociais e biológicos “. Tal perspectiva reforça a necessidade de considerar o usuário de forma integral, indo além dos aspectos puramente técnicos do sistema.

Dessa forma, a adoção de métodos centrados no usuário justifica-se pela compreensão de que o design de sistemas assistivos deve contemplar simultaneamente fatores técnicos, humanos e sociais. Essa abordagem permite desenvolver soluções mais adequadas à realidade dos usuários promovendo maior eficiência funcional, conforto na interação e aceitação do sistema no contexto de uso.

## **5.2 Design Centrado no Usuário (DCU)**

O projeto é baseado no princípio do Design Centrado no Usuário. Este modelo é representado por Mike Baxter (2011) em relação ao design de produtos, considerando as seguintes etapas:

1. Estar ciente do usuário e da situação de uso
2. Especificações de requisitos

### 3. Prototipagem e teste

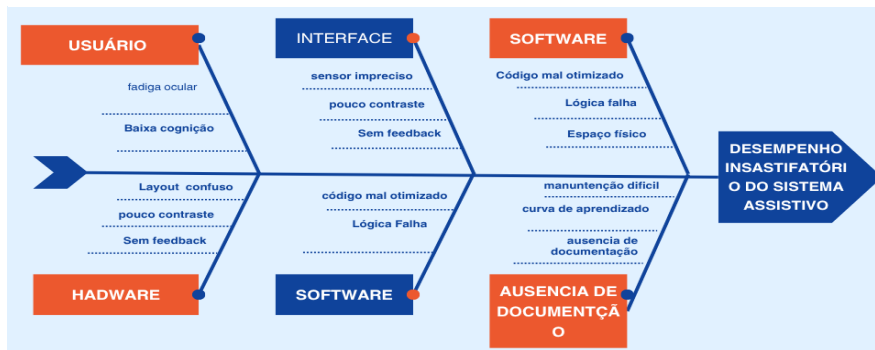
- **Avaliação prática**

O design centrado no usuário tem como objetivo desenvolver produtos úteis, utilizáveis e desejáveis, a partir de um entendimento profundo das tarefas, necessidades e desejos das pessoas (BAXTER, 2011, p.230).

### 5.3 Diagrama de Ishikawa

Para antecipar causas de desempenho insatisfatório do sistema, utilizou-se o Diagrama de Causa e Efeito (Ishikawa, 1986), conforme a Figura 8.

Figura 8 - diagrama de Ishikawa

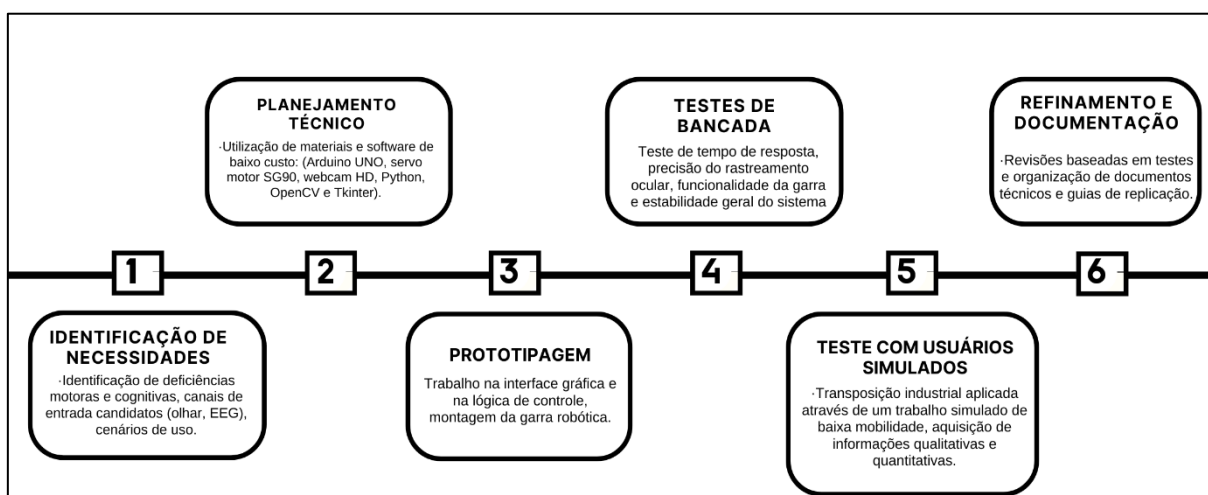


Fonte – elaborada pelo autor

### 5.4 Ciclo de Desenvolvimento

O desenvolvimento do sistema foi guiado por um ciclo iterativo baseado nos estudos de Dusik (2018) e Grandini (2017), reforçado pelos conceitos de usabilidade e neuro design (Soler, 2024; Norman, 2002). A figura 9 apresenta, em formato de linha do tempo, as principais etapas de desenvolvimento desse sistema. Desde a identificação das necessidades dos usuários até a fase final de documentação técnica, o diagrama sintetiza o caminho percorrido dentro deste projeto. A proposta combina inovação tecnológicas com responsabilidade social, reforçando o papel do design na criação de soluções transformadoras.

Figura 9 - Diagrama de desenvolvimento de um sistema assistivo



Fonte - elaborado pelo autor

#### 5.4.1 Estratégia de Aprendizado Autodidata Assistido

Para viabilizar a execução técnica do Sistema Gestus, o autor adotou uma metodologia de aprendizado contínuo ao longo de nove meses. Diante da necessidade de dominar um ecossistema *Full-Stack* (do processamento de dados ao desenvolvimento de interfaces), utilizou-se a Inteligência Artificial (IA) não como uma ferramenta de geração de código, mas como um tutor pedagógico interativo. O objetivo central foi a transição do conhecimento em Design para a implementação em Engenharia, permitindo que o autor projetasse e escrevesse cada linha de código do sistema de forma consciente e fundamentada.

#### 5.4.2 Protocolo de Instrução (Engenharia de Prompt)

O aprendizado foi estruturado através de um protocolo rigoroso de consulta à IA. Para garantir que a ferramenta atuasse estritamente na capacitação, foi desenvolvido um "Prompt Mestre" que configurou a IA como um professor especializado em educação inclusiva e didática progressiva.

**Diretriz de Aprendizado utilizada:** *"Atue como um professor de programação. Crie um curso de 9 meses, do Python ao JavaScript. As aulas devem ser curtas, didáticas e progressivas. Utilize analogias visuais e verbais, exemplos reais*

*e exercícios de fixação. Não forneça o código pronto; ensine o passo a passo para que eu mesmo possa construir o projeto final."*

Este método permitiu que o autor superasse barreiras complexas de sintaxe e lógica, focando na compreensão profunda dos conceitos em vez da simples cópia de algoritmos.

### **5.4.3. Cronograma de Maturação Técnica**

A capacitação foi dividida em três trimestres, garantindo que a base lógica fosse sólida antes da implementação da interface visual:

1. **Fundamentos e Back-end (Mês 1 ao 3):** Foco em lógica de programação com Python, manipulação de variáveis e estruturas de controle. Esta base foi essencial para o processamento das coordenadas oculares.
2. **Sistemas em Tempo Real (Mês 4 ao 5):** Estudo de tarefas assíncronas e comunicação via WebSockets, permitindo que o Python (visão computacional) conversasse com a interface web sem atrasos (*delay*).
3. **Front-end e Integração (Mês 6 ao 9):** Domínio de HTML5, CSS3 e JavaScript. O foco final foi a integração de todos os módulos com a garra robótica, resultando em um sistema funcional e responsivo.

### **5.5 Avaliação e Técnicas de Validação**

A validação do sistema é realizada nos seguintes dois níveis:

#### **Avaliação Técnica:**

- Tempo de resposta do sistema;
- Precisão do rastreamento ocular;

- Confiabilidade da ativação física.

### **Avaliação de Usabilidade e Experiência:**

- Escala de Usabilidade do Sistema (SUS), por Brooke (1996), personalizada para uso assistivo;
- Entrevistas com usuários simulados;
- Observação de sinais diretos (atitudes faciais, hesitações, fluência);
- Avaliação subjetiva da sensação de controle, clareza e conforto visual.

"Experiência é a conexão emocional que transforma tecnologia em valor."  
(GABRIEL; KISO, 2020, p. 122)

### **5.6 Design de interfaces e Design modular**

A metodologia adotada neste trabalho fundamenta-se em uma abordagem projetual centrada no usuário, integrando princípio do design de interfaces e do design modular como estratégias metodológicas para o desenvolvimento de sistemas assistivos de interação humano - computador. Tal abordagem busca garantir que as decisões de projeto sejam orientadas pelas capacidades limitações e necessidades reais dos usuários, especialmente em contextos de comprometimento motor severo.

No que se refere ao design de interfaces, esse foi empregado como método para estruturar a relação entre o usuário e sistema, orientando a definição da arquitetura da informação, a organização dos elementos interativos e os fluxos de navegação. Norman (2013) destaca que a qualidade da interface está diretamente relacionada à clareza com que o sistema comunica suas possibilidades de ação. Em complemento a isso, Dieter Rams estabelece que a excelência de um projeto reside na capacidade de se tornar auto explicativo e minimalista, conforme sintetizado nos seus dez princípios:

“O bom design é inovador. O bom design torna o produto útil. O bom design é estético. O bom design torna o produto compreensível. O bom design é discreto. O bom design é honesto. O bom design é duradouro. O bom design é meticuloso até o último detalhe. O bom design respeita o meio ambiente. O bom design é o mínimo de design possível (Rams, 2017, p. 22).”

Ainda segundo Norman (2013),” quando as pessoas utilizam objetos mal projetados, não é culpa delas, mas do design” (NORMAN, 2013, p. 15). Essa afirmação reforça a necessidade metodológica de reduzir a carga cognitiva e minimizar erros de interação, especialmente em tecnologias assistivas, nas quais o esforço físico e mental do usuário deve ser reduzido ao máximo.

Sob essa perspectiva do design centrado no usuário, a metodologia adotada alinha-se às diretrizes propostas por ISO 9241-210 (2019), que define que sistemas interativos devem ser desenvolvidos com base na compreensão explícita dos usuários, de suas tarefas e de seus ambientes de uso. A norma enfatiza, que “*o design centrado no ser humano visa tornar os sistemas utilizáveis e úteis, focando nos usuários, suas necessidades e requisitos*” (ISO 9241-210,2019, p.3). Tal orientação reforça a adoção dessa abordagem como método neste trabalho. Complementarmente, o design modular foi incorporado como abordagem metodológica para a organização estrutural do sistema, orientando sua decomposição em unidades funcionais independentes e interconectadas.

Essa característica mostra-se particularmente relevante no desenvolvimento de tecnologias assistivas, uma vez que, conforme Dusik (2018), soluções inclusivas devem ser capazes de adaptar-se às condições específicas de cada usuário. Dessa forma, a integração entre design de interfaces, design centrado no usuário e design modular, enquanto abordagens metodológicas, orientou todas as etapas do processo projetual, desde a concepção entre estrutura, interação e uso, além de alinhar o desenvolvimento do sistema aos objetivos de acessibilidade, flexibilidade e reprodutibilidade estabelecidos neste trabalho. As decisões metodológicas aqui descritas fundamentam a contração da solução apresentada nos

capítulos subsequentes, nos quais a materialização desses princípios é detalhada e analisada.

## **5.7. Diretrizes projetuais**

Os princípios incluídos na diretriz de design são princípios que ajudam a definir decisões técnicas, funcionais, estéticas e éticas no desenvolvimento do sistema assistivo proposto. Estas foram as regras, baseadas em uma revisão teórica e experiências anteriores, os seguintes autores também produziram as mesmas recomendações (Baxter, 2011, Soler, 2024, Norman, 2002). Elas estão lá para garantir que o trabalho do sistema em construção seja:

- Funcionalmente acessível;
- Tecnicamente viável;
- Cognitivamente possível;
- Socialmente inclusivo;

As recomendações foram divididas em cinco eixos:

### **5.7.1 Acessibilidade e Inclusão**

O design do sistema foi fundamentado nos princípios do Design Universal, onde um sistema deve ser utilizável pelo maior número de pessoas possível, independentemente de suas demandas de funcionamento adaptativo.

- O sistema é livre de mãos ou membros superiores e utiliza o olhar como canal de entrada.
- O eletroencefalograma (EEG) é considerado uma tecnologia suplementar e opcional para uso em testes piloto para aqueles que têm perda completa do controle ocular.

- A interface de tela e os comandos físicos também devem estar disponíveis para pessoas com baixa visão de acordo com regras de contraste, espaço, tipografia grande e sobrecarga não visual.
- As instruções devem ser fáceis de usar com baixa demanda física e mental, sem a necessidade de precisão, força ou mesmo procedimentos complicados.

### **5.7.2 Usabilidade, Interação e Neuro design**

Layout, cores, tempo de resposta e organização intersticial de itens apropriados para a fase foram projetados de acordo com os princípios de usabilidade (ISO 9241-11, 1998) e neuro design, conforme explicado por Soler (2024).

- O sistema precisa ser eficiente, eficaz, e os usuários devem estar satisfeitos apesar das limitações sensoriais e motoras com os sentidos e órgãos.
- Deve haver uma hierarquia visual clara para os elementos da interface, com ênfase na atenção focal em primeiro plano e conteúdo de fundo minimamente distrativo.
- Rótulos ou ícones pouco claros e semelhantes devem ser evitados na interface para que esta seja previsível e fácil de aprender.
- A latência entre a instrução visual (fixações do olhar) e a ação real (movimento da garra) deve ser  $<1$  s para interação natural.
- Todas as ações no sistema devem resultar em uma resposta visual, auditiva ou tátil explícita, para transmitir uma sensação de controle.

"Sistemas que são acessíveis devem primeiro ser cognitivamente e afetivamente acessíveis ao usuário" (SOLER, 2024, p. 115).

### 5.7.3 Baixo Custo, Sustentabilidade e Reprodutibilidade

A viabilidade prática foi a prioridade no design do sistema, à luz de escolas técnicas, ONGs e famílias de baixa renda.

- Todas as peças devem ser de baixo custo disponíveis no Brasil, por exemplo, a webcam comum, servo motor SG90, Arduino UNO e etc.
- Usar bibliotecas e software de código aberto ou gratuitos (Python, OpenCV, Tkinter, Firmata, Arduino IDE, e assim por diante).
- A garra robótica precisa ser construída a partir de peças recicláveis ou impressas em 3D e ser capaz de ser montada em oficinas básicas.
- O projeto deve ser muito bem documentado, acompanhado de manual de montagem, diagramas de fiação e comentários no código-fonte para permitir que outros o reproduzam facilmente.

Nisso, foram estabelecidos dados orçamentários referentes aos dispositivos, com base nos princípios de baixo custo. Além disso, no que diz respeito a ambientes de desenvolvimentos, serão utilizados softwares de códigos abertos Arduino IDE, Visual Studio Code (VScode) e outras plataformas livres, visando a acessibilidade tecnológica e a viabilidade do projeto. A Tabela 2, apresentada a seguir, detalha os custos estimados dos componentes e ferramentas no protótipo.

Tabela 2 - Dados orçamentários dos materiais

<b>Componentes</b>	<b>Valor estimado (R\$)</b>	<b>Fonte /observação</b>
--------------------	---------------------------------	------------------------------

Arduino uno original/compatível	R\$ 60,00 – R\$ 90,00	Mercado livre /Smartkits
Servo motor SG90	R\$ 15,00 – R\$ 25,00	Mercado livre /Smartkits
Garra robótica (acrílico, mdf ou 3D impresso)	R\$ 40,00 – R\$ 70,00	Feito sob encomenda / arquivo stl
Webcam HD (mín. 720p)	R\$ 60,00 – R\$ 120,00	Kabum / Amazon
Sensor EEG (NeuroSky MindWave)	R\$ 800,00 – R\$ 1200,00	Aliexpress
Protoboard + cabos + fonte 5V	R\$ 25,00 – R\$ 40,00	Kits básico do Arduino
Kinect Microsoft	R\$112,00 – R\$ 349,00	Mercado livre / Amazon
<b>Total estimado (sem EEG)</b>	R\$ 312,00 - 694,00	
<b>Total estimado (com EEG)</b>	R\$ 1.112,00-1.894,00	

#### 5.7.4 Modularidade e Expansão

A arquitetura do sistema precisa ser modular, a fim de permitir reconfiguração, substituição de elementos, integração de novas tecnologias assistivas.

- O trabalho será dividido em três módulos principais:

1. Entrada (rastreamento ocular / EEG);
2. Processamento lógico (software);
3. Saída (garra robótica).

O código-fonte deve ser estruturado para ser construído em funções / módulos distintos que podem ser chamados ou reutilizados, visando clareza e preservação.

- Deve permitir crescimento futuro, por exemplo:
  - Reconhecimento de voz;
  - Integração com sensores de movimento (giroscópio, acelerômetro);
  - Operação de cadeiras de rodas, braços articulados, etc.

"A modularidade é fundamental para a escalabilidade e durabilidade dos sistemas assistivos abertos." (Grandini, 2017)

### **5.7.5 Ética, Humanização e Experiência do Usuário**

Por definição, a tecnologia assistiva deve ser *ferramentas* que capacitam o humano, em vez de automatizar as coisas. Assim, o projeto aborda o aspecto ético e afetivo do contato.

- A pessoa com deficiência não deve ser um objeto para a aplicação, o ator no processo de design e verificação tanto quanto possível.
- A interação com o sistema deve criar confiança, valor, autonomia, prevenir frustração, esforços extras desnecessários ou sentimentos de dependência.

- Decisões baseadas em informações visuais, auditivas e físicas devem honrar o conforto, privacidade e dignidade do usuário.
- Testes com humanos devem aderir aos padrões de pesquisa eticamente declarados para envolvimento de humanos (Resolução CNS 466/2012), cobrindo consentimento livre e esclarecido, anonimato e não exposição.

## **5.8 Aspectos Éticos**

O estudo está em conformidade com a Resolução CNS 466/2012 que estabelece as normas que regulam a pesquisa envolvendo seres humanos no Brasil. Foram previstos:

- Um termo de consentimento livre e esclarecido dos participantes;
- Garantia de confidencialidade dos dados pessoais;
- Simulação de testes onde participantes com deficiência não estão disponíveis;
- Sem riscos físicos ou psicológicos.

## **5.9 Desenvolvimento Iterativo: Ciclos de Prototipagem e Refinamento do Sistema**

O desenvolvimento do Sistema Gestus seguiu uma abordagem de Design Centrado no Usuário, onde a evolução da interface e dos algoritmos foi pautada por ciclos experimentais de testes e correções. Esta fase, ocorrida no último quadrimestre do projeto, foi fundamental para transpor as barreiras técnicas da captura ocular e garantir a usabilidade da tecnologia assistiva.

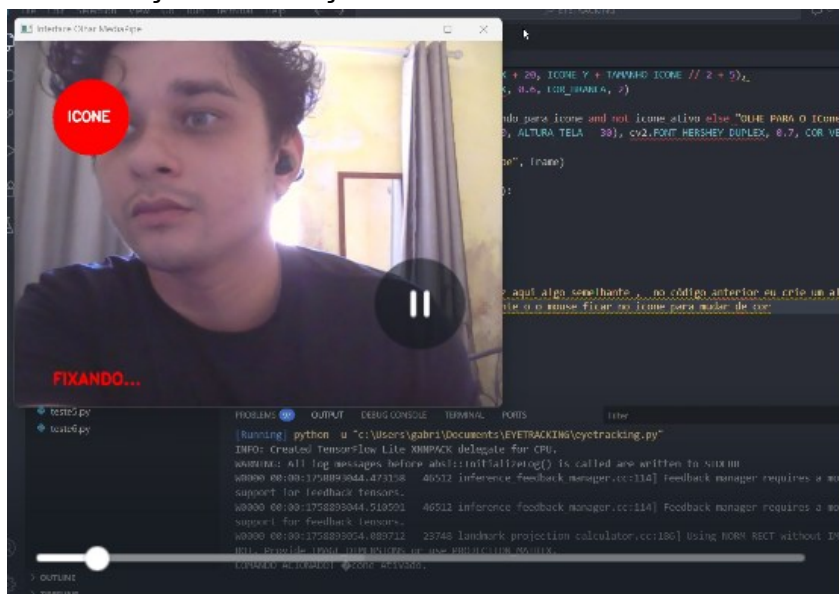
### **5.9.1. Fase 1 (setembro): Desafios de Rastreamento Inicial**

Os primeiros testes realizados em setembro focaram na calibração básica do algoritmo de eye tracking via MediaPipe. Nesta etapa, identificou-se que a captura

do movimento do olho apresentava instabilidades severas. O sistema frequentemente perdia o foco do olhar (pupila), exigindo que o usuário realizasse movimentos compensatórios com a cabeça para que o ícone fosse ativado.

Outro problema detectado foi o acionamento indesejado de ícones por proximidade, o que gerava uma carga cognitiva elevada e frustração, uma vez que o cursor não possuía a precisão necessária para distinguir entre a observação de um elemento e a intenção de selecioná-lo.

Figura 10 - Interface inicial de testes em setembro, evidenciando a instabilidade do rastreo ocular e a necessidade de inclinação da cabeça do usuário



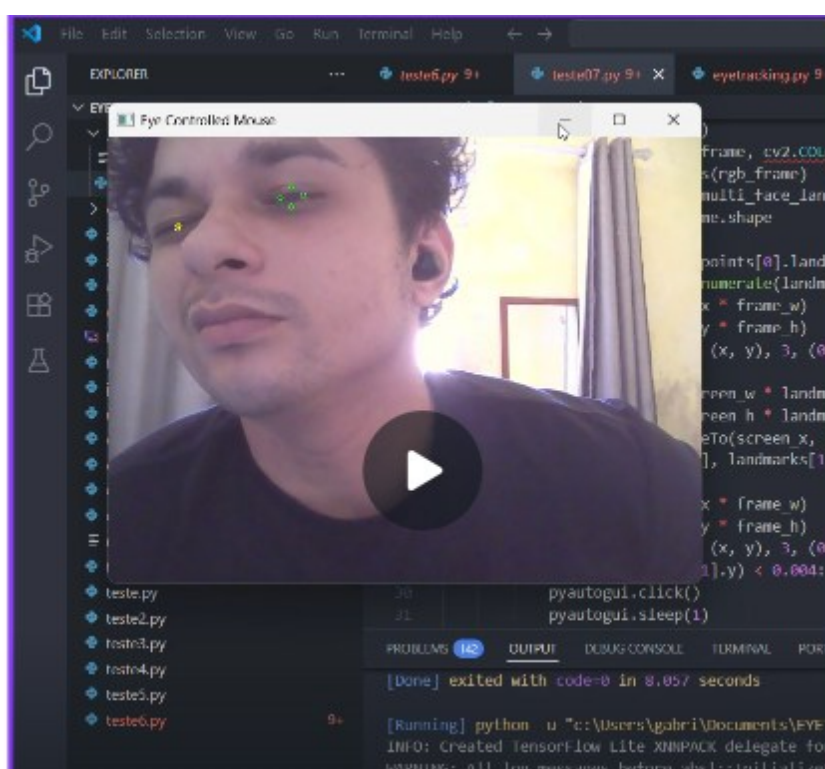
Fonte: elaborada pelo autor(2025)

### 5.9.2. Fase 2 (outubro): Otimização Algorítmica e Filtro de Kalman

No segundo ciclo de testes, em outubro, as dificuldades com a precisão ocular ainda persistiam, apresentando uma acurácia classificada como média-baixa devido ao ruído natural dos dados captados pela webcam (o fenômeno de jitter). Para mitigar essa oscilação sem exigir imobilidade total do usuário, foi implementado o Filtro de Kalman.

Este algoritmo matemático permitiu suavizar as coordenadas do olhar em tempo real, realizando uma predição da posição da pupila e filtrando as variações bruscas causadas pela iluminação ou micro movimentos. Embora a precisão tenha melhorado significativamente, o sistema ainda apresentava dependência de movimentos cervicais, o que levou à necessidade de uma revisão completa na lógica de interação para o mês seguinte.

Figura 11 - Representação visual dos dados brutos vs. dados filtrados pelo algoritmo de Kalman no rastreamento ocular de outubro



Fonte: elaborada pelo autor(2025)

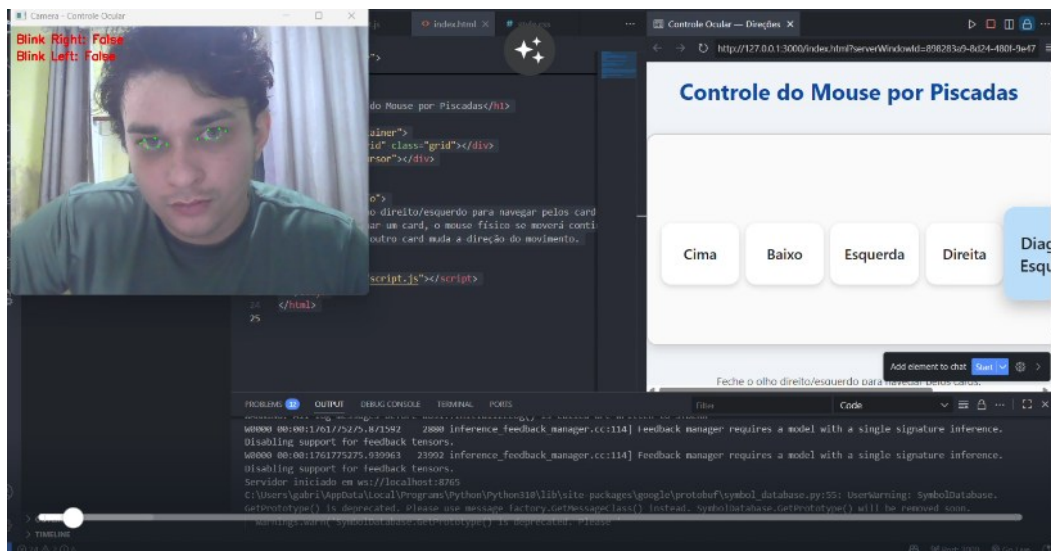
### 5.9.3. Fase 3 (novembro-dezembro): Integração e Solução Final

O estágio final de desenvolvimento (novembro e dezembro) marcou a maturidade do sistema. Houve um ajuste profundo nos algoritmos, permitindo que a seleção ocorresse sem qualquer necessidade de movimentação da cabeça, focando

exclusivamente na região ocular. Retomou-se, de forma aprimorada, o mecanismo de detecção de piscadas para funções específicas, mas a principal inovação foi a adoção da Dinâmica de Cards.

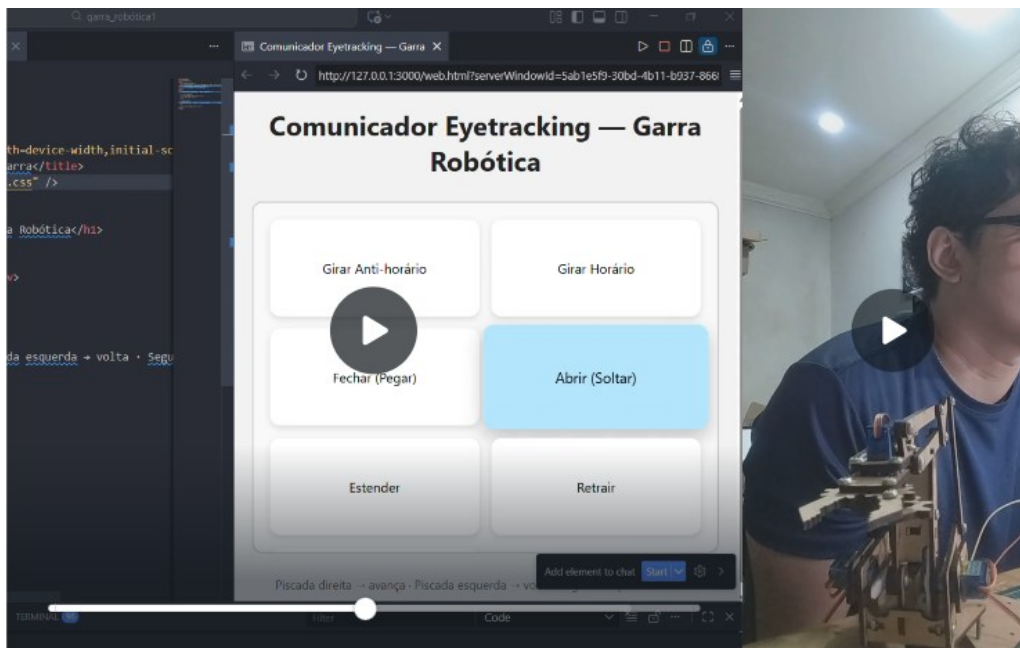
Inspiração na análise de projetos de interface do laboratório (Santos Vieira), a interface foi redesenhada em cartões interativos de grandes dimensões. A seleção passou a funcionar por Dwell Time (tempo de permanência): ao manter o olhar fixo sobre um card por um período pré-determinado, o "click automático" é disparado. Toda essa lógica foi integrada em linguagem Python, permitindo uma comunicação fluida entre a interface gráfica e o hardware. O fechamento desta fase ocorreu com a integração bem-sucedida da garra robótica. O comando selecionado na interface via olhar era processado e enviado ao atuador físico, concretizando o objetivo de mobilidade e autonomia do projeto.

Figura 12 - Interface final do Sistema Gestus em novembro, apresentando a dinâmica de cards e o indicador visual de tempo de seleção (dwell time).



Fonte: elaborada pelo autor(2025)

Figura 13 - Teste de integração final: comando ocular via Python resultando no movimento físico da garra robótica



Fonte: elaborada pelo autor(2025)

## 5.8 Tecnologias Consideradas e Delimitação do Escopo

No processo de investigação e definição das tecnologias assistivas a serem exploradas, foram consideradas abordagens baseadas na captação de sinais eletroencefalográficos (EEG), amplamente discutidas na literatura como alternativas para interação humano-computador em contextos de mobilidade extremamente limitada. Essas tecnologias permitem a interpretação de sinais cerebrais como forma de comando, ampliando o espectro de acessibilidade para usuário sem controle motor voluntário.

Entretanto, apesar de seu potencial, a utilização de sistemas baseados em EEG não foi implementada neste trabalho, em função de fatores com maior complexidade técnica, necessidade de equipamentos específicos de custo elevado, maior tempo de calibração e curva de aprendizado acentuada para usuário. Assim, optou-se por tratar EEG como uma tecnologia de referência e comparação conceitual, servindo como base teórica para discussão de acessibilidade extrema, sem adaptação prática no protótipo desenvolvido. Essa decisão metodológica está alinhada à proposta de trabalho de priorizar soluções viáveis, acessíveis e reproduzíveis,

mantendo o foco no desenvolvimento de um sistema funcional baseado em rastreamento ocular e piscadas, tecnologia que apresenta melhor equilíbrio entre custo, complexidade e usabilidade.

## **6. SISTEMA GESTUS – CONCEPÇÃO E DESENVOLVIMENTO**

A partir da análise das tecnologias assistivas, observou -se que muitas soluções voltadas para pessoas com mobilidade reduzida concentram-se em aspectos específicos, como comunicação alternativa ou controle digital, frequentemente dependendo de hardwares proprietários e alto custo. Embora eficientes em determinados contextos, essas soluções apresentam limitações no que se refere à acessibilidade econômica, à personalização e à integração com o ambiente físico. Diante desse cenário, tornou-se evidente a necessidade de uma solução que articulasse comunicação, controle digital e interação física, respeitando os princípios de baixo custo, modularidade e adaptação ao usuário. Nesse contexto, a solução viável identificada foi a construção do sistema GESTUS.

O sistema Gestus é concebido como uma plataforma de interação assistiva baseada em rastreamento ocular e detecção de piscadas, cujo objetivo central é traduzir a intenção do usuário em ações significativas, tanto no ambiente digital quanto no físico. Diferentemente de soluções fragmentadas, o GESTUS propõe uma abordagem sistêmica, na qual diferentes modos de interação são organizados de forma modular e integrada.

Desde sua concepção, o sistema foi pensado não apenas como um software de comunicação ou controle de interface, mas como sistema híbrido, capaz de se conectar a dispositivos físicos. Nesse contexto, a garra robótica surge como uma extensão natural do conceito do GESTUS, permitindo que comandos gerados pelo olhar sejam materializados em ações concretas no mundo real.

A integração da garra robótica reforça o entendimento do sistema entre o usuário e o ambiente, ampliando sua autonomia funcional e simbólica. Assim, o GESTUS não se limita à mediação digital, mas busca atuar diretamente na relação do

indivíduo com objetos e tarefas cotidianas. Os aspectos relacionados à arquitetura do sistema, ao design modular, à interface e à integração com dispositivos são aprofundados a seguir.

O design desempenha um papel central no desenvolvimento da solução de proposta, não apenas como elemento estético, mas como estrutura organizadora da interação e mediadora entre usuário e a tecnologia. No contexto deste trabalho, o design é compreendido como um processo estratégico de planejamento da informação, das interfaces e das formas de interação, fundamentado em princípios de usabilidade, acessibilidade e clareza funcional.

A diagramação das interfaces foi concebida de modo a favorecer a leitura visual, a previsibilidade das ações e redução da carga cognitiva, aspectos essenciais em sistemas assistivos baseados em rastreamento ocular. A organização espacial dos elementos gráficos, o uso de hierarquia visual e padronização dos componentes contribuem para uma interação mais intuitiva, minimizando erros e esforço do usuário, conforme defendido por Norman (2013) e Rams (2017).

Além disso, a arquitetura da informação foi estruturada a partir da definição clara de funções, tarefas e fluxos de navegação, garantindo que cada bloco de interação apresentasse objetivos específicos e compreensíveis. Essa organização permite que o usuário construa um modelo mental consistente do sistema de aprendizagem e uso contínuo da solução.

## **6.1 Apresentação Geral do Sistema**

O Sistema Gestus foi inicialmente concebido com foco no desenvolvimento de um dispositivo físico de tecnologia assistiva, especificamente uma garra robótica, voltada à ampliação da autonomia de pessoas com mobilidade reduzida. No entanto, ao longo do processo projetual, o escopo do trabalho foi progressivamente ampliado, incorporando um sistema digital de interação ocular baseado em visão computacional, comunicação alternativa e blocos de interação estruturados em cards.

Essa ampliação conceitual e técnica ocorreu a partir da identificação de limites funcionais no uso exclusivo do dispositivo físico e, sobretudo, da influência direta do Projeto Santos Vieira – Tecnologia Assistiva por Comunicação Ocular, coordenado pelo Prof. Dr. Roberto César Cavalcante Vieira. Tal projeto evidenciou o potencial dos sistemas de comunicação ocular baseados em interfaces visuais estruturadas, de baixo custo e com forte embasamento em Comunicação Aumentativa e Alternativa (CAA), servindo como referência metodológica e conceitual para a evolução do Gestus.

## **6.2 Introdução ao Design Modular**

O design modular é uma abordagem projetual que organiza produtos e sistemas a partir de módulos independentes, interligados por interfaces bem definidas, permitindo flexibilidade, personalização e adaptação ao longo do tempo. Segundo Monteiro (2002), o design modular surge como resposta à necessidade de lidar com sistemas complexos, possibilitando a reorganização funcional e formal sem comprometer a integridade do conjunto.

No contexto do presente trabalho, o design modular adotado como estratégia central para o desenvolvimento do sistema GESTUS, uma vez que tecnologias assistivas demandam soluções adaptáveis às diferentes necessidades e limitações dos usuários. A modularidade permite que o sistema seja configurado conforme o perfil do usuário, evitando soluções rígidas e excludentes.

Essa abordagem está alinhada também com o design centrado no usuário, que, segundo Baxter (2011), deve considerar simultaneamente aspectos funcionais, estruturais e de uso, especialmente em projetos que envolvem sistemas interativos. Norman (2013) complementa ao afirmar que sistemas bem projetados reduzem a carga cognitiva e tornam a interação mais previsível e segura

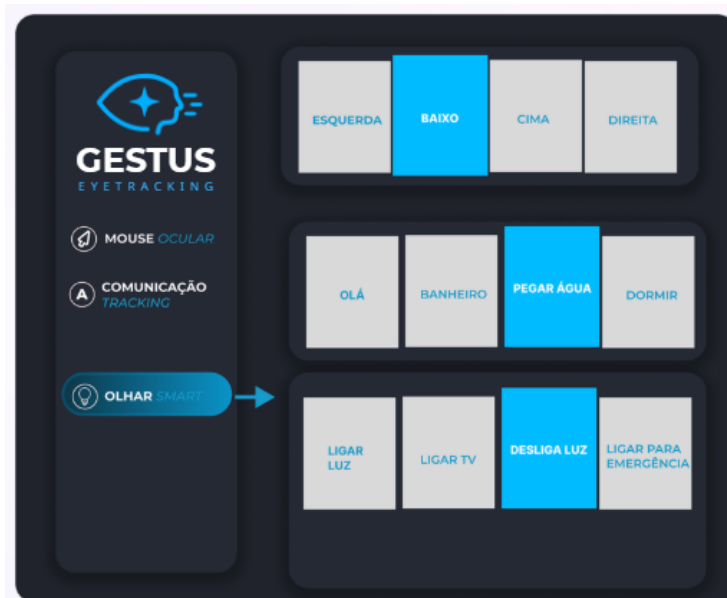
## **6.3 Design Modular aplicado ao Design de interface**

No sistema GESTUS, o design modular não se restringe apenas à arquitetura técnica do software, mas estende-se diretamente ao design de interface.

Cada bloco de interação - Mouse ocular, Comunicador Ocular e Olhar Smart - foi concebido como um módulo funcional independente, com objetivos específicos, porém interligados a uma linguagem visual e interacional comum.

No GESTUS, essas interfaces são representadas pelos *cards*, que funcionam como elementos de mediação entre o usuário e as funções do sistema, essa escolha facilita a compreensão, a navegação e até mesmo a memorização das ações disponíveis. A modularidade aplicada à interface permite reduzir a complexidade visual, uma vez que o usuário interage apenas com módulos necessários à tarefa em execução, reforçando princípios de usabilidade e acessibilidade. A figura 10 ilustra a interface do sistema, com o foco na objetividade, funcionalidade e facilidade de controle e acesso

Figura 14 - interface gráfica do sistema Gestus



Fonte- elaborada pelo autor.

### 6.3 Arquitetura da Informação do Sistema GESTUS

A arquitetura da informação do sistema Gestus foi estruturada com base na organização clara de funções, tarefas e fluxos de navegação, considerando as limitações motoras e cognitivas do usuário. A proposta, que está dentro das diretrizes,

é desenvolver um sistema “funcionalmente acessível “, o que compete introduzir um sistema fácil e objetivo, com baixa complexidade operacional.

#### **Funções do sistema:**

- Controle do cursor do mouse
- Comunicação alternativa por meio de síntese de voz
- Controle de aplicações e funções do computador
- Controle de dispositivos físicos

#### **Tarefas do usuário:**

- Navegar entre cards
- Selecionar comandos
- Confirmar ações
- Retornar ou cancelar operações

No Gestus, a separação das funções em blocos distintos permite que o usuário compreenda o sistema de forma progressiva sem sobrecarga cognitiva. O Mapa de navegação do sistema GESTUS segue uma estrutura linear e cíclica, adequada à interação por piscadas. O usuário percorre os cards sequencialmente, avançando ou retrocedendo conforme o comando ocular, até alcançar a função desejada.

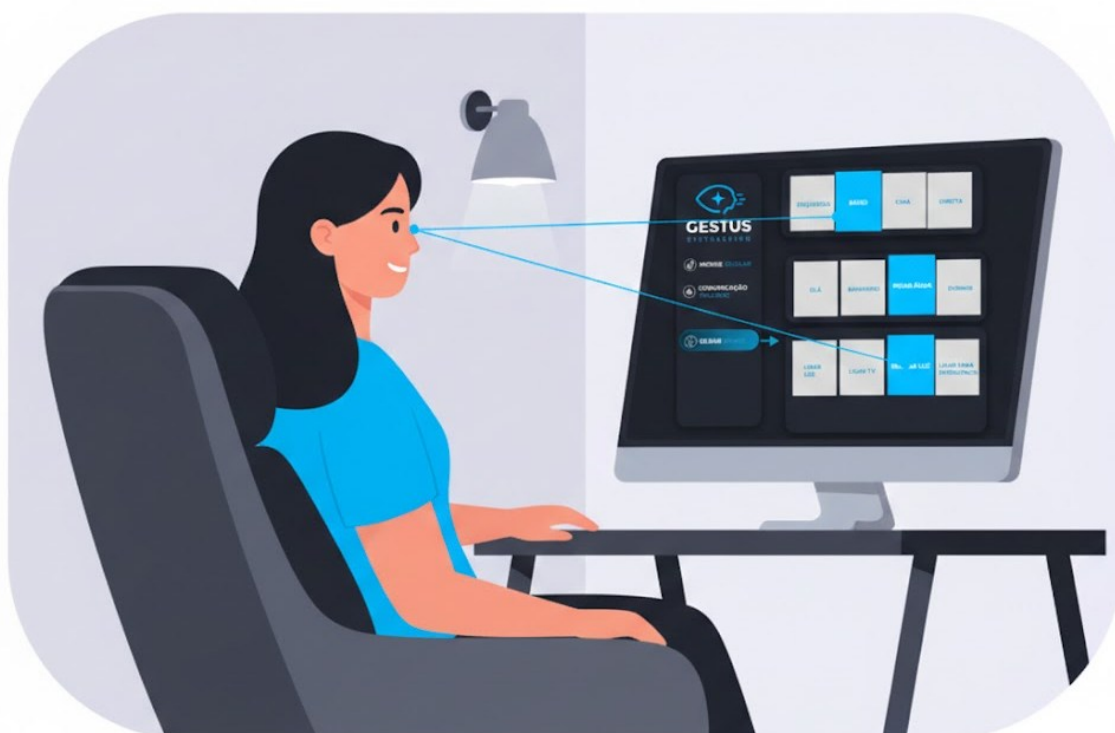
#### **Esse modelo foi escolhido por:**

- Evitar menus profundos
- Reduzir ambiguidades
- Facilitar o aprendizado
- Manter consistência entre os módulos

Norman (2013), destaca que sistemas previsíveis aumentam a confiabilidade do usuário. No contexto assistivo, essa previsibilidade é essencial para garantir

autonomia e segurança de uso. Com o aprendizado guiado, que o próprio sistema fornece, em que o usuário precisa somente de poucos comandos para interagir com os *cards* e acionar as funções, diminuindo a complexidade, como também a descarga cognitiva. A figura 11 ilustra como utilizar o sistema.

Figura 15 - Interface do sistema Gestus e exemplo de usuário utilizando em tempo real



Fonte - elaborada por IA

#### 6.4 Jornada do Usuário

A jornada do usuário no sistema GESTUS descreve o percurso completo desde o primeiro contato com o sistema até a execução de uma ação. Essa jornada foi projetada para ser curta, repetível e de baixo esforço físico, respeitando as capacidades residuais do usuário

De forma indireta, Baxter (2011) destaca que sistemas centrados no usuário devem organizar suas etapas de uso de modo a reduzir frustrações e facilitar o aprendizado progressivo. No GESTUS, essa premissa se traduz em uma sequência de ações simples e consistentes.

A jornada típica do usuário ocorre da seguinte forma:

1. Posicionamento em frente à câmera
2. Reconhecimento do olhar pelo sistema
3. Navegação entre os *cards*
4. Seleção do comando desejado
5. Confirmação da ação
6. Execução das respostas do sistema

Essas etapas do processo do usuário ao interagir com o sistema, são fundamentadas na construção de “bons sistemas”, abordada por Norman (2013), em que ele ressalta a importância de ações visíveis e previsíveis como uma forma de reduzir a necessidade de memorização por parte do usuário. A partir dessa afirmação, o sistema Gestus, estabelece a dinâmica dos *cards* como uma decisão visual e de *feedback* constante durante toda a jornada de uso.

Essa redução da complexidade, torna-se uma ferramenta essencial para interação dos blocos do sistema contribuindo com a previsibilidade e a confiança do usuário. Essa jornada simples, aumenta a eficiência do uso e reduz a ocorrência de erros, especialmente em sistemas voltados para públicos com limitações motoras. Além disso, a jornada do usuário foi pensada para permitir pausas, correções e retornos, sem penalizar o usuário, reconhecendo a possibilidade de fadiga ocular e variações na atenção ao longo do uso contínuo do sistema.

## **6.5 integrações de dispositivos Físicos**

De acordo com Monteiro em sua tese de mestrado (2002), sistemas modulares permitem a incorporação de novos módulos desde que existam interfaces

claras entre as partes. O sistema GESTUS, essa lógica é aplicada tanto aos módulos digitais quanto à garra robótica, compreendida como módulo físico conectado à arquitetura do sistema. A modularidade garante que a garra robótica possa ser utilizada, removida ou substituída sem comprometer o funcionamento dos demais blocos, reforçando a flexibilidade e a sustentabilidade do projeto.

Do ponto de vista do design, esse modelo físico é compreendido como um dispositivo assistivo externo, integrado ao sistema por meio da mesma lógica de interação baseada em rastreamento ocular e piscadas. Essa decisão elimina a necessidade de múltiplos sistemas de controle, mantendo a integridade e previsibilidade da experiência do usuário.

Em que, mantém a coerência entre o sistema, e reduz a carga cognitiva e o esforço de aprendizado. No GESTUS, essa coerência se manifesta na utilização da mesma linguagem de interação para os módulos tanto físicos quanto digitais. Além disso, relaciona o aspecto funcional, representando a materialização entre a intenção do usuário, transformando o olhar — frequentemente um dos poucos movimentos preservados, principalmente, para pessoas com Esclerose Lateral Amiotrófica (ELA) — em ação concreta. Essa transformação reforça o caráter inclusivo do sistema e amplia seu potencial de impacto social.

## **6.6 Processo de Desenvolvimento e Principais Impasses Projetuais**

O desenvolvimento do Sistema Gestus ocorreu de forma iterativa e incremental, passando por diferentes níveis de prototipação funcional. Inicialmente, a lógica de interação era baseada exclusivamente na movimentação contínua do olhar, em que o deslocamento da pupila controlava a seleção de elementos na interface. Embora funcional do ponto de vista técnico, essa abordagem revelou um impasse significativo: o cansaço ocular excessivo após períodos prolongados de uso.

Esse problema está diretamente relacionado à alta frequência de micro ajustes do olhar humano e à instabilidade natural da captura por webcam comum, especialmente em ambientes com variações de iluminação. Nos primeiros testes,

observou-se ruído elevado nos dados de rastreamento ocular, resultando em seleções involuntárias e baixa previsibilidade da interação.

Para mitigar esses efeitos, foram realizados testes com técnicas de suavização de sinal, culminando na adoção de filtros estatísticos, com destaque para o Filtro de Kalman. Esse filtro permitiu reduzir oscilações abruptas na estimativa da posição ocular, promovendo maior estabilidade na interação. Contudo, mesmo com a filtragem, o esforço visual contínuo ainda se mostrava inadequado para usuários com limitações motoras ou neurológicas.

A partir desse impasse, a lógica de interação foi reformulada, substituindo o movimento contínuo do olhar por piscadas intencionais como mecanismo principal de comando. Essa decisão reduziu significativamente o esforço cognitivo e visual, além de aumentar a confiabilidade da detecção de intenções do usuário.

## **6.7 Influência do Projeto Santos Vieira e a Adoção dos Cards de Interação**

O Projeto Santos Vieira teve papel central na consolidação da solução adotada no Sistema Gestus. Desenvolvido no contexto hospitalar, o projeto propõe o uso de interfaces baseadas em pranchas e elementos visuais organizados, permitindo que pacientes impossibilitados de falar realizem seleções por meio do olhar.

Inspirado nessa abordagem, o Gestus incorporou a lógica de cards de interação, em que cada card representa uma ação, comando ou conteúdo comunicacional. Essa estrutura apresenta vantagens técnicas e cognitivas relevantes:

- Redução da carga cognitiva do usuário;
- Padronização da interface;
- Facilidade de escalabilidade do sistema;
- Integração direta com ações digitais e físicas.

A utilização de cards também dialoga diretamente com princípios da Comunicação Aumentativa e Alternativa, amplamente discutidos no Projeto Santos Vieira, reforçando o caráter inclusivo e acessível do Gestus.

## **6.8 Integração dos Blocos de Interação e Paradigma IoT**

Do ponto de vista arquitetural, o Sistema Gestus pode ser compreendido como um ecossistema de blocos de interação, funcionando de maneira semelhante a um sistema de Internet das Coisas (IoT). Nesse paradigma, diferentes módulos — comunicador ocular, mouse ocular, controle de aplicações e dispositivos físicos — comunicam-se por meio de protocolos de rede.

A implementação técnica utiliza uma arquitetura cliente-servidor baseada em WebSockets, em que:

- O servidor (Python) realiza o processamento de visão computacional com OpenCV e MediaPipe Face Mesh;
- Os eventos de piscada são interpretados e transmitidos em tempo real;
- O cliente web (HTML, CSS e JavaScript) recebe esses eventos e atualiza a interface de cards;
- Comandos podem ser enviados de volta ao servidor para acionar aplicações do sistema operacional ou dispositivos externos, como a garra robótica.

## **6.9 Garra Robótica no Contexto do Sistema Gestus**

A garra robótica permanece como elemento central do Gestus no contexto de equipamentos físicos assistivos. Sua função é materializar, no mundo físico, as decisões tomadas pelo usuário por meio da interface ocular.

Embora o foco deste memorial esteja no sistema integrado, é fundamental destacar que a garra robótica foi projetada considerando:

- Limitações motoras severas dos usuários;
- Necessidade de comandos discretos e confiáveis;
- Integração com sistemas digitais por meio de sinais computacionais.

Nesse sentido, a garra não é tratada como um elemento isolado, mas como um bloco de interação física, coerente com a lógica modular inspirada no Projeto Santos Vieira.

#### **6.10 Parceria interdisciplinar e divisão de contribuições**

O desenvolvimento do sistema Gestus ocorreu por meio de uma parceria interdisciplinar envolvendo estudantes e docentes das áreas de Design e Engenharia de Computação. A colaboração contou com orientação do Professor Edilson do curso de Engenharia de computação e do bolsista Luís Carlos integrante do projeto. O objetivo comum da equipe foi o desenvolvimento de um sistema de comunicação assistiva, capaz de ampliar as possibilidades de interação de pessoas com mobilidade reduzida por meio de interfaces alternativas.

No que se refere à divisão das contribuições, o autor deste trabalho, Gabriel Eduardo Oliveira da Silva, concentrou sua atuação principalmente, no desenvolvimento da interface do sistema, na concepção dos *cards* de interação, na organização da arquitetura modular, no design modular e na idealização da garra robótica enquanto elemento simbólico e funcional de interação física. O outro bolsista ficou responsável pela investigação técnica relacionada à movimentação do mouse ocular, ao estudo diferentes formas de interação digital e à análise de métodos computacionais voltados à tradução dos comandos visuais em ações sistema.

Ao longo de todo o processo, a interação entre os bolsistas e os professores coordenadores ocorreu de forma contínua por meio de reuniões, testes, discussões e

ajustes iterativos, caracterizando o projeto como um trabalho coletivo, interdisciplinar e colaborativo. Além disso, o desenvolvimento do sistema Gestus ocorreu no contexto do Programa de Extensão PROTECS — Programa de tecnologias, Inovação e inclusão da Universidade Federal do Ceará — iniciativa voltada à promoção de soluções com impacto social, especialmente no campo da acessibilidade e da inclusão de pessoas com deficiência.

O PROTECS atua como um espaço de integração entre ensino, pesquisa e extensão, reunindo estudantes e professores de diferentes áreas com o objetivo de desenvolver tecnologias assistivas capazes de transformar realidades e ampliar a autonomia dos usuários.

## **7. FUNCIONAMENTO DO SISTEMA GESTUS**

O sistema Gestus, foi concebido a partir de uma arquitetura computacional orientada à interação assistiva em tempo real, combinando técnicas de visão computacional, comunicação assíncrona e interfaces gráficas baseadas em tecnologias web. A escolha dessa arquitetura decorre da necessidade de garantir baixa latência, robustez na detecção de eventos oculares e flexibilidade para expansão futura, especialmente considerando usuários com limitações motoras severas.

A organização do sistema adota princípios do design modular, nos quais cada camada desempenha uma função bem definida, reduzindo acoplamentos e facilitando adaptações do sistema para novos blocos de interação, como comunicadores, controle de dispositivos ou interfaces híbridas com EEG.

### **7.1 Camada de Aquisição de Dados Visuais**

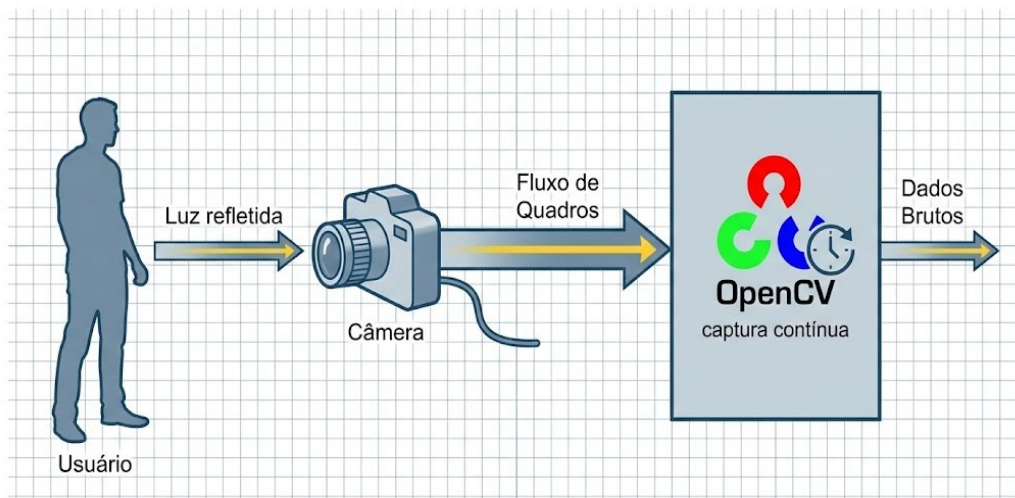
A camada de aquisição é responsável por capturar continuamente as imagens do usuário por meio de uma câmera convencional. Essa função é desempenhada pela biblioteca OpenCV, amplamente utilizada em sistemas de visão computacional por sua estabilidade, desempenho e compatibilidade multiplataforma.

o uso do OpenCV permite:

- Captura contínua de quadros em tempo real
- Conversão eficiente entre espaços de cores
- Integração direta com bibliotecas de aprendizado de máquina

A decisão por uma câmera comum, aliada ao OpenCV, reforça o caráter acessível e replicável do Gestus, evitando dependência de hardware proprietário ou dispositivos especializados de alto custo.

Figura 16 - Fluxo de interação biblioteca OpenCV



Fonte - elaborada por IA

## 7.2 Camada de Análise Facial e Rastreamento Ocular

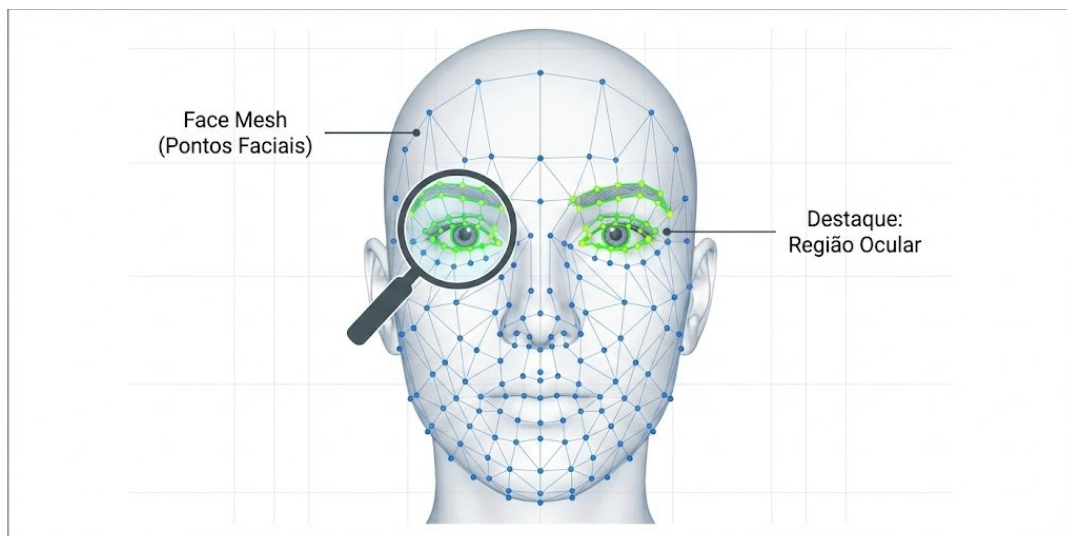
A interpretação dos dados visuais é realizada por meio da biblioteca MediaPipe, especificamente utilizando o modelo *Face Mesh*. Essa tecnologia emprega redes neurais treinadas para identificar centenas de pontos faciais com alta precisão, mesmo em condições de iluminação e posicionamento variáveis.

No Gestus, o Face Mesh é utilizado para identificar pontos específicos da região ocular, permitindo mapear geometricamente:

- A abertura dos olhos;
- A relação entre pálpebras;
- Variações mínimas associadas ao piscar voluntário.

A escolha do MediaPipe está diretamente relacionada à sua capacidade de operar em tempo real, sem necessidade de treinamento local de modelos, o que reduz drasticamente a complexidade computacional e torna o sistema viável em computadores pessoais. Essa camada representa um ponto central de convergência entre engenharia e design de interação, pois a confiabilidade da leitura ocular impacta diretamente a experiência do usuário.

Figura 17 - MediaPipe , captação de pontos faciais



Fonte - elaborada por IA

### 7.3 Processamento Matemático e Inferência de Piscadas

A partir dos pontos faciais detectados, o sistema realiza cálculos geométricos para inferir o estado ocular. Para isso, é aplicada a métrica conhecida como Eye Aspect Ratio (EAR), amplamente adotada em pesquisa de rastreamento ocular e fadiga visual. Os cálculos são realizados com auxílio da biblioteca Numpy, responsável por operações matemáticas vetoriais de alta performance. O uso dessa biblioteca garante precisão numérica e desempenho adequado por análise contínua de dados. Entretanto, a simples variação do EAR não é suficiente para garantir uma

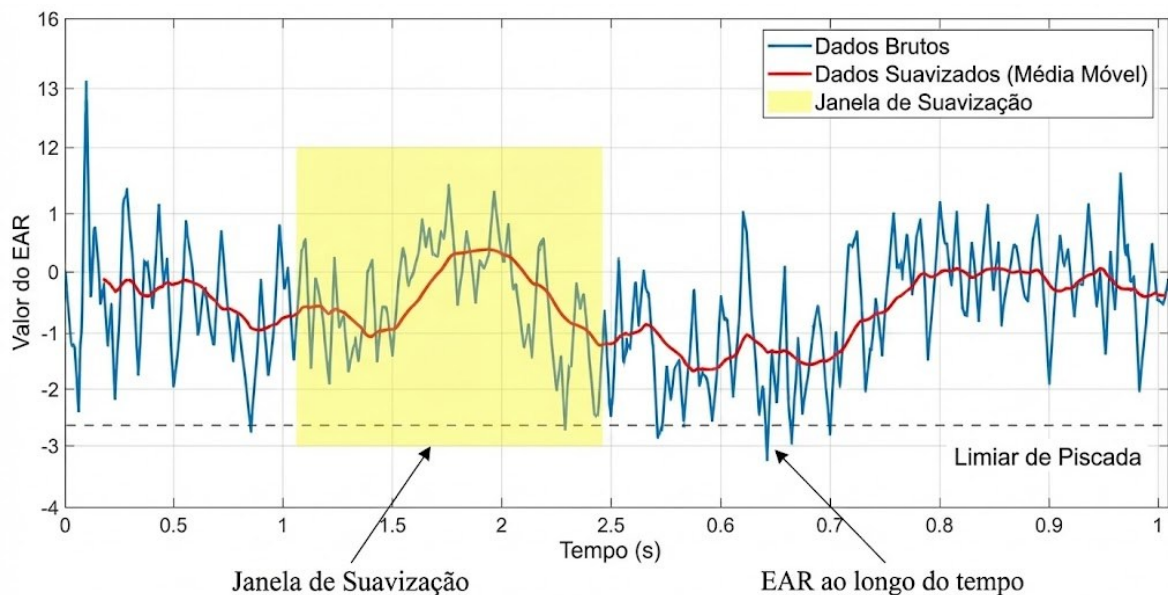
interação confiável. Piscadas involuntárias, microexpressões faciais ou ruídos visuais poderiam gerar falsas ativações. Para mitigar esse problema, o sistema utiliza buffers temporais baseados em estruturas tipo deque .

Esses buffers permitem:

- Analisar múltiplos quadros consecutivos;
- Confirmar padrões consistentes de piscadas
- Reduzir drasticamente falsos positivos

Essa decisão técnica reflete uma preocupação projetual com segurança cognitiva evitando frustrações e ações não intencionais por parte do usuário.

Figura 18 - Filtro de Kalman



Fonte - elaborada pelo autor

## 7.4 Comunicação Assíncrona e Sincronização de Eventos

Uma vez identificados os eventos de piscadas, esses dados precisam ser transmitidos para a interface gráfica de uma forma rápida e confiável. Para isso o Gestus utiliza a comunicação assíncrona baseada em WebSockets, implementada com biblioteca *WebSockets* em Python.

Essa abordagem permite que o sistema opere de forma reativa, mantendo uma conexão persistente entre o módulo de processamento visual e a interface. O uso de WebSockets reduz latência, evita sobrecarga de requisições repetidas e garante sincronização contínua entre percepção e resposta.

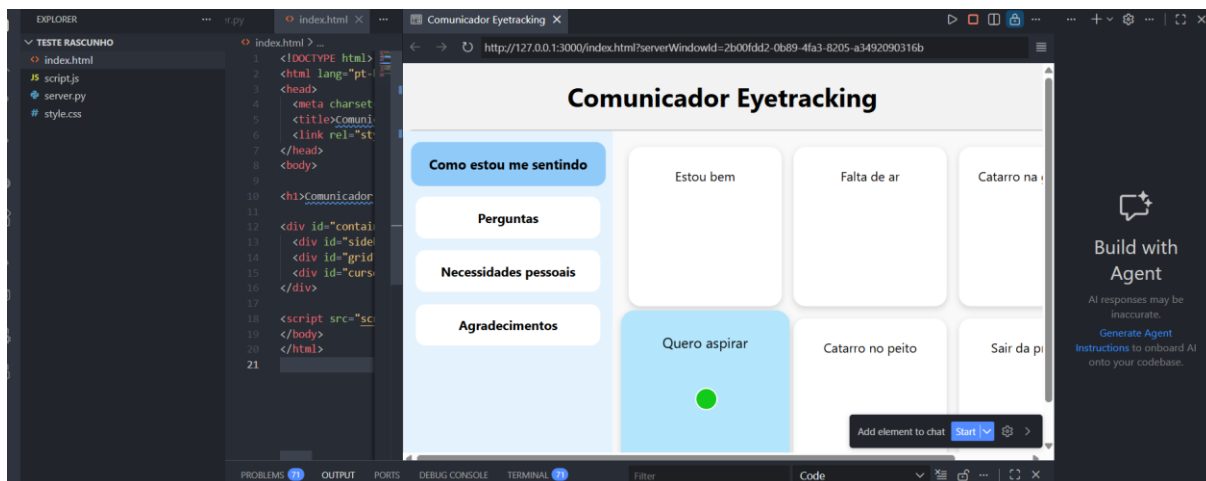
Os dados são transmitidos em formato JSON, o que torna o sistema:

- Independente de linguagem da interface;
- Fácil de estender;
- Compatível com múltiplas plataformas.

## 7.5 Camada de Interface e Feedback Multissensorial

A camada de interface do Gestus é desenvolvida com HTML, CSS e JavaScript, garantindo portabilidade e adaptações e a diferentes dispositivos. A organização visual da interface utiliza *cards*, disposto em um frade modular, permitindo, escalabilidade do conteúdo sem comprometer a legibilidade. O JavaScript desempenha papel central na interpretação dos eventos recebidos, transformando piscadas em ações de navegação e seleção. O uso de mecanismos como *dwell time* reforça a intencionalidade da interação, reduzindo erros e aumentando a previsibilidade. Além do feedback visual, o sistema incorpora um feedback auditivo por meio da Web Speech API, permitindo que o conteúdo seja verbalizado.

Figura 19 - print da tela do bloco de interação Comunicador Ocular utilizando tabelas das necessidades



Fonte - elaborada pelo autor

A arquitetura do Gestus demonstra que soluções assistivas eficazes não dependem exclusivamente de tecnologias complexas ou proprietárias, mas sim de integrações bem planejadas, decisões projetuais conscientes e foco na experiência do usuário. O uso de bibliotecas consolidadas, aliado a uma estrutura modular, possibilita que o sistema seja adaptado para novos blocos funcionais, como controle de dispositivos físicos, integração com EEG ou expansão para ambientes imersivos.

## **8. LÓGICA DE INTERAÇÃO POR PISCADAS E NAVEGAÇÃO NO SISTEMA GESTUS**

A lógica de interação do sistema Gestus foi projetada para substituir completamente dispositivos tradicionais de entrada — como mouse e teclado — por eventos oculares intencionais, respeitando limitações motoras severa e reduzindo a carga cognitiva do usuário. Diferentemente de sistemas baseados em rastreamento contínuo do olhar, o Gestus utiliza piscadas direcionais como principal mecanismo de controle, priorizando robustez, e simplicidade de uso.

Essa decisão está diretamente alinhada aos princípios do Design Centrado no Usuário (Norman, 2013), ao considerar não apenas a capacidade técnica do sistema, mas sobretudo as condições reais de uso por pessoas com mobilidade reduzida. O desenvolvimento do sistema *Gestus* foi realizado no ambiente Visual Studio Code (*VS Code*), editor de código-fonte multiplataforma desenvolvido pela Microsoft. O VS Code foi utilizado como principal ambiente de programação e testes do projeto, permitindo a emulação e execução integrada dos códigos em Python, JavaScript, HTML e CSS.

A ferramenta possibilitou a depuração (*debug*), organização modular dos arquivos e integração com bibliotecas externas como OpenCV, MediaPipe e WebSockets, favorecendo a prototipagem rápida e os testes iterativos do sistema de rastreamento ocular e comunicação assistiva. Os principais trechos de código responsáveis pela detecção de piscadas, navegação por *dwell time* e estrutura da interface encontram-se nos Apêndices A, B, C e D.

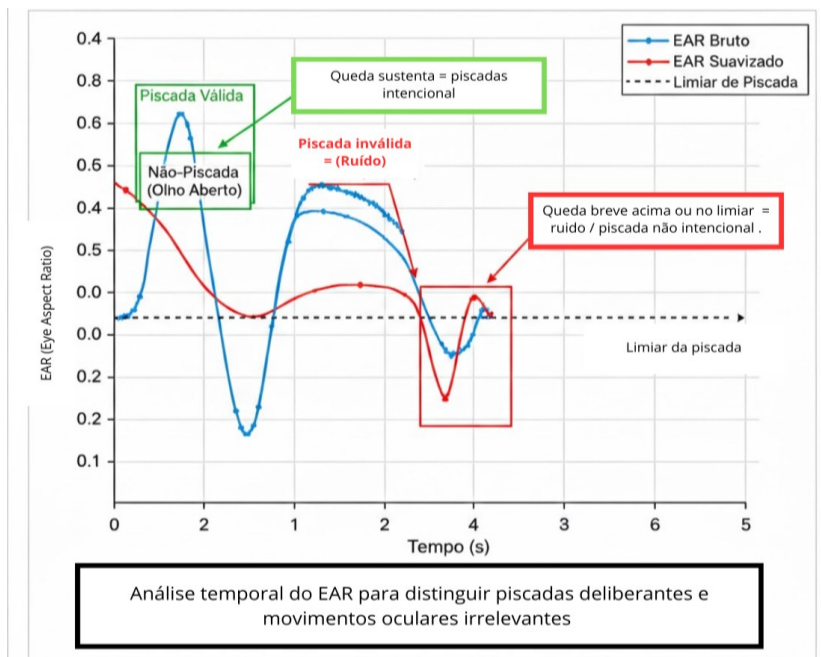
## 8.1 Piscadas como Evento de Entrada Intencional

No Gestus, a piscada não é interpretada como um Simples fechamento ocular, mas como um evento computacional validado, construído a partir de múltiplos critérios técnicos:

- Detecção geométrica de abertura ocular;
- Análise temporal da duração da piscada;
- Confirmação por repetição em janelas sucessivas de tempo

A utilização da métrica Eye Aspect Ratio (EAR), permite distinguir piscadas voluntário de movimentos involuntários, como micro fechamentos naturais dos olhos. No entanto, reconhecendo as limitações nessa métrica isolada, o sistema implementa a camada adicional de verificação baseada em buffers temporais. Essa abordagem reduz significativamente falsos positivos, garantindo que apenas piscadas com padrão consistente sejam reconhecidas como comandos válidos.

Figura 20 - Gráfico de análise de suavização



Fonte - elaborada pelo autor

## 8.2 Diferenciação Direcional: Piscada Direita e Piscada Esquerda

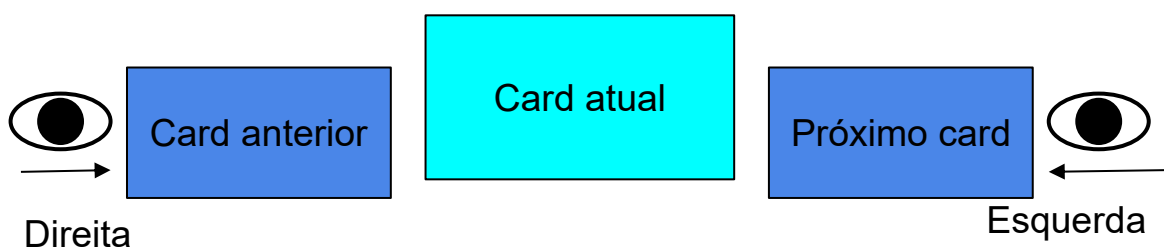
Um dos diferenciais do Gestus é a distinção entre piscadas do olho direito e do olho esquerdo, permitindo a criação de uma navegação bidirecional sem esforço físico adicional.

Essa lógica possibilita:

- Piscada direta - avançar elementos de interface
- Piscada esquerda -retroceder elementos de interface

A separação direcional amplia o vocabulário de interação sem aumentar a complexidade do sistema, mantendo a coerência mental do usuário. Tal estratégia dialoga com os princípios de mapeamento natural propostos por Norman (2013), nos quais ações e respostas devem possuir relação intuitiva. Do ponto de vista técnico, essa diferenciação é possível graças à análise independente dos conjuntos de landmarks oculares tratados de forma paralela dentro do sistema.

Figura 21 - fluxo de movimentação dos cards



Fonte - elaborada pelo autor

## 8.3 Navegação Linear e Controle Cognitivo

A navegação no Gestus adota um modelo Linear e sequencial no qual os elementos da interface são percorridos uma a um, no qual os elementos da interface são percorridos um a um. Essa decisão projetual evita saltos imprevisíveis e reduz o esforço cognitivo, especialmente importante para usuários com limitações motoras

associadas à fadiga mental. Cada elemento da interface é tratado como um estado ativo, visualmente destacado, garantindo que o usuário sempre saiba:

- Onde está;
- Qual ação está preste a executar;
- Como voltar ou avançar.

Esse modelo reforça a previsibilidade do sistema e está alinhada às de usabilidade e acessibilidade, conforme apontado por Baxter (2011), ao defender interfaces claras, consistentes e de fácil aprendizado.

#### **8.4 Tempo de Permanência (Dwell Time) como Confirmação de Ação**

Para evitar ativações acidentais, o Gestus incorpora o conceito de tempo de permanência (dwell time). Nesse modelo, a seleção de um modelo ocorre apenas quando o usuário permanece no mesmo item por um intervalo de tempo predefinido. Essa estratégia:

- Elimina a necessidade de cliques físicos
- Funciona como mecanismo de confirmação implícita
- Reduz erros operacionais.

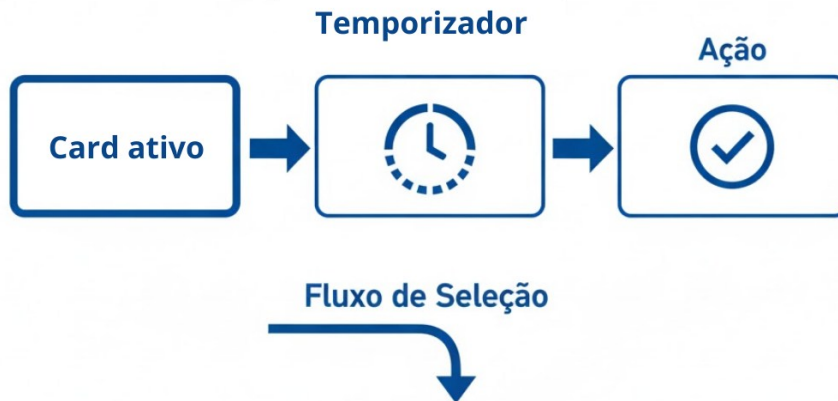
Do ponto de vista do design de interação, o dwell time atua como filtro cognitivo, permitindo que o usuário revise mentalmente a ação antes de executá-la. Essa abordagem também dialoga com os princípios de feedback e controle defendidos por Norman (2013) e por Rams (1970), em que tratou a importância de o design de um produto ser compreensível:

"O bom design torna um produto compreensível. Ele esclarece a estrutura do produto. Melhor ainda, ele pode fazer o produto falar. No seu melhor, é autorreferencial." (LOVELL, 2011, p. 350).

A proposta estabelecida, entra em um plano de alinhamento ao oferecer uma quebra da complexidade do usuário ao sistema. A conduta, funciona com a

seleção do card e a permanência nele. A ativação ocorre quando o tempo de espera ultrapassa um segundo, acionando a função do card estabelecido no sistema.

Figura 22 - Sistema dwell time para ativação dos cards



Fonte - elaborada pelo autor

### 8.5 Feedback Visual e Auditivo com Reforço da Interação

O sistema Gestus utiliza múltiplas camadas de feedback para reforçar a interação e reduzir ambiguidades. O feedback visual ocorre por meio de:

- Destaque do card ativo;
- Movimento do curso virtual;
- Animações suaves de transições.

Paralelamente, o feedback auditivo é acionado após a confirmação da seleção, utilizando síntese de voz para verbalizar o conteúdo escolhido. Essa combinação multimodal amplia a acessibilidade e reforça a compreensão da ação executada. Nisso trabalhar com sistemas eficazes que devem fornecer feedback imediato e compreensível, princípio claramente incorporado pelo sistema Gestus.

## 8.6. Relação entre Lógica de Interação e Design Modular

A lógica de interação por piscadas adotada no Sistema Gestus não é concebida apenas como uma solução técnica de sustentar diferentes funcionalidade e contextos de uso. Essa decisão está diretamente relacionada aos princípios do design modular, na qual sistemas complexos são organizados a partir de módulos independentes, porém interconectados, permitindo flexibilidade, escalabilidade e reutilização.

Segundo Monteiro (2010), o design modular caracteriza-se pela decomposição de um sistema em unidades funcionais, autônomas, capazes de operar do conjunto. No contexto do Gestus, essa decomposição ocorre principalmente na separação entre:

- Lógica de interação (piscadas, navegação, confirmação);
- Função do Módulo (comunicação, controle de mouse, automação);
- Interface gráfica (cards, layout, feedback visual e auditivo).

Essa separação garante que a lógica de piscadas funcione como um módulo-base, reutilizável, em diferentes blocos de interação, sem necessidade de reestruturação do sistema como um todo.

No Gestus, a modularidade não se restringe à organização do código, mas se manifesta como uma decisão projetual de design de interfaces. A lógica de navegação por piscadas permanece constante, enquanto os conteúdos, ações e respostas do sistema variam conforme o módulo ativo. Essa abordagem e Baxter (2011), ao defender sistemas bem projetados devem apresentar consistência funcional, permitindo ao usuário transferir aprendizados entre diferentes partes do produto. Assim, uma vez compreendida a lógica de interação do Gestus, o usuário consegue operar qualquer bloco do sistema sem necessidade de reaprendizado

Do ponto de vista cognitivo, essa consistência reduz a carga mental e aumenta a confiança do usuário, aspectos fundamentais em tecnologias assistivas voltadas a indivíduos com limitações severas.

A Lógica de interação por piscadas foi estruturada de forma a não depender do conteúdo específico no apresentado na interface. Isso possibilita que novos módulos sejam incorporados ao Gestus, como:

- Controle de dispositivos físicos (e.: garra robótica);
- Interfaces de automação residencial;
- integração futura com sinais EEG;
- Ambientes de comunicação personalizados.

Essa independência funcional reforça o caráter evolutivo do sistema, permitindo que ele seja expandido sem comprometer sua estabilidade. Conforme comentado por Norman (2013), sistemas interativos devem ser projetados para considerar não apenas o seu uso imediato, mas também a sua capacidade de adaptação

“Bons sistemas são aqueles que antecipam mudanças e permitem adaptações sem exigir reformulações completas.” (NORMAN, 2014, p. 248).

No contexto das tecnologias assistivas, a modularidade assume um papel ainda mais relevante. Usuários com deficiência apresentam necessidades altamente individuais, o que torna inviável a adoção de soluções rígidas e monolíticas. O Gestus responde a esse desafio ao permitir que diferentes módulos sejam ativados, desativados ou personalizados conforme o perfil do usuário. Essa abordagem reforça a ideia de que a tecnologia deve se adaptar ao usuário, e não ao contrário, princípio defendido por autores como até então comentados nesta pesquisa. Assim, a lógica modular do Gestus contribui diretamente para sua reprodutibilidade, personalização e inclusão social.

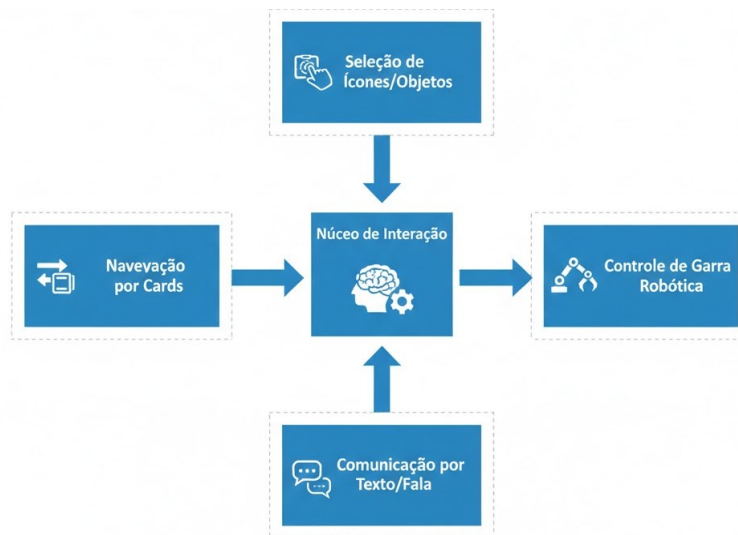
## 9. BLOCOS DE INTERAÇÃO DO SISTEMA GESTUS

A partir da lógica de interação apresentada anteriormente, o sistema Gestus organiza suas funcionalidades em blocos de interação, concebidos segundo os princípios do design modular. Embora compartilhem a mesma base algorítmica de rastreamento ocular e navegação por piscadas, cada bloco foi desenvolvido para atender a necessidades funcionais específicas, explorando diferentes contextos de uso e níveis de complexidade.

Essa organização modular permite que o sistema seja adaptado conforme o perfil do usuário, possibilitando a ativação ou desativação de módulos de acordo com suas capacidades, objetivos e ambiente de utilização, do ponto de vista projetual, essa abordagem reforça a separação entre a lógica de interação, função do sistema e interface favorecendo a escalabilidade, personalização e a reprodutibilidade da solução.

Os blocos apresentados a seguir — Mouse ocular, Comunicador Ocular, Olhar smart e integração com Garra Robótica — demonstram como uma mesma lógica interacional pode ser aplicada a diferentes domínios, ampliando o alcance funcional do Gestus sem comprometer sua usabilidade ou coerência visual.

Figura 23 - Fluxo de interação com algoritmo base



Fonte - imagem gerada por IA

## 9.1 Bloco Mouse ocular

O bloco Mouse Ocular constitui um dos módulos centrais dos sistemas Gestus, sendo responsável por possibilitar o controle do cursor do sistema operacional por meio de piscadas oculares, sem a necessidade de dispositivos físicos de entrada como, mouse ou teclado. Esse bloco foi concebido para usuário com comprometimento motor severo, oferecendo uma alternativa funcional para interação com interfaces convencionais de computadores pessoais. Diferentemente do comunicador ocular, cujo o foco está na expressão verbal mediada por síntese de voz, o Mouse ocular atua diretamente sobre o ambiente computacional do sistema operacional, permitindo navegação, seleção e execução de ações em aplicações externas ao Gestus.

### 9.1.1 Arquitetura Funcional do Bloco Mouse Ocular

O funcionamento do Mouse ocular baseia-se na integração de três camadas principais:

1. **Camada de percepção ocular**, responsável pela detecção de piscadas;
2. **Camada de interpretação de comandos**, que traduz eventos oculares em direções de movimento;
3. **Camada de atuação**, responsável pelo controle efetivo do curso do sistema.

Essa separação reforça o caráter modular e reutilizável do Gestus, permitindo que o mesmo núcleo de percepção ocular seja empregado em diferentes contextos, variando apenas o tipo de ação executada.

### 9.1.2 Bibliotecas Utilizada e Justificativas Técnicas

- **OpenCV (cv2)**

A biblioteca OpenCV é utilizada para a captura contínua de imagens da câmera do usuário. Sua escolha está relacionada à estabilidade, desempenho em

tempo real e ampla adoção em sistemas de visão computacional. No mouse ocular, o OpenCV permite a obtenção de quadros em alta frequência condição essencial para detecção precisa de piscadas.

- **MediaPipe (Face Mesh)**

O MediaPipe é empregado para o rastreamento facial e identificação de landmarks oculares. O modelo Face Mesh fornece pontos faciais detalhados que possibilitam o cálculo geométrico da abertura dos olhos. A opção por essa biblioteca elimina a necessidade de treinamento de modelos próprios, reduzindo a complexidade do sistema e tornando-o viável em computadores convencionais.

- **Numpy**

A biblioteca Numpy é utilizada para operações matemática vetoriais, especialmente no cálculo da métrica Eye Aspect Ratio (EAR). Seu uso garante precisão e eficiência no processamento numérico, fundamental para análise contínua dos dados oculares.

- **Collections(deque)**

Estruturas do tipo deque são empregadas como buffers temporais, permitindo suavização dos dados de piscadas. Essas estratégias reduzem ruídos, piscadas involuntárias e falsas ativações, contribuindo para maior confiabilidade do sistema.

- **Asycio e WebSockets**

A comunicação entre o módulo de percepção ocular e a interface de controle é realizada por meio do WebSockets, com suporte do framework assíncrono asycio. Essa abordagem garante baixa latência e sincronização em tempo real entre os eventos oculares e as respostas do sistema.

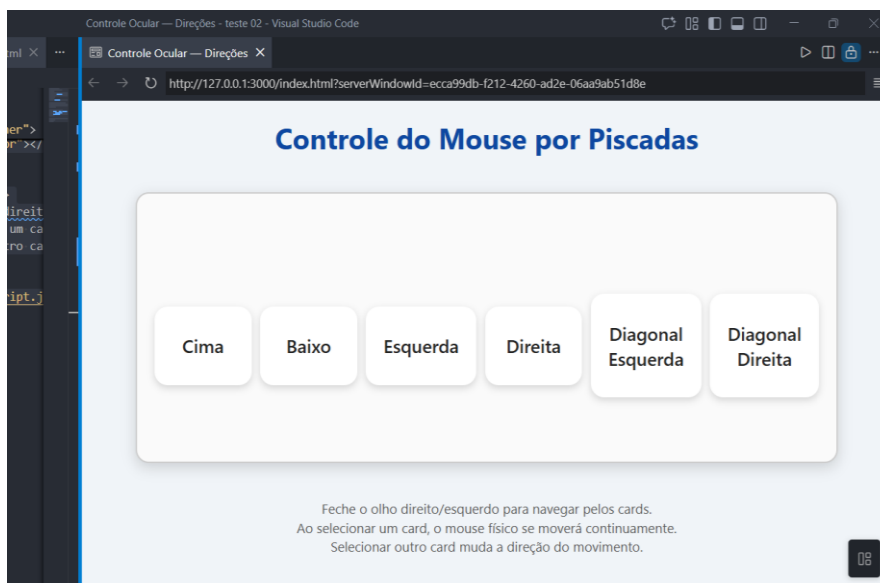
- **PyAutoGUI**

A biblioteca PyAutoGUI, é responsável pela atuação direta sobre o sistema operacional, permitindo mover o cursor e executar clique automáticas. Sua adoção possibilita que o Mouse Ocular funcione de forma transparente sobre qualquer aplicação do sistema, ampliando o alcance funcional.

### 9.1.3 Lógica de Controle do Cursor

No Mouse Ocular, as piscadas não são associadas diretamente as posições do curso, mas sim a direções de movimento discretas, representadas visuais por cards na interface. Cada card corresponde a uma direção específica como, cima, baixo, esquerda, direita ou diagonais. A navegação entre os cards ocorre por meio de piscadas direcionais, conforme a lógica apresentada nos capítulos anteriores. Uma vez selecionado um card o sistema inicia o movimento contínuo do curso naquela direção, até que outro comando seja acionado. Essa decisão projetual reduz a complexidade cognitiva do sistema, evitando a necessidade de rastreamento preciso do olhar sobre a tela, o que poderia ser cansativo ou inviável para determinados usuários.

Figura 24- print da tela do bloco de interação Mouse Ocular



Fonte - elaborada pelo autor

#### **9.1.4 Clique Automático por Permanência**

Além do movimento do cursor, o Mouse Ocular implementa um mecanismo de clique automático baseado em tempo de permanência. Quando o cursor permanece estático por um intervalo de tempo, o sistema interpreta essa condição como intenção de clique e executa ação automaticamente. Essa funcionalidade elimina a necessidade de comandos adicionais para clicar, simplificando a interação e reduzindo o esforço cognitivo e físico do usuário. Do ponto de vista de design de interação, esse mecanismo funciona como uma confirmação implícita, alinhada aos princípios de previsibilidade e controle defendidos por Norman (2013).

### **9.2 Bloco Comunicador Ocular**

O bloco Comunicador Ocular do sistema Gestus foi desenvolvido com o objetivo de possibilitar a comunicação alternativa aumentativa (CAA) para usuários com limitações motoras severas, permitindo a expressão de necessidades, sentimentos e intenções por meio de seleção ocular de frases pré-definidas. Diferentemente do bloco de mouse ocular, cuja o foco está na interação com o sistema operacional, o Comunicador Ocular atua diretamente na mediação da linguagem, assumindo um papel fundamental no processo de inclusão social, autonomia e dignidade do usuário

#### **9.2.1 Base Técnica Compartilhada**

O comunicador Ocular utiliza a mesma infraestrutura técnica previamente apresentada no bloco Mouse Ocular, especialmente no que se refere à:

- Captura de imagens;
- Rastreamento ocular;
- Detecção e validação de piscadas
- Comunicação assíncrona entre processamento e interface

Dessa forma, não há duplicação lógica algorítmica, reforçando o caráter modular reutilizável do sistema Gestus. A principal diferença reside na interpretação

semântica das ações do usuário, que, neste bloco, estão associadas à produção de fala.

### 9.2.2 Interface Baseada em Cards e Arquitetura da Informação

A interface do Comunicador Ocular é estruturada a partir de Cards organizados em grade, cada um representando uma frase ou expressão. Essa organização foi definida a partir de princípios de arquitetura da informação, priorizando clareza, legibilidade e facilidade de navegação.

A disposição dos cards permite:

- Rápida identificação visual;
- Navegação linear previsível
- Adaptação de conteúdo conforme o perfil do usuário.

A escolha por frases prontas reduz o esforço cognitivo e acelera o processo comunicacional, especialmente em situações que demandam respostas rápidas, como pedidos de ajuda ou necessidades básicas.

Figura 25 - print da tela do bloco de interação Comunicador Ocular



Fonte - elaborada pelo autor(2025)

### 9.2.3 Seleção por Permanência e Confirmação Implícita

Assim como nos demais blocos do Gestus, o Comunicador Ocular adota mecanismo de tempo de permanência (dwell time) como forma de confirmação. Uma vez selecionado um card, o sistema aguarda um intervalo de tempo antes de executar a ação, garantindo que a escolha seja intencional. Esse modelo elimina a necessidade de comandos adicionais de confirmação, reduzindo a complexidade da interação e promovendo uma experiência mais fluida e acessível

### 9.2.4 Síntese de voz como Feedback Auditivo

A principal especificidade técnica do comunicador Ocular é a integração com mecanismos de síntese de voz, responsáveis por transformar o conteúdo textual dos cards em fala audível. Para isso o sistema utiliza recursos nativos de síntese de fala disponíveis no ambiente web. Essa funcionalidade amplia o alcance do sistema, permitindo que o usuário:

- Se comunique verbalmente em ambientes sociais;
- expresse necessidades imediatas
- participe ativamente de interações cotidianas

Do ponto de vista do design de interação, a síntese de voz atua como o feedback auditivo confirmatório, reforçando a ação executada e aumentando a segurança do usuário quanto à sua escolha.

Figura 26 - Fluxo de ativação com feedback



Fonte - elaborada pelo autor

### **9.2.3 Personalização e Escalabilidade do Conteúdo**

Um dos aspectos centrais do Comunicador Ocular é sua capacidade de personalização. As frases apresentadas nos cards podem ser alteradas, adicionadas ou reorganizadas conforme o contexto de uso, idioma ou necessidades específicas do usuário. Essa flexibilidade reforça o conceito de design centrado no usuário, permitindo que o sistema seja moldado às particularidades individuais, ao invés de impor estruturas rígidas de comunicação.

### **9.3 Bloco Olhar Smart**

O bloco Olhar Smart constitui a camada mais avançada do sistema Gestus, ao expandir o uso do rastreamento ocular para o controle direto de ações no sistema operacional e a integração com dispositivos externos, como a garra robótica. Diferentemente dos blocos anteriores — centrados na navegação e na comunicação —, este módulo é orientado à execução de comandos, aproximando o Gestus de um ecossistema de automação inspirado nos princípios da Internet das Coisas (IoT).

Nesse bloco, o olhar deixa de atuar apenas como mecanismo de seleção e passa a funcionar como dispositivo de comando, capaz de acionar aplicações, serviços digitais e, potencialmente, atuadores físicos. Embora o esse bloco de interação não seja, em sua forma atual, um sistema de Internet das Coisas tradicional, sua arquitetura se alinha conceitualmente aos tais princípios. Nesse contexto, o olhar do usuário atua como sensor, o sistema Gestus funciona como unidade de processamento inteligente e as ações executadas representam atuadores digitais ou físicos. Essa lógica permite compreender o Olhar Smart como um modelo IoT centrado no usuário, na qual sinais biológicos substituem sensores convencionais, reforçando a proposta da acessibilidade e inclusão.

#### **9.3.1 Arquitetura Geral do Bloco Olhar Smart**

O Olhar Smart adota uma arquitetura cliente–servidor assíncrona, baseada em comunicação via WebSockets, permitindo troca contínua de dados em tempo real

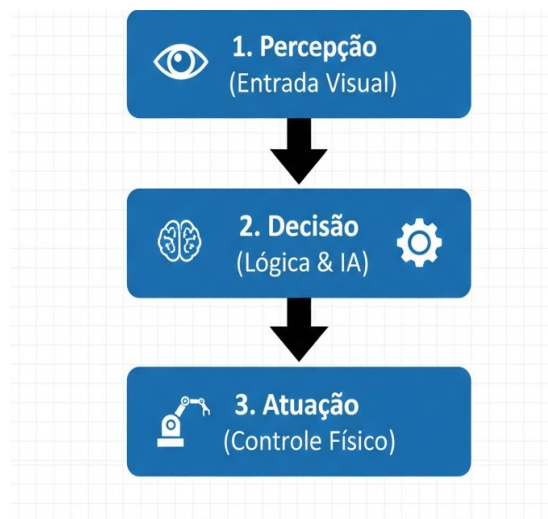
entre o módulo de rastreamento ocular (servidor Python) e a interface de interação (cliente web).

A arquitetura é composta por três camadas principais:

1. Camada de percepção: responsável pela captura e análise dos movimentos oculares;
2. Camada de decisão: Interpreta piscadas como comandos lógicos;
3. Camada de atuação: executa ações no sistema operacional ou em dispositivos externos

Essa separação reforça o caráter modular e extensível do Gestus, permitindo que novos tipos de atuadores — como garra robótica — sejam incorporados sem reestruturação do núcleo do sistema.

Figura 27 - Fluxo de ativação com garra robótica



Fonte - elaborado pelo autor

### 9.3.2 Integração Conceitual com a Garra robótica

A integração da garra robótica ao bloco Olhar Smart representa a transposição da interação ocular do ambiente digital para o ambiente físico, configurando o Gestus como um sistema híbrido de interação humano-máquina

assistiva. Enquanto os comandos inicialmente são direcionados a aplicações e serviços computacionais, a mesma estrutura lógica e comunicacional permite o acionamento de atuadores físicos, como motores e servomecanismos, por meio do olhar.

Do ponto de vista sistêmico, a garra robótica é compreendida como um atuador físico inserido no ecossistema Gestus, recebendo comandos de alto nível derivados da interpretação das intenções do usuário. Assim, o usuário não controla diretamente cada movimento mecânico, mas aciona ações abstratas, como abrir, fechar, segurar ou soltar objetos, respeitando princípios de acessibilidade cognitiva e redução de esforço mental. Essa abordagem está alinhada aos conceitos de design centrado no usuário, nos quais a complexidade técnica é encapsulada pelo sistema, permitindo que o usuário interaja por meio de metáforas simples e previsíveis. Dessa forma, o olhar atua como um meio de comando simbólico, e não como um controle manual contínuo, o que é especialmente relevante para usuários com severas limitações motoras.

Sob a perspectiva arquitetural, a integração da garra robótica pode ser compreendida como uma extensão natural da lógica já empregada no Olhar Smart: sinais oculares são interpretados, transformados em comandos estruturados e transmitidos a um módulo de atuação. Nesse contexto, a substituição de aplicações digitais por dispositivos físicos não altera a lógica central do sistema, apenas o destino final dos comandos.

Essa característica reforça o caráter modular e escalável do Gestus, permitindo que diferentes tipos de dispositivos físicos sejam integrados futuramente, como cadeiras de rodas motorizadas, braços robóticos ou sistemas de automação residencial. A garra robótica, portanto, funciona como um caso de aplicação representativo, demonstrando a viabilidade da arquitetura proposta.

Além disso, a relação entre o Olhar Smart e a garra robótica aproxima o sistema dos princípios da Internet das Coisas (IoT), ainda que em um contexto assistivo. Nesse modelo, o usuário assume o papel de sensor biológico, o sistema

Gestus atua como unidade de processamento inteligente e a garra robótica configura-se como atuador físico conectado em rede. Essa organização evidencia um paradigma de IoT centrado no humano, no qual a tecnologia se adapta às limitações do usuário, e não o contrário.

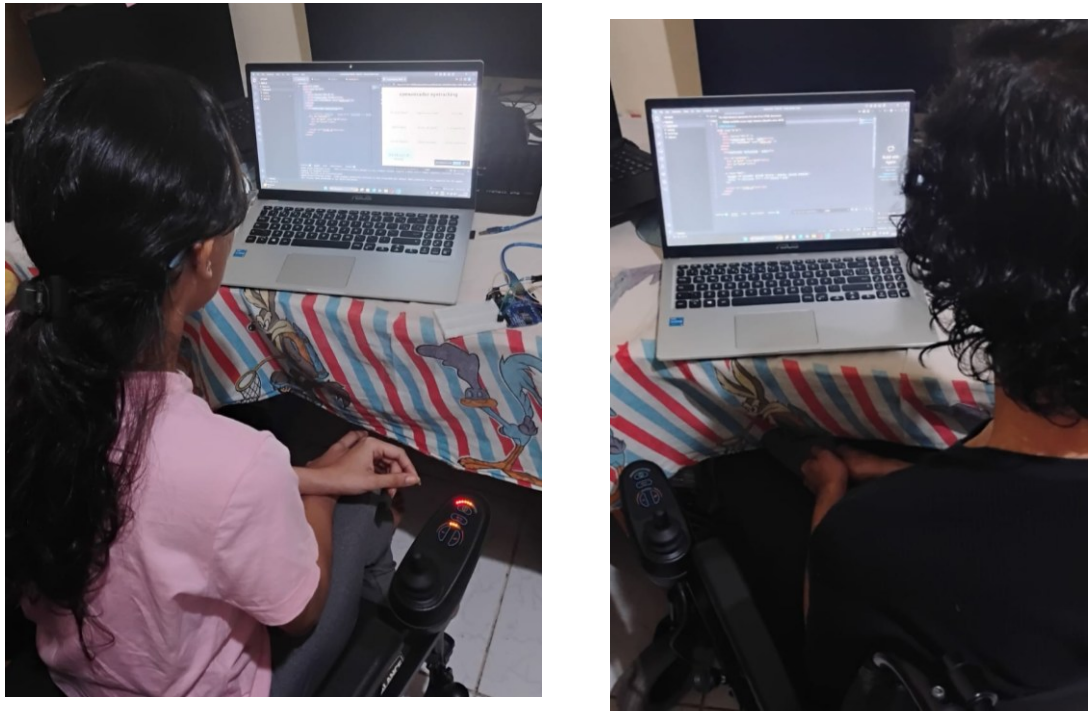
Por fim, a inclusão da garra robótica no projeto não tem como objetivo apenas demonstrar viabilidade técnica, mas evidenciar o potencial do Gestus como uma plataforma de ampliação da autonomia funcional, possibilitando interações com o mundo físico que normalmente exigiriam habilidades motoras preservadas. Dessa forma, a integração proposta reforça o caráter social, inclusivo e projetual do sistema, em consonância com os objetivos da Tecnologia Assistiva.

## **10. TESTES COM USUÁRIOS E VALIDAÇÃO DO SISTEMA**

Com o objetivo de validar a usabilidade, a compreensão da interface e a viabilidade da interação proposta, foram realizados testes com usuário utilizando o protótipo funcional do sistema Gestus. Os testes tiveram caráter exploratório e avaliativo, sendo compatíveis com a natureza projetual desta pesquisa, que se concentra na investigação de soluções de interação acessível para pessoas com mobilidade reduzida. O processo de validação buscou identificar limitações, pontos de melhoria e percepções subjetivas dos participantes, contribuindo diretamente para o refinamento da interface das diretrizes projetuais do sistema. Participaram do processo de validação usuários indivíduo A e indivíduo B, com a faixa etária entre 22 e 26 anos, selecionados por conveniência.

Nessa etapa inicial do projeto, os testes foram realizados com usuários simulados, estratégia amplamente utilizada em pesquisas de design e interação humano-computador, especialmente em fases conceituais e protótipos, permitindo avaliar aspectos de usabilidade antes da aplicação clínica ou assistiva real. Os participantes possuíam familiaridade básica com o uso de computadores, não sendo exigido conhecimento prévio sobre tecnologias assistivas

Figura 28 - Teste com usuário utilizando o sistema Gestus em tempo



Fonte - elaborada pelo autor

### 10.1 Procedimento do teste

Os testes foram conduzidos em ambiente controlado, utilizando um notebook com webcam e o protótipo do sistema Gestus. Inicialmente, os participantes receberam uma breve explicação sobre o objetivo do sistema e sobre os princípios de interação por rastreamento ocular. Os usuários foram convidados a executar tarefas previamente definidas, incluindo:

- Navegação entre telas da interface;
- Seleção de elementos por tempo de fixação do olhar;
- Ativação de comandos por piscadas intencionais;
- Exploração dos blocos de interação do sistema;
- Controle da garra robótica por interface gráfica

Durante a execução, foram observados aspectos como tempo de aprendizagem, taxa de erros, dificuldades e necessidades de auxílio externos.

## 10.2 Instrumentos de avaliação

Para validação do sistema Gestus , foi desenvolvido um formulário digital estruturado a partir da avaliação individual dos blocos de interação , respeitando o caráter modular do sistema . Cada módulo foi analisado separadamente, considerando diferentes formas de interação exigem critérios específicos de desempenho, não sendo adequada a aplicação de uma métrica única para todos os blocos.

O instrumento de avaliação foi composto por:

- questões de escala numérica de 0 a 5;
- perguntas fechadas de verificação funcional;
- questões abertas destinadas à coleta de sugestões e percepções subjetivas dos usuários

Para a avaliação da usabilidade do sistema GESTUS, foi adotada uma adaptação da System Usability Scale (SUS), aplicada por meio de um formulário digital elaborado na plataforma Google Forms. A escolha pela adaptação se deu em função das especificidades do público-alvo e da natureza experimental do protótipo, priorizando clareza, acessibilidade e facilidade de resposta.

O instrumento foi estruturado com perguntas fechadas baseadas em escala de satisfação do tipo Likert, variando de 1 (discordo totalmente / muito insatisfeito) a 5 (concordo totalmente / muito satisfeito), permitindo a coleta de dados quantitativos sobre aspectos como facilidade de uso, compreensão da interface, previsibilidade das ações e conforto visual.

Além das questões fechadas, o formulário incluiu perguntas abertas, com o objetivo de captar percepções subjetivas dos participantes, identificar dificuldades não previstas e levantar sugestões de melhoria. Essa combinação de dados quantitativos e qualitativos possibilitou uma análise mais ampla da experiência do

usuário, conforme recomendado por autores da área de Interação Humano-Computador e Design Centrado no Usuário (NORMAN, 2013; BAXTER, 2011).

Ressalta-se que o instrumento utilizado não corresponde à aplicação integral da escala SUS original, mas sim a uma versão adaptada, adequada ao contexto acadêmico e ao estágio de desenvolvimento do sistema, mantendo, contudo, o princípio central da avaliação de usabilidade.

### **10.3 Avaliação por blocos de interação**

A seguir são apresentados os critérios de avaliação de cada módulo do sistema Gestus.

- **Bloco 01 - Comunicador Ocular**

Responsável pela emissão de textos falados pelo sistema após a seleção do usuário.

Critério avaliados:

- facilidade de uso do bloco;
- clareza dos textos e respostas;
- capacidade de comunicação das intenções do usuário;

Além da avaliação numéricas, foram incluídas questões abertas sobre:

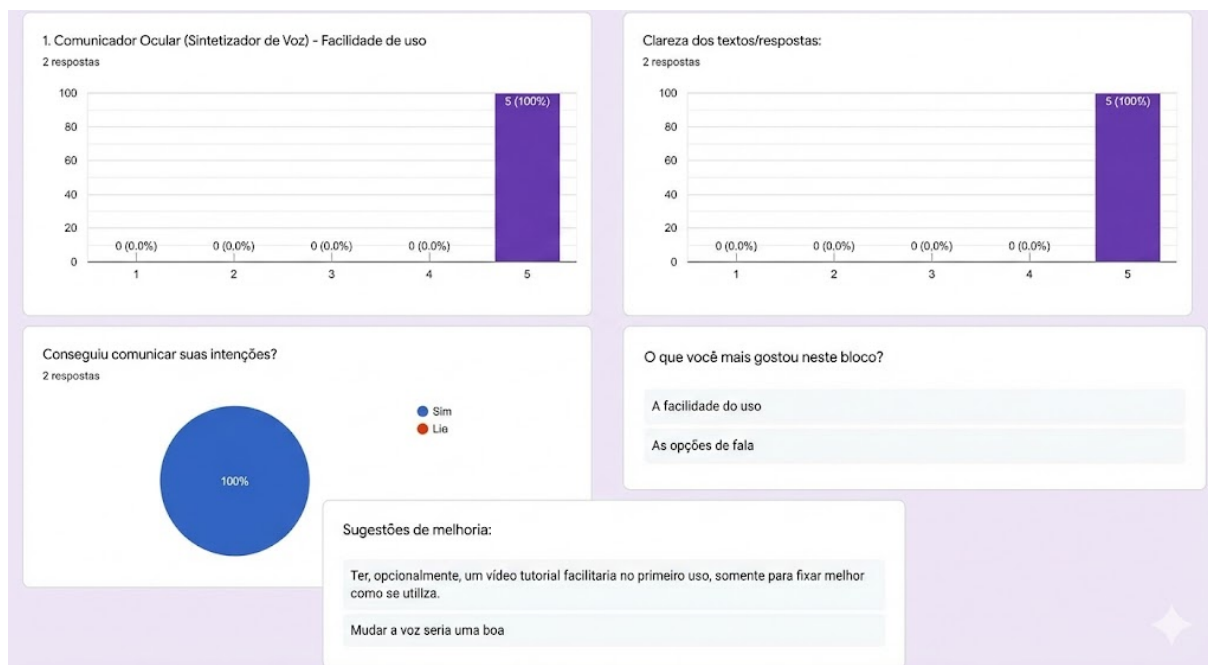
- aspectos positivos e percebidos;
- sugestões de melhoria do sistema de voz;

### **Resultados**

O bloco de comunicação apresentou os maiores índices de aprovação, obtendo notas máxima nos critérios de facilidade de uso e da clareza das respostas. Os participantes relataram que o sistema permitiu comunicar intenções de

forma rápida e compreensível, destacando a simplicidade do fluxo de seleção como ponto positivo.

Figura 29 - print dos feedbacks dos usuários em relação ao primeiro bloco de interação o Comunicador Ocular



Fonte - elaborada pelo autor

## ● Bloco 02 - Mouse ocular

Módulo responsável pelo controle do cursor e execução de cliques por meio de rastreamento ocular e piscadas intencionais:

Critérios avaliados:

- facilidade geral de uso;
- precisão do controle do cursor;
- capacidade de realizar ações desejadas
- estabilidade do rastreamento ocular

As perguntas abertas permitiram identificar dificuldades relacionadas à calibração, foco visual e coordenação entre curso e interface.

## Resultados

O módulo Mouse Ocular apresentou bom desempenho geral, especialmente, especialmente quanto à inovação da interação por piscadas. Entretanto, observou-se variação nas notas relacionadas à precisão do cursor, indicando que:

- a calibração ocular influencia diretamente desempenho;
- o tempo de piscada necessita de padronização ou personalização;
- controle simultâneo entre o cursor e blocos exige maior adaptação inicial.

Apesar dessas limitações, os usuários conseguiram realizar as ações propostas, caracterizando o módulo como funcional e promissor.

Figura 30- print dos feedbacks dos usuários em relação ao primeiro bloco de interação o Mouse Ocular



Fonte - elaborada pelo autor

- **Bloco 03 - Olhar Smart**

Responsável pela execução de funções do sistema operacional, como abertura de navegador, calculadora e pastas

Critérios avaliados:

- facilidade de controle dos movimentos;
- tempo de resposta (latência)
- precisão da execução dos comandos;
- sensação de segurança durante a operação

As respostas abertas contribuíram para identificar aspectos técnicos relacionados à velocidade de resposta e execução, fluidez dos movimentos e possíveis evoluções estruturais do sistema.

### **Resultados**

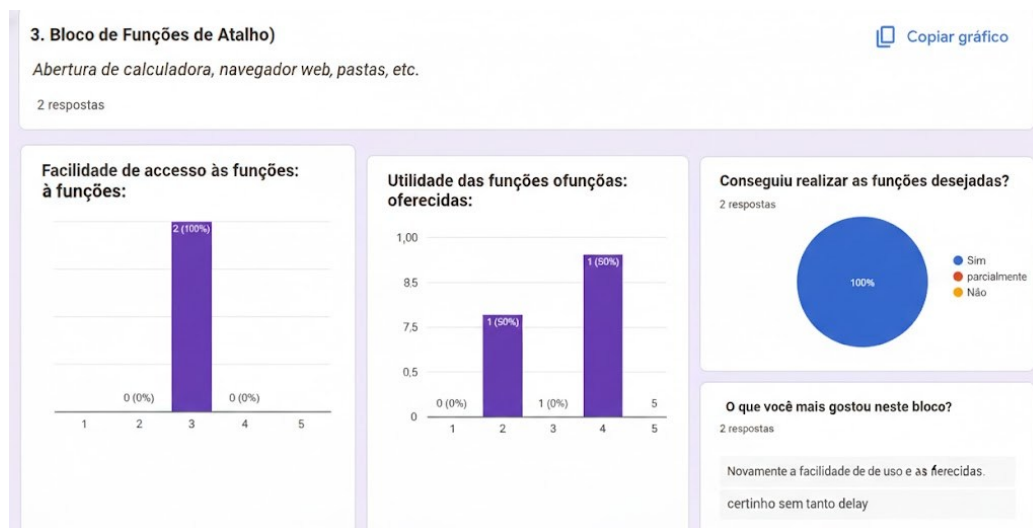
O bloco de funções apresentou avaliação predominantemente positiva, com destaque para:

- confiabilidade na abertura de programas;
- baixo tempo de resposta
- clareza dos cards de acesso rápido.

Algumas respostas indicaram a necessidade de:

- personalização das funções disponíveis;
- inclusão de mecanismo pesquisa;
- maior controle sobre múltiplas execuções simultâneas

Figura 31 - print dos feedbacks dos usuários em relação ao primeiro bloco de interação o Olhar smart



Fonte - elaborada pelo autor

#### ● Bloco 4 – Garra Robótica (Hardware Externo)

Módulo destinado ao controle de um dispositivo físico externo por meio do rastreamento ocular.

Critérios avaliados:

- facilidade de controle dos movimentos;
- tempo de resposta (latência);
- precisão da execução dos comandos;
- sensação de segurança durante a operação.

As respostas abertas contribuíram para identificar aspectos técnicos relacionados à velocidade de resposta, fluidez dos movimentos e possíveis evoluções estruturais do hardware.

### Resultados

O módulo de controle de hardware externo destacou-se como um dos elementos de maior impacto do sistema.

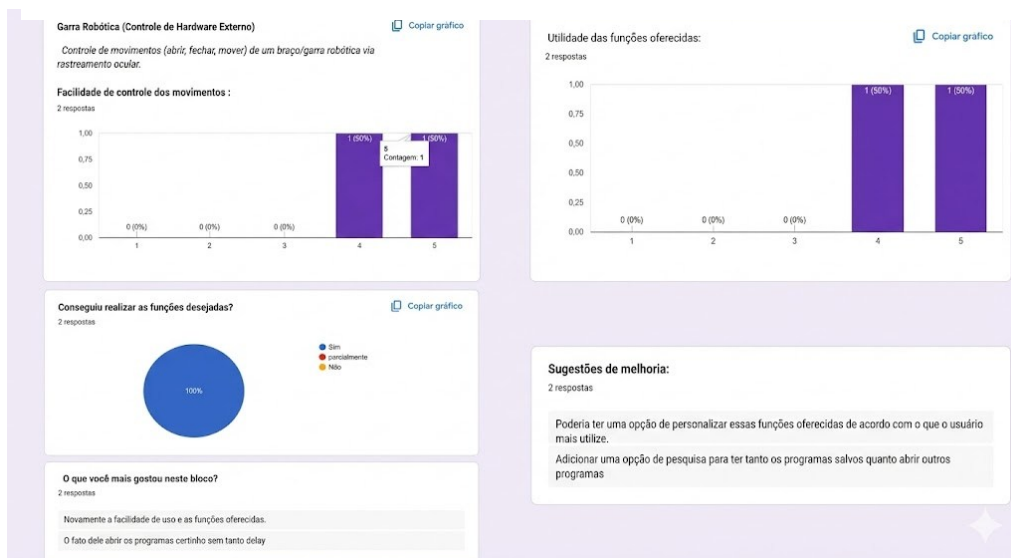
Os usuários atribuíram notas elevadas à experiência visual e funcional de observar o movimento físico do dispositivo sendo controlado exclusivamente pelo olhar.

Entretanto, os critérios relacionados à latência apresentaram notas inferiores, evidenciando:

- atraso perceptível entre comando e execução
- necessidade de otimização da comunicação com o hardware
- preocupação com movimentos acidentais

Mesmo com essas limitações, os participantes relataram sensação de controle e segurança durante a operação.

Figura 32 - print dos feedbacks dos usuários em relação ao primeiro bloco de interação garra robótica



Fonte - elaborada pelo autor

## 10.4 Síntese dos resultados

Na avaliação global os usuários, atribuíram notas máximas à experiência geral com o sistema entre os aspectos bem avaliados destacam-se:

- facilidade de uso geral
- integração entre blocos
- possibilidade de executar tarefas sem uso das mãos
- potencial de aplicação assistivas.

A maioria dos participantes afirmou que o sistema facilitou significativamente a interação com o computador, reforçando o propósito de acessibilidade proposto pelo o projeto.

As respostas abertas permitiram compreender de forma aprofundada a experiência dos usuários, complementando os dados quantitativos.

Entre os comentários positivos, destacam-se:

- “A facilidade de uso foi o ponto mais forte do sistema”
- “A precisão do controle pelo olhar me surpreendeu.”
- “Ver a garra robótica se movimentando foi algo impressionante”

Entre as principais sugestões de melhoria, foram recorrentes:

- inclusão de tutorial em vídeo
- possibilidade de ajuste da velocidade e tipo de piscada
- personalização de blocos de funções
- redução da latência do hardware externo;
- melhoria na estabilidade de execução simultânea de comandos.

Ao final dos testes e validação do usuário uma abordagem geral verificou-se que:

- O sistema Gestus apresentou alta aceitação pelos usuários;
- O conceito modular facilitou o aprendizado e a navegação;
- Os blocos de software demonstraram maior estabilidade que o hardware externo;
- ajuste técnicos podem ampliar significativamente a precisão e conforto da interação

Os resultados obtidos validam a proposta do sistema como uma solução viável para interação alternativa com computadores, especialmente no contexto da tecnologia assistiva.

## CONCLUSÃO

Este trabalho apresentou o desenvolvimento do Sistema GESTUS, um sistema assistivo modular baseado em rastreamento ocular e detecção de piscadas, voltado à ampliação da autonomia de pessoas com comprometimento motor severo. Partindo das limitações impostas por interfaces tradicionais, a proposta buscou oferecer alternativas de interação mais acessíveis, flexíveis e alinhadas às necessidades de usuários com diferentes níveis de mobilidade reduzida.

Ao longo do desenvolvimento, foi possível demonstrar que a utilização do eye tracking, associada a uma arquitetura modular organizada em blocos de interação, possibilita a adaptação do sistema a distintos contextos de uso, como comunicação alternativa, controle de interfaces digitais e interação com dispositivos físicos. Essa abordagem evidenciou potencial positivo em termos de usabilidade, acessibilidade e reprodutibilidade, especialmente por priorizar tecnologias de baixo custo e amplamente disponíveis.

A metodologia adotada, fundamentada nos princípios do Design Centrado no Usuário e da Interação Humano-Computador, mostrou-se adequada para orientar tanto a concepção quanto a avaliação do sistema. Os testes realizados permitiram identificar aspectos funcionais relevantes do Sistema GESTUS, bem como limitações técnicas e operacionais que indicam possibilidades claras de aprimoramento futuro.

Destaca-se que, embora interfaces cérebro-computador baseadas em sinais cerebrais (EEG) tenham sido discutidas no âmbito teórico deste trabalho, tais tecnologias não foram implementadas nesta etapa. O EEG foi abordado exclusivamente como parte do estado da arte e como possibilidade de expansão futura da arquitetura proposta, especialmente para usuários com níveis ainda mais severos de comprometimento motor.

Ressalta-se ainda que o processo de desenvolvimento do Sistema GESTUS contou com o apoio de ferramentas de Inteligência Artificial utilizadas como recurso educacional e organizacional. A Inteligência Artificial foi empregada para

auxiliar o pesquisador no aprendizado de programação, na compreensão da lógica dos algoritmos, na explicação de estruturas de código e na proposição de exercícios de fixação, ao longo de aproximadamente nove meses de estudo contínuo.

É fundamental explicitar que os códigos do sistema foram concebidos, implementados e testados pelo próprio autor, a partir do conhecimento adquirido durante esse processo de aprendizagem. A Inteligência Artificial não desenvolveu o sistema de forma autônoma, nem substituiu a atuação do pesquisador, atuando exclusivamente como ferramenta de apoio pedagógico, semelhante a um tutor digital, contribuindo para a organização do estudo, a revisão de conceitos e o amadurecimento técnico necessário para a execução do projeto.

Conclui-se, portanto, que o Sistema GESTUS representa uma contribuição relevante para o campo das tecnologias assistivas ao integrar design, computação e acessibilidade em uma solução modular, centrada no usuário e tecnicamente viável. Espera-se que este trabalho sirva como base para futuras pesquisas, aprimoramentos tecnológicos e aplicações práticas voltadas à inclusão digital e social de pessoas com deficiência motora.

## APÊNDICE A

Algoritmo de Detecção de Piscadas com Eye Tracking (Python + OpenCV + MediaPipe)

Este apêndice apresenta o trecho principal do algoritmo desenvolvido para a detecção de piscadas no sistema *Gestus*, utilizando visão computacional com as bibliotecas OpenCV e MediaPipe. A lógica baseia-se no cálculo do *Eye Aspect Ratio* (EAR), que permite identificar o fechamento intencional dos olhos. Para reduzir falsos positivos, é utilizado um buffer de suavização com deque, garantindo maior estabilidade na interpretação dos eventos oculares.

```
import asyncio
```

```
import WebSockets
```

```
import cv2
```

```
import MediaPipe as MP
```

```
import numpy as np
```

```
import json
```

```
from collections import deque
```

```
mp_face_mesh = mp.solutions.face_mesh
```

```
face_mesh = mp_face_mesh.FaceMesh(refine_landmarks=True)
```

```
RIGHT_EYE_LANDMARKS = [33, 160, 158, 133, 153, 144]
```

```
LEFT_EYE_LANDMARKS = [362, 385, 387, 263, 373, 380]
```

```
right_blinks = deque(maxlen=3)
```

```
left_blinks = deque(maxlen=3)
```

```
def eye_aspect_ratio(landmarks, points):
```

```
    a = np.linalg.norm(np.array([landmarks[points[1]].x,  
landmarks[points[1]].y]) -
```

```
np.array([landmarks[points[5]].x, landmarks[points[5]].y]))
```

```
    b = np.linalg.norm(np.array([landmarks[points[2]].x,  
landmarks[points[2]].y]) -
```

```
np.array([landmarks[points[4]].x, landmarks[points[4]].y]))
```

```
    c = np.linalg.norm(np.array([landmarks[points[0]].x,  
landmarks[points[0]].y]) -
```

```
np.array([landmarks[points[3]].x, landmarks[points[3]].y]))
```

```
    return (a + b) / (2.0 * c)
```

```
async def gaze_server(websocket):
```

```
    cap = cv2.VideoCapture(0)
```

```
    while True:
```

```
        ret, frame = cap.read()
```

```
        if not ret:
```

```
            continue
```

```
        rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
```

```
        results = face_mesh.process(rgb)
```

```
        data = {'blink_right': False, 'blink_left': False}
```

```
        if results.multi_face_landmarks:
```

```
            lm = results.multi_face_landmarks[0].landmark
```

```
            right_ear = eye_aspect_ratio(lm, RIGHT_EYE_LANDMARKS)
```

```
            left_ear = eye_aspect_ratio(lm, LEFT_EYE_LANDMARKS)
```

```
right_blinks.append(bool(right_ear < 0.23))
```

```
left_blinks.append(bool(left_ear < 0.23))
```

```
data['blink_right'] = bool(sum(right_blinks) >= 2)
```

```
data['blink_left'] = bool(sum(left_blinks) >= 2)
```

```
await websocket.send(json.dumps(data))
```

```
await asyncio.sleep(0.02)
```

```
async def main():
```

```
    async with websockets.serve(gaze_server, "localhost", 8765):
```

```
        await asyncio.Future()
```

```
if __name__ == "__main__":
```

```
    asyncio.run(main())
```

## APÊNDICE B

### *Lógica de Navegação por Cards e Dwell Time (JavaScript + WebSocket)*

Este apêndice apresenta o código responsável pela navegação linear entre os cards de interação no sistema Gestus, utilizando eventos de piscadas direita/esquerda para mover o foco e o tempo de permanência do olhar (dwell time) como confirmação de ação. A comunicação entre o Python e a interface ocorre via WebSocket.

```
const container = document.getElementById("container");
```

```
const cursor = document.getElementById("cursor");
```

```
const grid = document.getElementById("grid");
```

```
let currentIndex = 0;
```

```
let hoverTimer = null;
```

```
const DWELL_TIME = 1000;
```

```
const BLINK_DEBOUNCE = 300;
```

```
let lastBlinkTime = 0;
```

```
const dataCards = [
```

```
{ text: "Oi, tudo bem?" },  
  
{ text: "Estou com fome" },  
  
{ text: "Bom dia" },  
  
{ text: "Quero água" },  
  
{ text: "Muito obrigado" },  
  
{ text: "Ir ao banheiro" },  
  
{ text: "Chamar alguém" },  
  
{ text: "Estou cansado" },  
  
{ text: "Quero descansar" }  
  
];
```

```
function render() {  
  
  grid.innerHTML = "";  
  
  dataCards.forEach((item) => {  
  
    const div = document.createElement("div");  
  
    div.className = "card";  
  
    div.textContent = item.text;  
  
    grid.appendChild(div);  
  
  });  
  
}
```

```
,  
  
});  
  
updateSelectionVisual();  
  
moveCursorToIndex(currentIndex);  
  
}  
  
render();  
  
const socket = new WebSocket("ws://localhost:8765");  
  
socket.onmessage = (event) => {  
  
  const { blink_right, blink_left } = JSON.parse(event.data);  
  
  const now = Date.now();  
  
  if (now - lastBlinkTime < BLINK_DEBOUNCE) return;  
  
  if (blink_right && !blink_left) {  
  
    lastBlinkTime = now;  
  
    moveSelectionLinear(1);  
  
  }  
  
}
```

```
    } else if (blink_left && !blink_right) {
```

```
        lastBlinkTime = now;
```

```
        moveSelectionLinear(-1);
```

```
    }
```

```
};
```

```
function moveSelectionLinear(delta) {
```

```
    const cards = document.querySelectorAll(".card");
```

```
    if (!cards.length) return;
```

```
    cards[currentIndex].classList.remove("active");
```

```
    currentIndex = Math.max(0, Math.min(cards.length - 1, currentIndex +  
delta));
```

```
    cards[currentIndex].classList.add("active");
```

```
    moveCursorToIndex(currentIndex);
```

```

    ,

    if (hoverTimer) clearTimeout(hoverTimer);

    hoverTimer = setTimeout(() => handleSelection(cards[currentIndex]),
DWELL_TIME);

}

```

```
function moveCursorToIndex(index) {
```

```
    const cards = document.querySelectorAll(".card");
```

```
    const cardRect = cards[index].getBoundingClientRect();
```

```
    const containerRect = container.getBoundingClientRect();
```

```
    const centerX = cardRect.left - containerRect.left + cardRect.width / 2;
```

```
    const centerY = cardRect.top - containerRect.top + cardRect.height / 2;
```

```
    cursor.style.transform = `translate(${centerX}px, ${centerY}px) translate(-
50%, -50%)`;
```

```
}
```

```
function updateSelectionVisual() {
```

,

```
document.querySelectorAll(".card").forEach((c, i) =>
  c.classList.toggle("active", i === currentIndex)
);
}
```

```
function handleSelection(card) {
  const text = card.textContent.trim();
  const utterance = new SpeechSynthesisUtterance(text);
  utterance.lang = "pt-BR";
  speechSynthesis.cancel();
  speechSynthesis.speak(utterance);
}
```

## APÊNDICE C

### *Estilo da Interface Gestus (CSS)*

Este apêndice apresenta o código responsável pela estilização da interface gráfica do sistema Gestus, incluindo a organização em grid, os cartões interativos e o cursor visual que representa o foco do olhar do usuário durante a navegação

```
body {  
  
    background: #f2f2f2;  
  
    font-family: "Segoe UI", sans-serif;  
  
    margin: 0;  
  
    text-align: center;  
  
}
```

```
h1 {  
  
    margin-top: 20px;  
  
    color: #222;  
  
}
```

```
#container {  
  
    position: relative;
```

,

width: 600px;

height: 400px;

margin: 40px auto;

border: 2px solid #ccc;

border-radius: 15px;

background: #f9f9f9;

overflow: hidden;

}

.grid {

display: grid;

grid-template-columns: repeat(auto-fit, minmax(150px, 1fr));

gap: 15px;

padding: 20px;

height: 100%;

}

.card {

,

background: white;

border-radius: 15px;

box-shadow: 0 3px 8px rgba(0, 0, 0, 0.15);

padding: 30px 10px;

font-size: 1.2em;

transition: transform 0.2s, background 0.2s, box-shadow 0.2s;

cursor: none;

text-align: center;

}

.card.active {

transform: scale(1.1);

background: #b3e5fc;

box-shadow: 0 6px 15px rgba(0, 0, 0, 0.25);

}

#cursor {

position: absolute;

,

width: 25px;

height: 25px;

background: rgba(0, 255, 0, 0.8);

border: 2px solid white;

border-radius: 50%;

pointer-events: none;

transform: translate(-50%, -50%);

transition: transform 0.05s linear;

z-index: 100;

}

## APÊNDICE D

### *Estrutura da Interface Web do Comunicador Gestus (HTML)*

Este apêndice apresenta a estrutura base em HTML da interface do sistema Gestus. O código define o container principal, a área de exibição dos cartões de comunicação e o cursor visual que representa o foco do olhar do usuário. A página se conecta aos estilos (CSS) e à lógica de navegação (JavaScript) apresentados nos apêndices anteriores.

```
<!DOCTYPE html>

<html lang="pt-br">

  <head>

    <meta charset="UTF-8" />

    <title>Comunicador Blink</title>

    <link rel="stylesheet" href="style.css" />

  </head>

  <body>

    <h1>Comunicador EyeTracking</h1>
```

,

```
<!-- Container restrito: limita a área de interação do cursor -->
```

```
<div id="container">
```

```
  <div id="grid" class="grid"></div>
```

```
  <div id="cursor"></div>
```

```
</div>
```

```
<script src="script.js"></script>
```

```
</body>
```

```
</html>
```

## REFERÊNCIAS

TEIXEIRA, Cláudia Regina et al. **Introdução à tecnologia assistiva**. Brasília: Secretaria de Educação Continuada, Alfabetização, Diversidade e Inclusão, 2018.

BARBOSA, Simone; SILVA, Bruno Santana da; SILVA, Thiago Santana da. **Interação Humano-Computador e Experiência do Usuário**. Porto Alegre: Bookman, 2021.

TEIXEIRA, Cláudia Regina et al. **Introdução à tecnologia assistiva**. Brasília: Secretaria de Educação Continuada, Alfabetização, Diversidade e Inclusão, 2018.

BATISTA, Leonardo dos Santos; KUMADA, Kate Mamhy Oliveira; BENITEZ, Priscila. **Rastreamento ocular e a educação especial inclusiva: uma revisão sistemática**. *Olhar de Professor*, Ponta Grossa, v. 26, p. 1–22, 2023.

BURGER, Carlos Alberto Coletto; KNOLL, Graziela Frainer. **Eye tracking: possibilidades de uso da ferramenta de rastreamento ocular na publicidade**. *Fronteiras – Estudos Midiáticos*, São Leopoldo, v. 20, n. 3, p. 340-353, set./dez. 2018.

NORDI, Thiago Matheus. **Sistema de aquisição de biopotenciais e neuroestimulação para estimulação cerebral profunda**. 2021. 128 f. Dissertação (Mestrado em Engenharia Elétrica) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2021.

SIM, Kok Swee; KHO, Desmond Teck Kiang; LIM, Zheng You. **EEG Controlled Wheelchair**. ASTEC Web of Conferences, 2016.

DUSIK, Cláudio Luciano. **A interação entre interface cérebro-computador e sujeitos com incapacidade motora grave para comunicação**. 2018. 243 f. Tese (Doutorado em Educação) – Universidade Federal do Rio Grande do Sul, Faculdade de Educação, Porto Alegre, 2018.

GRANDINI, Jônatas Negri. **Interface cérebro máquina de baixo custo como tecnologia assistiva**. 2017. 85 f. Trabalho de Conclusão de Curso (Graduação em

Engenharia de Computação) – Universidade Federal de Santa Catarina, Campus Araranguá, Araranguá, 2017.

BRAIN SUPPORT. **Como funciona e quais são os tipos de Eye Tracking?** *Brain Support*, [s.l.], 2023. Disponível em: <https://brainsupport.com.br/blog/como-funciona-e-quais-sao-os-tipos-de-eye-tracking/485>. Acesso em: 19 jul. 2025.

GLOBO.COM. **Jovem parapléptico usa exoesqueleto e chuta bola na abertura da Copa.** *G1 – Ciência e Saúde*, [s.l.], 12 jun. 2014. Disponível em: <https://g1.globo.com/ciencia-e-saude/noticia/2014/06/jovem-paraplegico-usa-exoesqueleto-chuta-bola-na-abertura-da-copa.html>. Acesso em: 25 jul. 2025.

NORMAN, Donald A. **O design do dia a dia.** 2. ed. Rio de Janeiro: Rocco, 2002.

REZENDE, Sarah Corrêa Soler Albino Titz de. **Neurodesign: a neurociência aplicada ao design.** 1. ed. São Paulo: Blücher, 2024. 142 p. ISBN 978-8521220909.

BAXTER, Mike. **Projeto de produto: guia prático para o design de novos produtos.** São Paulo: Blücher, 2011. 344 p. 3. ed.

GABRIEL, Martha; KISO, Rafael. **Marketing na Era Digital: conceitos, plataformas e estratégias.** 2. ed. São Paulo: Atlas (Grupo GEN), 2020. 576 p. ISBN 978-85-97025-55-2

ISHIKAWA, Kaoru. **What is Total Quality Control? The Japanese Way.** Englewood Cliffs: Prentice Hall, 1985.

BRASIL. **Secretaria de Direitos Humanos da Presidência da República. Coordenadoria Nacional para Integração da Pessoa com Deficiência – CORDE. Cartilha de tecnologia assistiva.** Brasília: SDH/PR, 2009

SANTOS, Felipe Jucá dos; PORFÍRIO FILHO, Edilson Rocha. **Solução em automação assistiva utilizando Arduino e Raspberry Pi.** In: XLV Congresso Brasileiro de Educação em Engenharia, Joinville-SC, 26 set. 2017. Anais.

MENEZES, Gabriel dos Santos; GUIMARÃES, Lucas Siqueira; GUIMARÃES, Miguel de Oliveira. **Estudo exploratório e descritivo sobre tecnologia assistiva e automação doméstica com Arduino.** In: SIMPÓSIO DE ENGENHARIA DE PRODUÇÃO – SIMPEP, 31., 2024, Bauru. Anais [...]. Bauru: UNESP, 2024.

ESCOLA DIGITAL – PROFESSOR. **Tecnologias assistivas.** *Portal Escola Digital – Professor.* Disponível em: [https://professor.escoladigital.pr.gov.br/tecnologias\\_assistivas](https://professor.escoladigital.pr.gov.br/tecnologias_assistivas). Acesso em: 25 jul. 2025.

DAMÁSIO, António R. **O erro de Descartes: emoção, razão e o cérebro humano.** São Paulo: Companhia das Letras, 1996 (edição em português). Publicação original em inglês: *Descartes' Erro*, Putnam, 1994

LOH MEDICAL. **Inovações tecnológicas em cadeiras de rodas.** *Blog Loh Medical*, s.l., out. 2024. Disponível em: <https://www.lohmedical.com/pt-br/blog/inovacoes-tecnologicas-em-cadeiras-de-rodas>. Acesso em: 25 jul. 2025.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION – **ISO.** *ISO 9241-11: Ergonomic requirements for office work with visual display terminals (VDTs) – Part 11: Guidance on usability.* Genebra: ISO, 1998.

ARENDDT, Hannah. **A condição humana.** 10. ed. Rio de Janeiro: Forense Universitária, 2009. 341 p.

MARTINS, João Carlos Monteiro. **Introdução ao design do produto modular: considerações funcionais, estéticas e de produção.** 2002. 121 f. Dissertação (Mestrado em Design Industrial) – Faculdade de Engenharia da Universidade do Porto, Porto, 2002.

**IBGE – INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA.** Censo 2022: Brasil tem 14,4 milhões de pessoas com deficiência. *Agência de Notícias IBGE*, Rio de Janeiro, 07 dez. 2023. Disponível em: <https://agenciadenoticias.ibge.gov.br/agencia-noticias/2012-agencia-de-noticias/noticias/43463-censo-2022-brasil-tem-14-4-milhoes-de-pessoas-com-deficiencia>. Acesso em: 25 jul. 2025.

OPENAI. **ChatGPT.** Versão 4.0. San Francisco: OpenAI, 2025. Disponível em: <https://chat.openai.com>. Acesso em: [Data do seu último acesso, ex: 20 jan. 2026].

PYTHON SOFTWARE FOUNDATION. **Python Language Reference**, version 3.12. Disponível em: <https://www.python.org>. Acesso em: 15 nov. 2025.

MOZILLA CONTRIBUTORS. **JavaScript: MDN Web Docs.** 2025. Disponível em: <https://developer.mozilla.org/pt-BR/docs/Web/JavaScript>. Acesso em: 10 dez. 2025.

W3C. **HTML & CSS.** World Wide Web Consortium, 2025. Disponível em: <https://www.w3.org/standards/webdesign/htmlcss>. Acesso em: 05 out. 2025.

VYGOTSKY, Lev Semenovich. **A formação social da mente: o desenvolvimento dos processos psicológicos superiores.** 7. ed. São Paulo: Martins Fontes, 2007.

GOOGLE. **MediaPipe Solutions Guide.** 2024. Disponível em: <https://developers.google.com/mediapipe>. Acesso em: 12 set. 2025.

ITSEEZ. **Open Source Computer Vision Library.** 2024. Disponível em: <https://opencv.org>. Acesso em: 20 set. 2025.