

João Victor Ferreira Castex da Motta
Pedro Schuves Marodin
Rafael Merchiori de Souza

13/05/2025

SignSpeak – Óculos tradutor de libras em áudio

Prof. Dr. César Manuel Vargas Benitez

SUMÁRIO

Sumário	2
Lista de Figuras	3
Lista de Tabelas	4
Lista de Siglas e Abreviaturas	5
1 Introdução	6
2 Escopo do Projeto	7
2.1 Itens dentro do escopo	7
2.2 Itens fora do escopo	8
3 Representação Intuitiva dos Processos	9
4 Requisitos Funcionais e Não-Funcionais	11
4.1 Requisitos Funcionais	11
4.2 Requisitos Não-Funcionais	11
5 Integração	13
6 Análise de Riscos	14
7 Cronograma Detalhado	15
7.1 Estrutura Metodológica por Fase	15
7.2 Estimativa de Carga Horária por Integrante	15
7.3 Cronograma Detalhado por Fase	15
8 Considerações sobre LIBRAS e Acessibilidade no Projeto	17
8.1 Importância da LIBRAS e da Acessibilidade em Projetos Tecnológicos	17
8.2 Exemplos de Aplicação da LIBRAS e Acessibilidade em Tecnologia	17
8.3 Referências e Legislação Pertinente	18
8.4 Considerações Finais	18
9 Materiais e Métodos	19
9.1 Hardware	19
9.2 Software	19
9.3 Métodos	20

LISTA DE FIGURAS

Figura 1 – Diagrama lógico do sistema SignSpeak.	9
Figura 2 – Diagrama ilustrativo do sistema SignSpeak.	10

LISTA DE TABELAS

Tabela 1 – Principais Riscos do Projeto SignSpeak	14
Tabela 2 – Estimativa de horas por integrante	15
Tabela 3 – Cronograma por fase e subetapas do projeto SignSpeak	15

LISTA DE SIGLAS E ABREVIATURAS

CNN	Convolutional Neural Network (Rede Neural Convolucional)
CSI	Camera Serial Interface
FPS	Frames Per Second (Quadros por Segundo)
IA	Inteligência Artificial
LIBRAS	Língua Brasileira de Sinais
TTS	Text-to-Speech (Conversão de Texto para Fala)
USB	Universal Serial Bus

1 Introdução

A comunicação entre pessoas surdas e ouvintes ainda representa um grande desafio, especialmente em situações cotidianas nas quais não há a presença de intérpretes de LIBRAS (Língua Brasileira de Sinais). Essa limitação compromete a inclusão social e restringe a autonomia das pessoas surdas em contextos como atendimentos médicos, repartições públicas, ambientes educacionais e interações sociais em geral.

Considerando essa realidade, este projeto propõe o desenvolvimento do SignSpeak, um sistema portátil de tradução automática de gestos do alfabeto manual de LIBRAS para áudio em português falado. A solução será implementada em um protótipo de óculos inteligente com câmera embutida, capaz de capturar os gestos realizados pelo usuário. O reconhecimento dos sinais será feito por meio de Redes Neurais Convolucionais (Convolutional Neural Networks - CNNs), que identificarão em tempo real as letras representadas. Após o reconhecimento, o sistema converterá o gesto identificado em áudio, utilizando uma biblioteca de texto-para-fala (Text-to-Speech - TTS), sem a necessidade de conexão com a internet.

O SignSpeak integra técnicas de visão computacional, inteligência artificial e sistemas embarcados, oferecendo uma alternativa prática, acessível e discreta para melhorar a comunicação entre pessoas surdas e ouvintes. A proposta visa não apenas facilitar o diálogo em tempo real, mas também contribuir para a inclusão digital e social, promovendo maior independência e participação dos surdos na sociedade. Este projeto, ao focar na segurança e na comunicação, pode inspirar futuras adaptações que contemplem as necessidades de todos, evidenciando o potencial da tecnologia embarcada na criação de soluções inteligentes e inclusivas. Trabalhos futuros podem explorar melhorias na precisão da detecção, a expansão do vocabulário de gestos para além do alfabeto manual e a incorporação de sensores adicionais para um monitoramento ainda mais abrangente e contextualizado.

Para acompanhar o desenvolvimento do projeto, acesse o [blog oficial do SignSpeak](#).

2 Escopo do Projeto

O projeto SignSpeak tem como objetivo o desenvolvimento de um sistema vestível, integrado a um par de óculos, capaz de realizar a tradução em tempo real dos gestos do alfabeto manual da LIBRAS para áudio em português. O sistema será projetado para operar de forma totalmente offline, ou seja, sem necessidade de conexão com a internet, assegurando seu funcionamento em diversos ambientes e garantindo maior privacidade e autonomia ao usuário. Por exemplo, o usuário poderia utilizar os óculos em uma consulta médica, em uma loja ou em uma sala de aula, comunicando-se de forma mais fluida mesmo na ausência de um intérprete.

O foco inicial e principal do projeto está na tradução do alfabeto manual da LIBRAS. Este conjunto de gestos é padronizado e forma a base para a soletração de nomes próprios, termos técnicos ou palavras que ainda não possuem um sinal específico em LIBRAS. A escolha do alfabeto manual como ponto de partida representa uma base sólida para o desenvolvimento, treinamento e validação do sistema de reconhecimento por meio de Redes Neurais Convolucionais (Convolutional Neural Networks - CNNs). Um exemplo prático seria o usuário soletrar seu nome letra por letra utilizando os gestos do alfabeto manual, e o sistema vocalizar cada letra reconhecida, permitindo que um ouvinte compreenda o que foi soletrado.

O sistema será composto por uma microcâmera acoplada à armação dos óculos, posicionada para capturar os gestos realizados com as mãos do usuário. As imagens capturadas serão processadas por um microcomputador embarcado, como um Raspberry Pi, que executará o modelo de CNN treinado para identificar cada letra do alfabeto manual. Após o reconhecimento do gesto, um módulo de software converterá a letra identificada em um sinal de áudio correspondente, utilizando uma biblioteca de conversão de texto-para-fala (Text-to-Speech - TTS) que também opere offline. Este áudio será então reproduzido por um pequeno alto-falante ou fone de ouvido integrado aos óculos.

2.1 Itens dentro do escopo

- **Captura de imagens da mão do usuário:** Implementação de um sistema de captura contínua de imagens da(s) mão(s) do usuário realizando os gestos do alfabeto manual da LIBRAS, utilizando uma câmera acoplada aos óculos.
- **Processamento das imagens em tempo real:** Desenvolvimento e implementação de algoritmos baseados em Redes Neurais Convolucionais (CNNs) para o reconhecimento e classificação dos gestos correspondentes a cada letra do alfabeto manual de LIBRAS. Por exemplo, o sistema deve ser capaz de distinguir o gesto da letra 'A' do gesto da letra 'B' com alta acurácia.
- **Conversão do gesto reconhecido em áudio falado:** Utilização de uma biblioteca de Texto-Para-Fala (TTS) que funcione offline para converter a letra reconhecida (após a classificação pela CNN) em um som audível em português. Por exemplo, se o gesto 'L' for reconhecido, o sistema deve vocalizar "Lê".
- **Operação offline completa:** Garantia de que todo o processamento, desde a captura da imagem até a emissão do áudio, ocorra integralmente no dispositivo embarcado (Raspberry Pi), sem qualquer dependência de conexão com a internet ou serviços em nuvem.

- **Integração dos componentes em um protótipo funcional e portátil:** Montagem de um protótipo físico que integre a câmera, o microprocessador, a fonte de energia (bateria) e o sistema de áudio em uma estrutura de óculos, visando a portabilidade e o conforto mínimo para o usuário.
- **Testes de acurácia e latência:** Realização de testes para avaliar a precisão do reconhecimento dos gestos e o tempo de resposta do sistema (latência entre o gesto e a saída de áudio), buscando otimizar ambos para uma comunicação fluida.

2.2 Itens fora do escopo

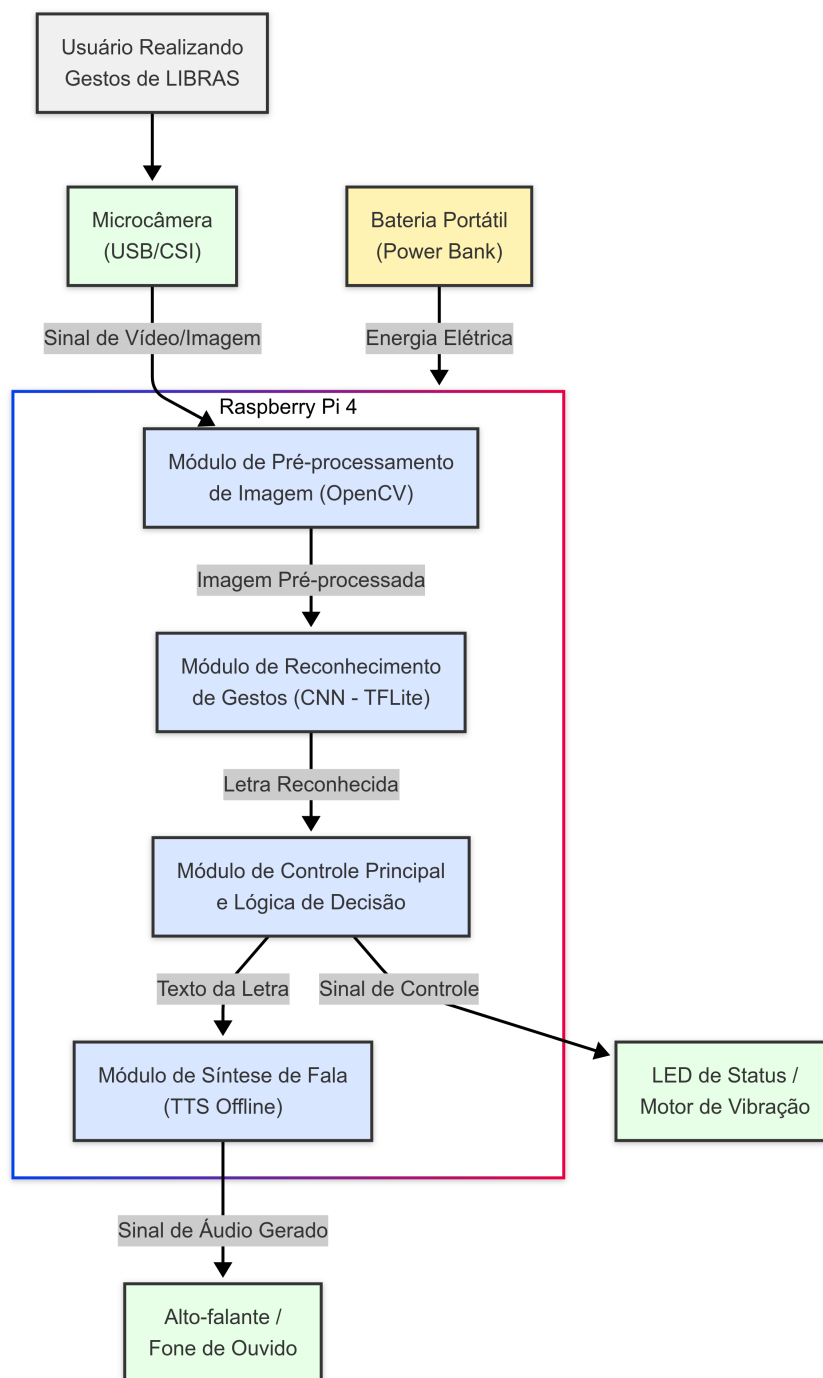
- **Reconhecimento de frases completas ou estruturas gramaticais complexas da LIBRAS:** O sistema se concentrará exclusivamente no alfabeto manual, não abrangendo a tradução de sinais que representam palavras inteiras, frases ou contextos gramaticais da LIBRAS, que possuem uma complexidade significativamente maior.
- **Tradução de outros elementos da LIBRAS:** Não serão incluídos no escopo o reconhecimento de expressões faciais, movimentos corporais, configurações de mão dinâmicas (que mudam durante o sinal) ou a localização espacial dos sinais, que são componentes cruciais da gramática da LIBRAS, mas extrapolam o foco no alfabeto manual.
- **Tradução bidirecional (Português para LIBRAS):** O sistema traduzirá apenas de LIBRAS (alfabeto manual) para áudio em português, não contemplando a funcionalidade inversa.
- **Integração com serviços em nuvem ou bancos de dados online:** Todas as funcionalidades serão locais para garantir a operação offline e a privacidade.
- **Design final do produto para comercialização:** O projeto foca no desenvolvimento de um protótipo funcional para validação do conceito, não incluindo etapas de design industrial para produção em massa ou comercialização.
- **Treinamento de múltiplos usuários sem recalibração extensiva:** Embora se busque robustez, o foco inicial será em um modelo treinado que funcione bem para o usuário principal ou com variações limitadas, não garantindo desempenho ótimo para qualquer novo usuário sem ajustes ou treinamento adicional do modelo.

3 Representação Intuitiva dos Processos

Para uma melhor compreensão da arquitetura funcional do sistema, as Figuras 1 e 2 apresentam dois fluxogramas complementares que descrevem as principais etapas do processo.

A Figura 1 apresenta um fluxograma lógico, estruturado em blocos funcionais que detalham cada componente do sistema de forma mais técnica, facilitando a visualização da arquitetura e da integração entre os módulos.

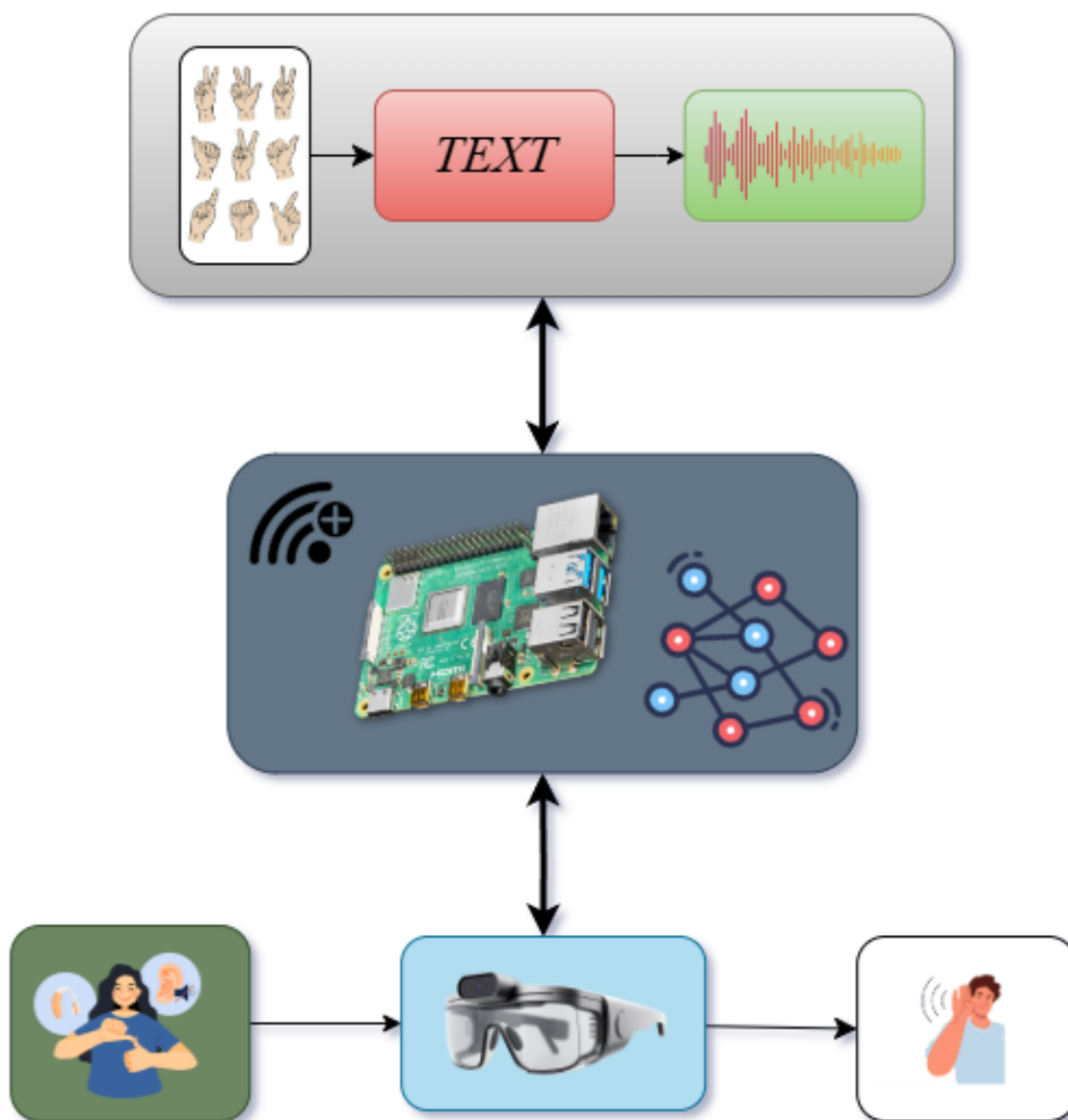
Figura 1 – Diagrama lógico do sistema SignSpeak.



Fonte: Autoria Própria

Já a Figura 2 traz um diagrama ilustrativo, com imagens que representam de forma intuitiva o fluxo de dados desde a aquisição da imagem até a emissão do áudio correspondente ao gesto identificado.

Figura 2 – Diagrama ilustrativo do sistema SignSpeak.



Fonte: Autoria Própria

4 Requisitos Funcionais e Não-Funcionais

Esta seção descreve os requisitos fundamentais para o desenvolvimento do sistema SignSpeak, divididos em funcionais e não-funcionais. Os requisitos funcionais definem os comportamentos e funcionalidades esperadas do sistema, enquanto os não-funcionais estabelecem critérios de desempenho, qualidade e usabilidade.

4.1 Requisitos Funcionais

- **RF01: Captura de imagem em tempo real:** O sistema deve ser capaz de capturar, continuamente e com clareza suficiente, imagens da(s) mão(s) do usuário por meio da câmera embutida nos óculos, focando na área onde os gestos do alfabeto manual são realizados. Por exemplo, a câmera deve ter uma taxa de quadros (frames per second - FPS) adequada para não perder movimentos rápidos da mão.
- **RF02: Reconhecimento de gestos do alfabeto manual de LIBRAS:** Utilizando Redes Neurais Convolucionais (CNNs) previamente treinadas, o sistema deve identificar, com um nível de precisão aceitável (a ser definido e validado, por exemplo, F1-score > 0,85 para cada letra em um conjunto de teste específico), o gesto realizado pelo usuário e associá-lo à letra correspondente do alfabeto manual da LIBRAS. O sistema deve ser capaz de diferenciar gestos visualmente similares, como 'A', 'E', 'S', 'O', 'T'.
- **RF03: Conversão de gesto reconhecido em áudio:** Após o reconhecimento da letra pela CNN, o sistema deve transformar a letra detectada em um sinal de áudio claro e inteligível em português, utilizando uma biblioteca de Texto-Para-Fala (TTS) offline. Por exemplo, ao reconhecer o gesto para 'P', o sistema deve vocalizar a palavra "Pê".
- **RF04: Reprodução do áudio pelo alto-falante embutido ou fone:** O som gerado pela biblioteca TTS deve ser emitido em tempo real através de um pequeno alto-falante integrado à armação dos óculos ou por meio de um fone de ouvido conectado, permitindo que um interlocutor ouvinte compreenda a letra soletrada.
- **RF05: Interface mínima de feedback ao usuário (Opcional, mas desejável):** O sistema pode fornecer um feedback simples ao usuário (ex: um LED que pisca brevemente ou uma leve vibração) para indicar que um gesto foi capturado e processado, ou para sinalizar erros de reconhecimento, se viável dentro do escopo do protótipo.

4.2 Requisitos Não-Funcionais

- **RNF01: Baixo consumo de energia:** O sistema deve ser energeticamente eficiente para permitir um tempo de uso prolongado com uma bateria portátil (ex: autonomia mínima de 2–3 horas de uso contínuo), sem gerar aquecimento excessivo nos componentes integrados aos óculos.
- **RNF02: Execução local (offline):** Todo o processamento de imagem, inferência da CNN e conversão para áudio deve ser realizado localmente no dispositivo embarcado (Raspberry Pi), sem qualquer dependência de conexão com a internet, para garantir privacidade e usabilidade em qualquer local.

- **RNF03: Alta performance no reconhecimento dos gestos:** O modelo de CNN deve apresentar um desempenho robusto, minimizando falsos positivos e falsos negativos. A métrica F1-score é adequada para avaliar o desempenho em dados potencialmente desbalanceados (algumas letras podem ser mais fáceis ou frequentes de gesticular). O sistema deve ser testado em diferentes condições de iluminação (dentro de limites razoáveis) e com pequenas variações no ângulo da mão.
- **RNF04: Tempo de resposta em “tempo real”:** O intervalo entre a realização completa do gesto pelo usuário e a reprodução do áudio correspondente deve ser suficientemente curto para não prejudicar a fluidez da comunicação (ex: idealmente inferior a 1–2 segundos por gesto).
- **RNF05: Portabilidade e ergonomia:** O protótipo do sistema deve ser leve, o mais compacto possível e integrado de forma relativamente discreta ao formato dos óculos, visando o conforto do usuário para uso contínuo por períodos moderados. O peso adicional dos componentes não deve causar desconforto significativo.
- **RNF06: Facilidade de manutenção e escalabilidade do software:** O código do projeto deve ser modular, bem documentado e versionado (ex: usando Git), permitindo futuras atualizações, como o retreinamento do modelo de CNN com mais dados, a expansão do vocabulário (se o escopo mudar no futuro) ou a inclusão de novas funcionalidades com esforço gerenciável.
- **RNF07: Robustez a variações de fundo e iluminação (limitada):** O sistema deve tentar manter um nível de acurácia aceitável sob variações moderadas de iluminação ambiente e diferentes fundos atrás da mão do usuário, embora condições extremas (muito escuro, muito claro, fundos muito complexos) possam degradar o desempenho.

5 Integração

O projeto SignSpeak envolve a integração de diversos componentes de hardware e software, unindo áreas como eletrônica, inteligência artificial (IA) e processamento de imagem em um único sistema vestível. A coesão entre esses elementos é fundamental para o sucesso do protótipo.

Do ponto de vista eletrônico, o sistema será construído em torno de um microcomputador Raspberry Pi 4. Este será o cérebro do dispositivo, responsável por gerenciar a câmera embutida nos óculos, executar os algoritmos de processamento dos gestos e controlar a reprodução do áudio resultante. A câmera, devidamente posicionada, capturará as imagens dos gestos manuais do alfabeto da LIBRAS (Língua Brasileira de Sinais). Estas imagens serão então processadas localmente no Raspberry Pi por uma Rede Neural Convolucional (Convolutional Neural Network - CNN) previamente treinada para esta tarefa específica de reconhecimento.

Após o reconhecimento do gesto pela CNN e a identificação da letra correspondente, o Raspberry Pi utilizará uma biblioteca de conversão de texto-para-fala (Text-to-Speech - TTS) que opera offline. Esta biblioteca converterá a letra identificada em um sinal de áudio em português. Esse áudio será, por fim, reproduzido por um pequeno alto-falante ou fone de ouvido integrado à estrutura dos óculos, permitindo que um interlocutor ouvinte compreenda a comunicação de forma discreta e eficiente.

A integração eficiente entre o hardware (câmera, Raspberry Pi, sistema de áudio, bateria) e o software (sistema operacional, drivers da câmera, modelo da CNN, biblioteca TTS, código de controle principal) é crucial. O objetivo é garantir que todas as etapas do processo – desde a captura da imagem, passando pelo processamento e reconhecimento, até a síntese e reprodução da fala – ocorram de maneira contínua, com baixa latência (em tempo real, conforme os requisitos) e sem a necessidade de conexão com a internet. Esta abordagem modular no desenvolvimento também visa facilitar futuras manutenções, depurações e potenciais expansões do sistema, como o aprimoramento do modelo de CNN ou a inclusão de novas funcionalidades, caso o escopo do projeto evolua.

6 Análise de Riscos

A gestão de riscos é fundamental para antecipar e mitigar problemas que possam comprometer o desenvolvimento do projeto *SignSpeak*. A seguir, são apresentados os riscos principais identificados, suas probabilidades, impactos e as respectivas ações preventivas ou mitigadoras. Diferentemente de meras dificuldades, os riscos aqui listados são eventos incertos que, se ocorrerem, podem ter um impacto negativo significativo nos objetivos do projeto.

Tabela 1 – Principais Riscos do Projeto SignSpeak

Risco	Prob. (0–5)	Impacto (0–5)	Ação Preventiva/Mitigação
Baixa acurácia persistente no reconhecimento dos gestos pela CNN	4	5	Coletar dataset amplo e bem anotado; testar diferentes arquiteturas de CNN; aplicar <i>data augmentation</i> ; usar métricas apropriadas (F1-score); definir critérios de aceitação; considerar simplificação do problema ou buscar consultoria especializada.
Atraso crítico na aquisição de componentes de hardware essenciais	3	4	Comprar componentes com antecedência; identificar fornecedores alternativos; ter plano B para componentes similares.
Consumo excessivo de energia e/ou superaquecimento dos componentes	3	4	Selecionar componentes com baixo consumo; otimizar software e modelo da CNN; implementar modos de economia de energia; usar dissipadores de calor ou ventoinha leve, se necessário.
Dificuldade significativa na integração entre os módulos de hardware e software	3	4	Projetar arquitetura modular; usar bibliotecas bem documentadas; realizar testes incrementais; versionar rigorosamente o código.
Latência alta entre gesto e saída de áudio	3	3	Otimizar CNN e processamento de imagem; usar biblioteca TTS eficiente; monitorar desempenho do Raspberry Pi e otimizar recursos.
Variações no estilo de gestos entre usuários não generalizadas pela CNN	4	3	Treinar com dados de múltiplos indivíduos; aplicar <i>data augmentation</i> ; documentar limitações quanto à generalização sem retreinamento.

7 Cronograma Detalhado

Para garantir o gerenciamento eficaz do projeto, foi elaborado um cronograma estruturado por fases e subetapas. Esse cronograma é dinâmico, sendo atualizado ao longo do semestre para refletir o progresso real. Ele orienta a divisão de responsabilidades, a alocação de tempo e permite margens para imprevistos.

7.1 Estrutura Metodológica por Fase

Cada fase do projeto segue uma estrutura padrão composta por três subetapas:

- **Planejamento:** definição de atividades, recursos e critérios de sucesso.
- **Execução:** realização prática das tarefas propostas.
- **Testes e Validação:** verificação da qualidade e adequação dos entregáveis.

7.2 Estimativa de Carga Horária por Integrante

Tabela 2 – Estimativa de horas por integrante

Item	Valor Estimado
Dedicação semanal	10 horas
Duração total do projeto	13 semanas
Total bruto de horas	130 horas
Total com 30% de folga	169 horas

7.3 Cronograma Detalhado por Fase

Tabela 3 – Cronograma por fase e subetapas do projeto SignSpeak

Fase	Período	Subetapa	Atividade	Tempo
F1: Planejamento Inicial	07/04 a 10/04	Planejamento	Reunião inicial, definição do escopo macro	6h
		Execução	Refinamento do escopo e conteúdo do site	–
		Testes	Validação interna do escopo	2h
F2: Plano de Projeto e Componentes	10/04 a 17/04	Planejamento	Estruturação do plano e pesquisa de componentes	5h
		Execução	Finalização do plano e montagem inicial do blog	–

Fase	Período	Subetapa	Atividade	Tempo
F3: Modelagem da Estrutura Física	17/04 a 25/04	Testes	Validação da lista de componentes	4h
		Planejamento	Design da armação e disposição dos componentes	4h
		Execução	Montagem inicial da estrutura física	15h
		Testes	Verificação de ergonomia e medidas	5h
F4: Montagem do Hardware	28/04 a 16/05	Planejamento	Planejamento elétrico e mecânico	3h
		Execução	Conexão dos dispositivos e ajustes físicos	–
		Testes	Testes elétricos e de integração física	8h
F5: Finalização de Hardware	19/05 a 23/05	Planejamento	Definição de critérios para testes finais	2h
		Execução	Testes finais e documentação técnica	–
		Testes	Validação do protótipo físico	3h
F6: Dataset e CNN	24/05 a 06/06	Planejamento	Estratégia de coleta e definição da CNN	8h
		Execução	Coleta, pré-processamento e treinamento	–
		Testes	Avaliação de acurácia e ajustes	13h
F7: TTS e Software Embarcado	06/06 a 13/06	Planejamento	Escolha da biblioteca TTS	2h
		Execução	Implementação de TTS e script de controle	12h
		Testes	Testes de integração entre CNN e TTS	–
F8: Integração Final	14/06 a 27/06	Planejamento	Definição de cenários de uso real	3h
		Execução	Integração total do sistema no protótipo	–
		Testes	Testes reais e ajustes finais	–
F9: Finalização Geral	28/06 a 11/07	–	Entrega final, relatório completo e apresentação	–

8 Considerações sobre LIBRAS e Acessibilidade no Projeto

A Língua Brasileira de Sinais (LIBRAS) é a língua natural da comunidade surda brasileira, reconhecida legalmente como meio de comunicação e expressão (Lei nº 10.436/2002). A promoção da acessibilidade comunicacional para pessoas surdas é um imperativo social e um direito, e a tecnologia desempenha um papel crucial nesse processo. O projeto SignSpeak, ao focar na tradução do alfabeto manual da LIBRAS para áudio, insere-se nesse contexto de busca por soluções que minimizem as barreiras de comunicação.

8.1 Importância da LIBRAS e da Acessibilidade em Projetos Tecnológicos

- **Inclusão Social e Cidadania:** Tecnologias que facilitam a comunicação em LIBRAS ou entre surdos e ouvintes promovem maior inclusão social, permitindo que pessoas surdas participem mais ativamente em diversas esferas da sociedade, como educação, trabalho, saúde e lazer. Por exemplo, um sistema como o SignSpeak poderia auxiliar um estudante surdo a soletrar um termo técnico para um colega ouvinte em uma atividade de grupo.
- **Autonomia:** Ferramentas de tradução ou facilitação da comunicação podem aumentar a autonomia de pessoas surdas em situações cotidianas, reduzindo a dependência de intérpretes humanos para interações mais simples ou emergenciais. Imagine a utilidade em um balcão de informações onde o atendente não sabe LIBRAS.
- **Direitos Linguísticos:** Reconhecer e incorporar a LIBRAS em projetos tecnológicos é uma forma de respeitar os direitos linguísticos da comunidade surda, valorizando sua língua e cultura.

8.2 Exemplos de Aplicação da LIBRAS e Acessibilidade em Tecnologia

Além do *SignSpeak*, há outras soluções tecnológicas que promovem a inclusão de pessoas surdas:

- **Softwares de Tradução Automática:** Plataformas como o Hand Talk ou VLibras (Suíte de Ferramentas VLibras) que traduzem texto e voz em português para LIBRAS (representada por avatares 3D) e, em alguns casos, o inverso (LIBRAS para texto/voz), embora este último seja um desafio tecnológico maior.
- **Conteúdo Digital Acessível:** Disponibilização de websites, aplicativos e vídeos com janelas de LIBRAS, legendas e transcrições. Por exemplo, videoaulas com intérprete de LIBRAS ou manuais de produtos com versão em LIBRAS.
- **Tecnologias Assistivas para Educação:** Jogos educativos em LIBRAS, dicionários digitais de LIBRAS, plataformas de e-learning com recursos de acessibilidade para surdos.
- **Sistemas de Atendimento:** Totens de autoatendimento ou sistemas de videochamada com opção de atendimento em LIBRAS.

8.3 Referências e Legislação Pertinente

A seguir, são apresentadas as principais legislações e diretrizes que fundamentam a acessibilidade comunicacional e o uso da LIBRAS:

- **Lei nº 10.436/2002:** Dispõe sobre a Língua Brasileira de Sinais - LIBRAS e dá outras providências.
- **Decreto nº 5.626/2005:** Regulamenta a Lei nº 10.436/02, que dispõe sobre a Língua Brasileira de Sinais - Libras, e o art. 18 da Lei nº 10.098/00 (Lei de Acessibilidade).
- **Lei nº 13.146/2015 (Estatuto da Pessoa com Deficiência):** Assegura e promove, em condições de igualdade, o exercício dos direitos e das liberdades fundamentais por pessoa com deficiência, visando à sua inclusão social e cidadania. Prevê a acessibilidade comunicacional.
- **WCAG (Web Content Accessibility Guidelines):** Embora focadas em conteúdo web, os princípios de acessibilidade (perceptível, operável, compreensível, robusto) são aplicáveis a diversos projetos tecnológicos.

8.4 Considerações Finais

No contexto do *SignSpeak*, embora o escopo inicial seja o alfabeto manual, a conscientização sobre a complexidade e riqueza da LIBRAS como um todo é fundamental. O projeto pode servir como um ponto de partida para discussões e futuros desenvolvimentos que abranjam aspectos mais amplos da língua e da cultura surda, sempre buscando o feedback e a colaboração da própria comunidade surda para garantir que as soluções sejam verdadeiramente úteis e respeitosas.

9 Materiais e Métodos

9.1 Hardware

O protótipo do SignSpeak será desenvolvido com componentes de baixo consumo energético, alta capacidade de processamento para sistemas embarcados e compatibilidade, visando a portabilidade e uma integração eficiente. A lista de hardware essencial inclui:

- **Raspberry Pi 4 Model B:** Microcomputador single-board que atuará como a unidade central de processamento (CPU) do sistema. Será responsável pela execução do sistema operacional, do modelo de Rede Neural Convolucional (CNN), do código de controle da câmera, da biblioteca de Texto-Para-Fala (TTS) e da interface com os periféricos de áudio.
- **Câmera compatível com Raspberry Pi:** Uma microcâmera com resolução e taxa de quadros (FPS) adequadas para capturar os gestos manuais de LIBRAS (Língua Brasileira de Sinais) com clareza. Deverá ser pequena e leve para ser acoplada à armação dos óculos (ex: Câmera Module V2 para Raspberry Pi ou uma câmera USB compacta de boa qualidade).
- **Sistema de Áudio (Saída):** Um pequeno alto-falante de baixa impedância ou um fone de ouvido mono/estéreo compacto para a reprodução do áudio gerado pela conversão TTS. A escolha dependerá do design final dos óculos e da preferência por discrição ou clareza em ambientes ruidosos.
- **Microfone (Entrada - Opcional/Futuro):** Embora não no escopo primário de tradução LIBRAS para áudio, um microfone poderia ser considerado para futuras expansões (ex: comandos de voz para controlar o sistema).
- **Bateria Portátil (Power Bank):** Uma bateria de Li-Ion ou Li-Po com capacidade suficiente (ex: 5000mAh ou mais, dependendo do consumo total) para alimentar o Raspberry Pi 4 e os periféricos por um período de uso razoável (conforme RNF01).
- **Armação de Óculos Adaptada:** Uma armação de óculos (pode ser sem lentes ou com lentes neutras) que servirá de base para a montagem da câmera, do Raspberry Pi (possivelmente em um case compacto acoplado à haste ou levado no bolso e conectado por fio, dependendo do protótipo) e do sistema de áudio.
- **Cabos e Conectores:** Cabos USB, jumpers, e conectores necessários para interligar os componentes.

9.2 Software

O desenvolvimento do software será realizado utilizando tecnologias open-source e bibliotecas amplamente testadas pela comunidade, facilitando a implementação e a manutenção.

- **Sistema Operacional:** Raspberry Pi OS (anteriormente Raspbian), uma distribuição Linux baseada no Debian, otimizada para o hardware do Raspberry Pi.
- **Linguagem de Programação Principal:** Python, devido à sua vasta gama de bibliotecas para visão computacional, aprendizado de máquina, e facilidade de prototipagem.

- **Visão Computacional:** Biblioteca OpenCV (Open Source Computer Vision Library) para Python, utilizada para a captura de imagens da câmera, pré-processamento das imagens (ex: redimensionamento, normalização, conversão de espaço de cores), detecção de mãos/região de interesse (ROI), e possivelmente para extração de características, se necessário antes de alimentar a CNN.
- **Aprendizado de Máquina/Inteligência Artificial:** Frameworks como TensorFlow (com Keras API) ou PyTorch para o design, treinamento e implementação da Rede Neural Convolutiva (CNN) responsável pelo reconhecimento dos gestos do alfabeto manual da LIBRAS. O modelo treinado será convertido para um formato otimizado para inferência no Raspberry Pi (ex: TensorFlow Lite).
- **Conversão Texto-Para-Fala (TTS):** Uma biblioteca TTS offline para Python, como eSpeak NG, pyttsx3 (que pode usar engines offline como SAPI5, NSSpeechSynthesizer, espeak), ou Festival. A escolha priorizará a clareza da voz, baixo consumo de recursos e funcionamento offline.
- **Bibliotecas de controle de hardware do Raspberry Pi:** Exemplo: RPi.GPIO para controle dos pinos GPIO (General Purpose Input/Output), bibliotecas para controle do display LCD, se específico.
- **Ferramentas de Desenvolvimento:** Ambientes de Desenvolvimento Integrado (IDEs) como VS Code, Thonny (para Raspberry Pi), e sistema de controle de versão Git com um repositório remoto (ex: GitHub, GitLab) para gerenciamento do código-fonte.

9.3 Métodos

O desenvolvimento do SignSpeak seguirá uma metodologia iterativa e incremental, com foco na prototipagem rápida e validação contínua das funcionalidades.

1. **Pesquisa e Definição Detalhada:** Aprofundamento na pesquisa sobre técnicas de reconhecimento de gestos manuais de LIBRAS, arquiteturas de CNN eficientes para sistemas embarcados, e bibliotecas TTS offline. Definição detalhada das especificações de cada componente de hardware e software.
2. **Coleta e Preparação do Dataset de Imagens:** Criação ou curadoria de um dataset de imagens de alta qualidade do alfabeto manual da LIBRAS. Este dataset deverá conter variações de iluminação, ângulos, e diferentes indivíduos (se possível) para treinar um modelo de CNN robusto. As imagens serão devidamente anotadas com as letras correspondentes.
3. **Treinamento e Validação do Modelo de CNN:** Desenvolvimento, treinamento e validação da CNN utilizando o dataset preparado. Serão experimentadas diferentes arquiteturas e hiperparâmetros para otimizar a acurácia e a velocidade de inferência. O modelo será avaliado com métricas como precisão, recall, F1-score e matriz de confusão.
4. **Desenvolvimento do Software Embarcado:** Implementação do pipeline de software no Raspberry Pi: captura de imagem pela câmera, pré-processamento, inferência com o modelo CNN treinado, e conversão da letra reconhecida para áudio usando a biblioteca TTS. O código será otimizado para desempenho em tempo real.

5. **Montagem do Protótipo Físico:** Integração dos componentes de hardware (Raspberry Pi, câmera, sistema de áudio, bateria) na armação dos óculos adaptada. Foco na ergonomia, portabilidade e estabilidade dos componentes.
6. **Testes e Iteração:**
 - **Testes Unitários:** Teste individual de cada módulo (captura de imagem, CNN, TTS, áudio).
 - **Testes de Integração:** Teste da comunicação e funcionamento conjunto dos módulos.
 - **Testes de Sistema e Usabilidade:** Teste do protótipo completo em cenários de uso simulados, avaliando a acurácia do reconhecimento, a latência, a clareza do áudio, a autonomia da bateria e o conforto do usuário. Coleta de feedback para identificar pontos de melhoria.
7. **Documentação:** Documentação contínua de todo o processo de desenvolvimento, incluindo o design do sistema, o código-fonte (comentado), os resultados dos testes, os desafios encontrados e as soluções adotadas. O blog do projeto será atualizado regularmente.

Esta abordagem metodológica permitirá que a equipe identifique e resolva problemas de forma ágil, garantindo que o protótipo final atenda aos requisitos estabelecidos e demonstre a viabilidade do conceito SignSpeak.