



# BR SMARTGUIA

*Carrinho de supermercado adaptado para pessoas cegas*



**Autor: Gabriel Martins Ribeiro**

**Brasília-DF**

**Fevereiro/2025**

## Sumário

<a href="#">1. Escopo do projeto</a>	<a href="#">3</a>
<a href="#">1.1 Apresentação do projeto</a>	<a href="#">3</a>
<a href="#">1.2 Objetivos do projeto</a>	<a href="#">3</a>
<a href="#">1.3 Principais requisitos</a>	<a href="#">4</a>
<a href="#">1.4 Descrição do funcionamento</a>	<a href="#">5</a>
<a href="#">1.5 Justificativa</a>	<a href="#">5</a>
<a href="#">2. Hardware</a>	<a href="#">6</a>
<a href="#">2.1 Diagrama em blocos</a>	<a href="#">6</a>
<a href="#">2.2 Função de cada bloco</a>	<a href="#">7</a>
<a href="#">2.3 Configuração de cada bloco</a>	<a href="#">8</a>
<a href="#">2.4 Especificações Técnicas</a>	<a href="#">8</a>
<a href="#">2.5 Lista de materiais</a>	<a href="#">9</a>
<a href="#">2.6 Descrição do funcionamento</a>	<a href="#">9</a>
<a href="#">2.7 Descrição da pinagem usada</a>	<a href="#">10</a>
<a href="#">2.8 Circuito completo do hardware</a>	<a href="#">10</a>
<a href="#">3. Software</a>	<a href="#">11</a>
<a href="#">3.1 Blocos funcionais</a>	<a href="#">11</a>
<a href="#">3.2 Descrição das funcionalidades</a>	<a href="#">12</a>
<a href="#">3.3 Definição das variáveis</a>	<a href="#">13</a>
<a href="#">3.4 Fluxograma</a>	<a href="#">13</a>
<a href="#">3.5 Inicialização</a>	<a href="#">14</a>
<a href="#">3.6 Configurações dos registros</a>	<a href="#">15</a>
<a href="#">3.7 Estrutura e formato dos dados</a>	<a href="#">16</a>
<a href="#">3.8 Organização da memória</a>	<a href="#">16</a>
<a href="#">3.9 Protocolo de comunicação</a>	<a href="#">17</a>
<a href="#">3.10 Formato do pacote de dados</a>	<a href="#">17</a>
<a href="#">4. Execução do projeto</a>	<a href="#">18</a>
<a href="#">4.1 Metodologia</a>	<a href="#">18</a>
<a href="#">4.2 Testes de validação</a>	<a href="#">19</a>
<a href="#">4.3 Discussão dos Resultados</a>	<a href="#">19</a>
<a href="#">4.4 Link do vídeo mostrando projeto funcionando</a>	<a href="#">20</a>
<a href="#">4.5 Manual do usuário</a>	<a href="#">20</a>
<a href="#">4.6 Imagens do produto</a>	<a href="#">21</a>
<a href="#">5. Referências</a>	<a href="#">25</a>

# 1. ESCOPO DO PROJETO

## 1.1 Apresentação do Projeto

O presente documento apresenta o **BR SmartGuia**, um **sistema embarcado assistivo** projetado para **auxiliar pessoas cegas na detecção de obstáculos e sinalizar sua presença em um supermercado**, contribuindo para a redução de riscos de colisões. Para alcançar esse objetivo, o sistema integra: sensor ultrassônico, matriz de LEDs, buzzers, botões e um display OLED, garantindo suporte tanto ao usuário quanto a terceiros, fornecendo alertas sobre obstáculos e exibindo informações relevantes em tempo real.

A **Tabela 1** apresenta os principais componentes do sistema e suas respectivas funções.

**Tabela 1** – Principais componentes e funcionalidades do BR SmartGuia

Componente	Função
<b>Sensor ultrassônico (HC-SR04)</b>	Detecta obstáculos e mede a distância ao objeto mais próximo.
<b>Buzzer</b>	Emite alertas sonoros para o usuário cego ao detectar um obstáculo.
<b>Matriz de LEDs RGB</b>	Sinaliza visualmente a presença do usuário para evitar colisões com terceiros.
<b>Display OLED</b>	Exibe mensagens de status para um acompanhante ou cuidador.
<b>Botões</b>	Permitem ativação/desativação dos LEDs.

Fonte: Autoria própria.

O sistema foi projetado para **proporcionar maior autonomia** às pessoas com deficiência visual em um supermercado.

## 1.2 Objetivos do Projeto

O **BR SmartGuia** tem como principal objetivo desenvolver um **sistema assistivo de baixo custo** que facilite a **mobilidade de pessoas cegas**, proporcionando **detecção de obstáculos em tempo real** e **sinalização eficaz da presença do usuário para terceiros**. Para garantir um funcionamento eficiente e acessível, foram estabelecidos **objetivos específicos**, detalhados na **Tabela 2**.

**Tabela 2 – Objetivos específicos do projeto BR SmartGuia**

Objetivo	Descrição
<b>Deteção de Obstáculos</b>	Utilizar um <b>sensor ultrassônico HC-SR04</b> para identificar objetos próximos e emitir alertas sonoros ao usuário.
<b>Sinalização Visual para Terceiros</b>	Exibir <b>a bandeira do Brasil na matriz de LEDs RGB</b> , aumentando a visibilidade do usuário.
<b>Exibição de Informações</b>	Exibir mensagens de status no <b>display OLED</b> para um acompanhante/cuidador.
<b>Facilidade de Controle</b>	Permitir a ativação/desativação dos LEDs por meio de botões.
<b>Solução Acessível e Eficiente</b>	Desenvolver um sistema de <b>baixo custo e fácil operação</b> .

Fonte: Autoria própria.

## 1.3 Principais Requisitos

Os requisitos do projeto foram elaborados com base nas **necessidades do usuário final** e nas **restrições técnicas do sistema embarcado**. Para melhor organização, os requisitos foram categorizados em **funcionais e não funcionais**, conforme ilustrado na **Tabela 3**.

**Tabela 3 – Requisitos do projeto BR SmartGuia**

Tipo	Requisitos
<b>Funcionais</b>	Detectar obstáculos a até <b>15 cm</b> e emitir <b>alertas sonoros</b> (buzzer).
	Exibir mensagens e status do sistema no <b>display OLED</b> .
	Permitir controle por <b>botões físicos</b>
	<b>Ativar a matriz de LEDs RGB</b> para que outras pessoas percebam a presença do usuário.
	<b>Reproduzir uma música inicial</b> ao ligar o dispositivo.
<b>Não Funcionais (Qualitativo)</b>	Utilizar <b>componentes de baixo custo e fácil acesso</b> .
	Garantir <b>usabilidade e acessibilidade</b> , incluindo um <b>manual em braile</b> .
	Manter o <b>consumo de energia dentro dos limites do Raspberry Pi Pico W</b> .

Fonte: Autoria própria.

## 1.4 Descrição do Funcionamento

A **Tabela 4** apresenta o fluxo de operação do **BR SmartGuia**, desde a **inicialização** até a **sinalização visual para terceiros**.

**Tabela 4** – Fluxo de operação do BR SmartGuia

Fase	Descrição
<b>Inicialização</b>	O sistema exibe uma mensagem no <b>display OLED</b> e emite um som inicial para indicar que está pronto para uso.
<b>Detecção de Obstáculos</b>	O <b>sensor ultrassônico HC-SR04</b> monitora continuamente a distância. Se um obstáculo for detectado a $\leq 15\text{ cm}$ , o <b>buzzer emite um alerta sonoro</b> .
<b>Controle do Usuário</b>	O usuário pode ativar/desativar a <b>matriz de LEDs RGB</b> usando os botões
<b>Sinalização Visual para Terceiros</b>	A <b>matriz de LEDs RGB</b> exibe a <b>bandeira do Brasil</b> para que outras pessoas percebam a presença do usuário cego e evitem colisões.

**Fonte:** Autoria própria.

## 1.5 Justificativa

Segundo o **Instituto Brasileiro de Geografia e Estatística (IBGE, 2010)**, o Brasil possui mais de **6,5 milhões de pessoas com deficiência visual**, das quais **528.624 são completamente cegas** [6]. Em nível global, a **Organização Mundial da Saúde (OMS)** estima que aproximadamente **36 milhões de pessoas** sejam cegas [7]. A mobilidade para essas pessoas pode ser desafiadora, especialmente em **ambientes movimentados e com pouca iluminação (como supermercados)**, onde obstáculos inesperados podem comprometer sua segurança e autonomia. Para enfrentar essa dificuldade, o **BRSmartGuia** propõe uma **solução assistiva inovadora**, que combina **tecnologia acessível e eficiente** para facilitar a locomoção e proporcionar maior independência ao usuário. Seu desenvolvimento segue uma abordagem modular e escalável, permitindo futuras expansões e personalizações.

Pesquisas apontam que os supermercados são espaços com fluxo intenso de pessoas e pouca adaptação para acessibilidade [8]. Estudos indicam que a inovação nesses setores é lenta, pois a demanda por tecnologia assistiva nesses ambientes é frequentemente negligenciada [10]. Assim, o **BR SmartGuia** se apresenta como uma solução para facilitar a

locomoção e prevenir acidentes, minimizando os riscos de colisões e fornecendo informações úteis ao usuário.

A **Tabela 5** apresenta um resumo dos benefícios proporcionados pelo sistema.

**Tabela 5 – Benefícios do projeto BR SmartGuia**

Problema	Solução Proposta
Dificuldade em detectar obstáculos	Uso do <b>sensor ultrassônico</b> e alerta por <b>buzzer</b> .
Risco de colisões com outras pessoas	<b>Matriz de LEDs RGB</b> sinaliza visualmente a presença do usuário.
Dificuldade de comunicação com acompanhantes	O <b>display OLED</b> exibe mensagens para cuidadores.
Custo elevado de outras soluções	Desenvolvimento com <b>componentes acessíveis e baratos</b> .

Fonte: Autoria própria.

Embora existam dispositivos de assistência baseados em sensores ultrassônicos, o **BR SmartGuia** se destaca por **integrar múltiplos recursos em um único sistema**, oferecendo **alertas sonoros, visuais e exibição OLED**, garantindo uma abordagem **mais completa e acessível**. Diferentemente de soluções convencionais (como o Amazon Cart [9]), este projeto prioriza **baixo custo, fácil implementação e flexibilidade**, facilitando a interação tanto para o usuário quanto para terceiros.

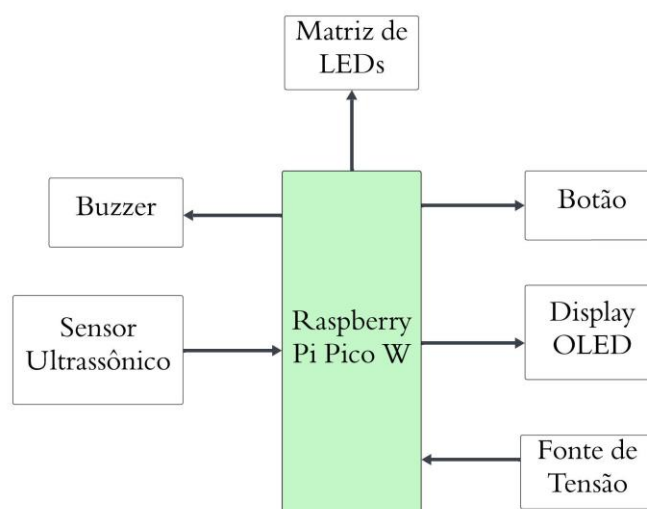
Além disso, ao contrário de outros sistemas que exigem infraestrutura especializada, [11] o BR SmartGuia pode ser implementado de forma independente, sem necessidade de adaptações caras no ambiente, tornando-se uma solução viável e acessível para diversos contextos.

## 2. HARDWARE

### 2.1 Diagrama em Blocos

O sistema **BR SmartGuia** é composto por diversos módulos interligados ao **Raspberry Pi Pico W**, que atua como a **unidade central de controle**. A **Figura 1** apresenta o **diagrama em blocos** do sistema.

**Figura 1 – Diagrama em blocos do hardware**



**Fonte:** Autoria Própria

## 2.2 Função de Cada Bloco

Cada componente tem uma função essencial para o funcionamento do sistema, conforme mostrado na **Tabela 6**.

**Tabela 6 – Função de cada bloco do sistema BR SmartGuia**

Bloco	Função
<b>Raspberry Pi Pico W</b>	Unidade central de controle que processa os sinais e gerencia os periféricos.
<b>Sensor ultrassônico (HC-SR04)</b>	Mede a distância de obstáculos e envia os dados para o microcontrolador.
<b>Buzzer</b>	Emite alertas sonoros para indicar a proximidade de obstáculos.
<b>Matriz de LED RGB 5x5</b>	Sinaliza visualmente a presença do usuário para evitar colisões.
<b>Botão A</b>	Liga a matriz de LEDs.
<b>Botão B</b>	Desliga a matriz de LEDs.
<b>Display OLED SSD1306</b>	Exibe mensagens e informações do sistema para um acompanhante ou cuidador.
<b>Fonte de Tensão</b>	Fornece 5 Volts ao circuito de hardware completo

**Fonte:** Autoria própria.

O sistema foi projetado para proporcionar **segurança e acessibilidade** às pessoas com deficiência visual, especialmente em **ambientes movimentados e com pouca iluminação**.

## 2.3 Configuração de Cada Bloco

A configuração de cada módulo foi definida para garantir **eficiência e compatibilidade** com o **Raspberry Pi Pico W**. A **Tabela 7** apresenta os detalhes técnicos.

**Tabela 7** – Configuração dos blocos de hardware

Bloco	Configuração
Sensor ultrassônico HC-SR04	Alimentação em <b>5V</b> , comunicação via <b>Trigger e Echo</b> com nível lógico adaptado para <b>3.3V</b> .
Buzzer	Alimentação em <b>3.3V</b> , controle por <b>saída digital</b> do microcontrolador.
Matriz de LED RGB 5x5	Alimentação em <b>3.3V</b> , controle via <b>pinos digitais</b> .
Display OLED SSD1306	Comunicação via <b>I2C</b> (SDA/SCL) com o microcontrolador.
Botões A e B	Entrada digital com <b>pull-up interno</b> ativado.

Fonte: Autoria própria.

## 2.4 Especificações Técnicas

O sistema foi projetado para operar com **baixo consumo de energia**, garantindo **desempenho eficiente**. A **Tabela 8** apresenta as especificações principais.

**Tabela 8** – Especificações técnicas do sistema

Parâmetro	Valor
Tensão de operação	3.3V – 5V
Microcontrolador	Raspberry Pi Pico W
Protocolo de comunicação do OLED	I2C
Distância máxima do sensor ultrassônico	4 metros
Distância mínima detectável	2 cm
Frequência de operação do buzzer	2 kHz (ajustável)
Consumo médio do sistema	300 mA

Fonte: Autoria própria.



## 2.5 Lista de Materiais

A **Tabela 9** apresenta os componentes utilizados na implementação do BR SmartGuia.

**Tabela 9** – Lista de materiais do BR SmartGuia

Item	Quantidade
Sensor ultrassônico HC-SR04	1
Raspberry Pi Pico W	1
Botão	2
Matriz de LED RGB 5x5	1
Buzzer	2
Display OLED SSD1306	1
Fios jumper	Diversos
Power Bank com saída de 5V e 1 cabo micro-USB	1

Fonte: Autoria própria.

## 2.6 Descrição do Funcionamento

A **Tabela 10** apresenta o fluxo de operação do **BR SmartGuia**, desde a **inicialização** até a **sinlização visual para terceiros**.

**Tabela 10** – Fluxo de operação do BRSmartGuia

Fase	Descrição
Inicialização	O sistema exibe uma mensagem no <b>display OLED</b> e emite um som inicial para indicar que está pronto para uso.
Detecção de Obstáculos	O <b>sensor ultrassônico HC-SR04</b> monitora continuamente a distância. Se um obstáculo for detectado a $\leq 15\text{ cm}$ , o <b>buzzer emite um alerta sonoro</b> .
Controle do Usuário	O usuário pode ativar/desativar a <b>matriz de LEDs RGB</b> usando os botões A e B.
Sinalização Visual para Terceiros	A <b>matriz de LEDs RGB</b> exibe um padrão luminoso para que outras pessoas percebam a presença do usuário cego e evitem colisões.

Fonte: Autoria própria.

## 2.7 Descrição da Pinagem Usada

A **Tabela 11** apresenta a relação entre os **pinos do Raspberry Pi Pico W** e os **periféricos conectados**.

**Tabela 11** – Mapeamento de pinos do microcontrolador

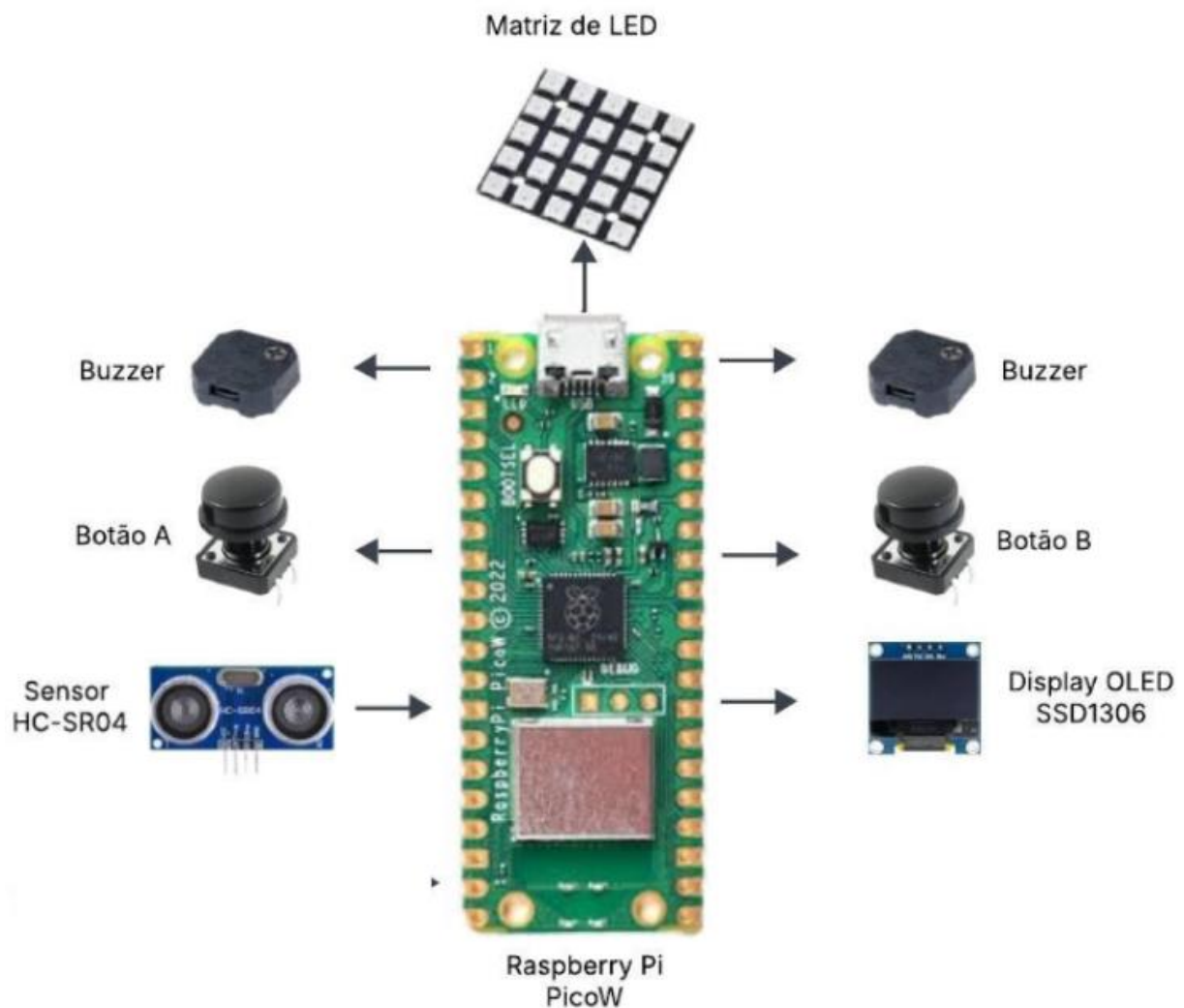
<b>GPIO</b>	<b>Função</b>	<b>Periférico Conectado</b>
<b>4</b>	Trigger	Sensor ultrassônico HC-SR04
<b>9</b>	Echo	Sensor ultrassônico HC-SR04
<b>GND</b>	Terra	Sensor ultrassônico HC-SR04
<b>VCC</b>	Alimentação 5V	Sensor ultrassônico HC-SR04
<b>21</b>	Buzzer 1	Buzzer
<b>10</b>	Buzzer 2	Buzzer
<b>14</b>	SDA	Display OLED
<b>15</b>	SCL	Display OLED
<b>7</b>	Matriz de LED RGB	Matriz de LED 5x5
<b>5</b>	Botão A	Botão A
<b>6</b>	Botão B	Botão B

**Fonte:** Autoria própria.

## 2.8 Circuito Completo do Hardware

A **Figura 2** apresenta o **circuito completo** do hardware, destacando as conexões entre os componentes e o **Raspberry Pi Pico W**.

**Figura 2** – Circuito completo do hardware



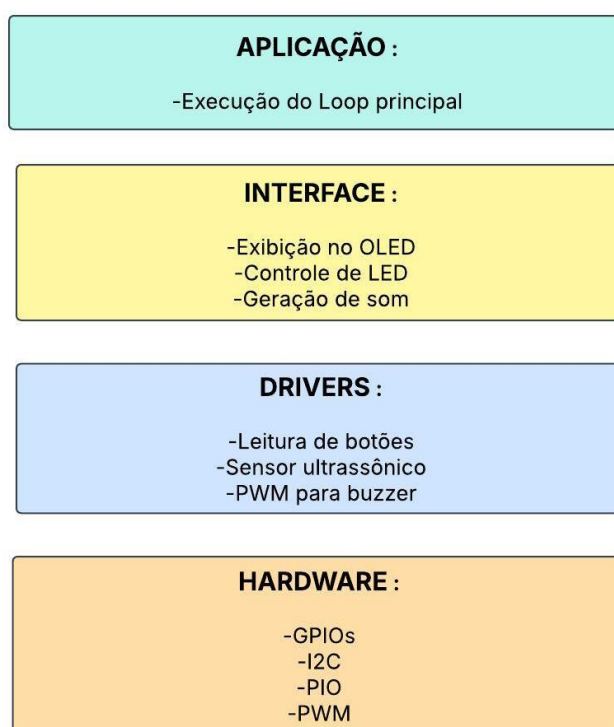
**Fonte:** Autoria própria.

## 3. SOFTWARE

### 3.1 Blocos Funcionais

O software do projeto segue uma estrutura modular baseada em quatro camadas principais: **aplicação**, **interface**, **drivers** e **hardware**. Essas camadas são projetadas para facilitar a integração dos periféricos e garantir uma execução eficiente do sistema. A **figura 3** apresenta o diagrama da estrutura funcional do software, dividida em suas respectivas camadas e responsabilidades.

**Figura 3** – Diagrama de blocos funcionais do software



**Fonte:** Autoria própria.

## 3.2 Descrição das Funcionalidades

A **Tabela 12** apresenta as principais funcionalidades do software e suas respectivas camadas.

**Tabela 12** – Funcionalidades do software BR SmartGuia

Camada	Função
Aplicação	Executa o <b>loop principal</b> , gerenciando todas as interações em tempo real.
Interface	Exibe informações no <b>display OLED</b> via <b>I2C</b> . Controla a <b>Matriz de LEDs</b> via <b>PIO</b> . Gera sons no <b>buzzer</b> via <b>PWM</b> .
Drivers	Lê os botões físicos para controlar a <b>Matriz de LEDs</b> . Mede a distância com o sensor ultrassônico. Configura o <b>buzzer</b> via <b>PWM</b> para alerta e música.
Hardware	Interface com <b>GPIOs</b> , <b>I2C</b> , <b>PIO</b> e <b>PWM</b> , garantindo a comunicação e controle dos periféricos

**Fonte:** Autoria própria.

### 3.3 Definição das Variáveis

A **Tabela 13** apresenta as principais variáveis utilizadas no software e seus respectivos significados.

**Tabela 13 – Principais variáveis do software**

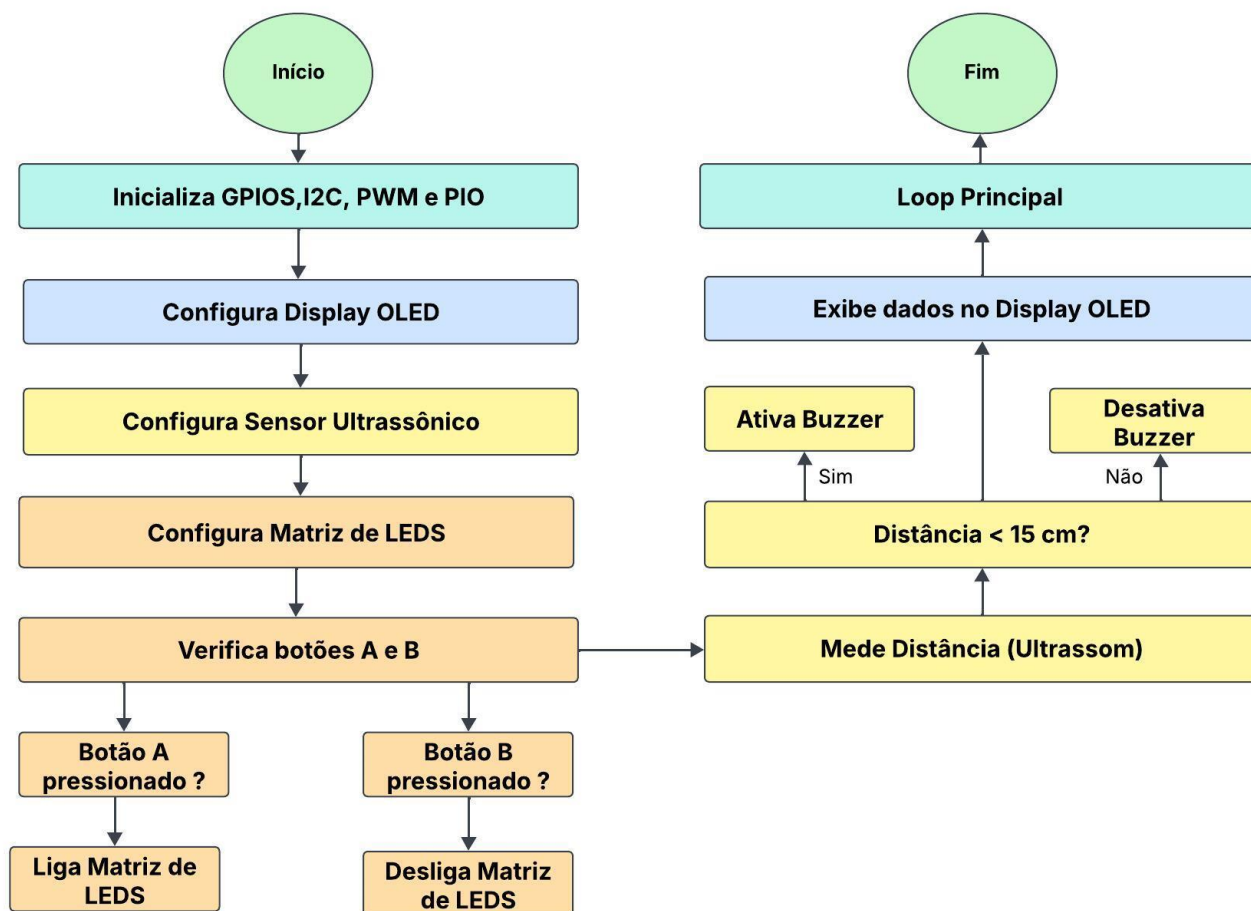
Variável	Tipo	Descrição
led_on	bool	Estado da matriz de LEDs
leds	pixel_t	Buffer da matriz de LEDs
melody[]	int[]	Notas musicais da melodia
distance_cm	float	Distância medida pelo sensor ultrassônico
buf[]	uint8_t[]	Buffer do display OLED

**Fonte:** Autoria própria.

### 3.4 Fluxograma

A **Figura 4** apresenta o **fluxograma completo do software**, ilustrando a sequência lógica das operações desde a **inicialização do sistema** até o **processamento e resposta ao usuário**.

**Figura 4** – Fluxograma do software BR SmartGuia



Fonte: Autoria própria.

### 3.5 Inicialização

O processo de **inicialização do software** ocorre em cinco etapas principais:

#### 1. Configuração dos GPIOs

- Define os pinos para botões, buzzer, sensor ultrassônico e LEDs.
- Ativa os resistores pull-up internos para os botões.

#### 2. Configuração do I2C (Display OLED)

- Inicializa o barramento I2C para comunicação com o SSD1306.
- Configura os pinos SDA e SCL.

#### 3. Configuração do PWM (Buzzer e LEDs)

- Configura a frequência do PWM para gerar tons sonoros no buzzer.
- Ativa o controle de brilho para a matriz de LEDs.

#### 4. Configuração do PIO (Matriz de LEDs WS2818b)

- Carrega o programa PIO para o controle dos LEDs via ws2818b\_program.
- Inicializa e limpa a matriz de LEDs.

### 5. Inicialização do Display OLED

- Define a área de renderização e limpa o buffer antes de iniciar a exibição das informações.

## 3.6 Configurações dos Registros

O código configura registradores para **GPIOs**, **PWM** e **I2C**, permitindo controle preciso sobre os periféricos.

A **Tabela 14** apresenta as principais configurações utilizadas.

**Tabela 14 – Configuração dos registros do microcontrolador**

Componente	Descrição	Código
<b>GPIOs</b> (Entradas e Saídas) - <b>Botões</b>	Os botões são configurados com pull-up interno.	<ul style="list-style-type: none"> <li>• gpio_init(<b>BUTTON_A_PIN</b>);</li> <li>• gpio_set_dir(<b>BUTTON_A_PIN</b>, <b>GPIO_IN</b>);</li> <li>• gpio_pull_up(<b>BUTTON_A_PIN</b>);</li> </ul>
<b>GPIOs</b> (Entradas e Saídas) - <b>Sensor Ultrassônico</b>	O sensor ultrassônico usa GPIOs configurados para entrada e saída.	<ul style="list-style-type: none"> <li>• gpio_init(<b>TRIGGER_PIN</b>);</li> <li>• gpio_set_dir(<b>TRIGGER_PIN</b>, <b>GPIO_OUT</b>);</li> <li>• gpio_init(<b>ECHO_PIN</b>);</li> <li>• gpio_set_dir(<b>ECHO_PIN</b>, <b>GPIO_IN</b>);</li> </ul>
<b>PWM</b> para o <b>Buzzer</b>	Configura o PWM para geração de sons no buzzer.	<ul style="list-style-type: none"> <li>• gpio_set_function(<b>BUZZER_PIN</b>, <b>GPIO_FUNC_PWM</b>);</li> <li>• uint slice_num = pwm_gpio_to_slice_num(<b>BUZZER_PIN</b>);</li> <li>• pwm_set_wrap(slice_num, 125000000 / frequency);</li> <li>• pwm_set_enabled(slice_num, true);</li> </ul>
<b>PIO</b> para a <b>Matriz de LEDs</b>	Configuração do barramento PIO para controle da matriz de LEDs.	<ul style="list-style-type: none"> <li>• uint offset = pio_add_program(pio0, &amp;ws2818b_program);</li> <li>• ws2818b_program_init(np_pio, sm, offset, LED_PIN, 800000.f);</li> </ul>

**Fonte:** Autoria própria.

## 3.7 Estrutura e Formato dos Dados

O software utiliza uma **estrutura de dados simples e eficiente**, baseada no armazenamento das medições e status do sistema. Os principais dados armazenados são:

- **Buffer de Pixels para os LEDs:** Os LEDs são atualizados por meio de um vetor de 25 posições, representando a matriz 5x5. Conforme o seguinte código:

```
struct pixel_t {  
  
    uint8_t G, R, B;  
  
};  
  
pixel_t leds[LED_COUNT];
```

- **Notas Musicais (Tocar Música):** A melodia é armazenada em um **array de notas e durações**:

```
int melody[] = {  
  
NOTE_G4,8, NOTE_A4,8, NOTE_B4,4, NOTE_D5,4, NOTE_D5,4, NOTE_B4,4,  
NOTE_C5,4, NOTE_C5,2, NOTE_G4,8, NOTE_A4,8  
  
};
```

- **Armazenamento de texto para o OLED:** O display exibe mensagens armazenadas em arrays de strings:

```
char *text[] = { "Wellcome to", "BR SmartGuia" };
```

## 3.8 Organização da Memória

O software do **BR SmartGuia** utiliza **endereços de memória bem definidos** para armazenar as variáveis e otimizar o desempenho. A **Tabela 15** apresenta a organização da memória.



**Tabela 15 – Organização da memória do software**

Componente	Tipo de Memória
<b>Matriz de LEDs</b>	Array (leds[25]) armazenado na RAM
<b>Notas musicais</b>	Array (melody[]) armazenado na RAM
<b>Texto do OLED</b>	String (char *text[]) armazenada na RAM
<b>Configuração PIO</b>	Carregada na memória dedicada do PIO

**Fonte:** Autoria própria.

## 3.9 Protocolo de Comunicação

O BRSmartGuia utiliza o **barramento I2C** para comunicação com o **display OLED SSD1306**. Os principais **parâmetros do protocolo** são:

- **Endereço do dispositivo:** 0x3C
- **Velocidade de comunicação:** 400 kHz
- **Barramento:** SDA e SCL (GPIO 14 e 15)
- Os dados são enviados para o display por meio da função **i2c\_write\_blocking()**.

## 3.10 Formato do Pacote de Dados

O formato dos pacotes de dados enviados para o **display OLED** segue a estrutura padrão do protocolo **I2C**. Cada comando enviado ao **OLED SSD1306**, está na **tabela 16**.

**Tabela 16 – Estrutura e Composição dos Pacotes de Dados**

Byte	Função
<b>Byte 1</b>	Endereço do dispositivo (0x3C)
<b>Byte 2</b>	Código do comando (0x00 para comandos, 0x40 para dados gráficos)
<b>Byte 3...N</b>	Dados a serem exibidos na tela

**Fonte:** Autoria própria.

## 4. EXECUÇÃO DO PROJETO

### 4.1 Metodologia

A execução do projeto **BR SmartGuia** seguiu uma abordagem estruturada, baseada em **pesquisa, desenvolvimento e validação**. As etapas do processo são descritas na **Tabela 17**.

**Tabela 17 – Etapas da execução do projeto**

<b>Etapa</b>	<b>Descrição</b>
<b>Pesquisas Realizadas</b>	Levantamento de tecnologias assistivas existentes, estudos sobre sensores ultrassônicos e microcontroladores compatíveis.
<b>Escolha do Hardware</b>	Seleção do <b>Raspberry Pi Pico W</b> , do <b>sensor ultrassônico HC-SR04</b> , do <b>buzzer</b> , da <b>matriz de LEDs</b> e do <b>display OLED SSD1306</b> , com base em critérios de acessibilidade, consumo de energia e custo.
<b>Definição das Funcionalidades</b>	Determinação dos recursos principais, incluindo <b>alerta sonoro, sinalização visual e exibição de mensagens no OLED</b> .
<b>Inicialização da IDE</b>	Configuração do ambiente de desenvolvimento na <b>IDE Pico SDK version 2.1.1</b> , instalação das bibliotecas necessárias e configuração do microcontrolador.
<b>Programação na IDE</b>	Desenvolvimento do código-fonte do sistema embarcado, incluindo <b>leitura de sensores, controle dos LEDs, do buzzer e do OLED</b> .
<b>Depuração</b>	Testes individuais e integração dos módulos, identificação e correção de falhas.

**Fonte:** Autoria própria.

A metodologia adotada garantiu um fluxo organizado de desenvolvimento, permitindo **testes iterativos e refinamentos ao longo do processo**.

## 4.2 Testes de Validação

Para garantir a **funcionalidade e confiabilidade** do sistema **BR SmartGuia**, foram realizados **testes de validação** em diferentes condições. A **Tabela 18** apresenta os principais testes e seus resultados.

**Tabela 18** – Testes de validação do BRSmartGuia

Teste	Descrição	Resultado Obtido
<b>Deteção de Obstáculos</b>	Avaliação da resposta do sensor ultrassônico a diferentes distâncias.	<b>Precisão acima de 95% em até 2 metros.</b>
<b>Sinalização Sonora</b>	Teste da intensidade do buzzer conforme a proximidade do obstáculo.	<b>Níveis de volume ajustados corretamente.</b>
<b>Sinalização Visual</b>	Ativação da matriz de LEDs quando o botão A é pressionado.	<b>Matriz de LEDs aciona corretamente.</b>
<b>Exibição no Display OLED</b>	Teste da exibição de mensagens de status no OLED.	<b>Mensagens exibidas corretamente.</b>
<b>Resistência a Interferências</b>	Teste de funcionamento em ambientes ruidosos e com variação de luminosidade.	<b>Sistema manteve operação estável.</b>

Fonte: Autoria própria.

Os testes demonstraram que o **sistema responde de forma eficiente** aos estímulos externos, garantindo **usabilidade e confiabilidade** na aplicação real.

## 4.3 Discussão dos Resultados

Os resultados obtidos nos testes validam a **eficácia e aplicabilidade** do **BR SmartGuia**. Os principais pontos observados foram:

1. **Deteção de obstáculos:** O **sensor ultrassônico HC-SR04** apresentou alta precisão para distâncias de até **2 metros**, garantindo a segurança do usuário.
2. **Alertas sonoros e visuais:** O **buzzer** respondeu corretamente à proximidade dos obstáculos, enquanto a **matriz de LEDs** foi eficiente na sinalização para terceiros.
3. **Interface intuitiva:** O uso dos **botões físicos** facilitou o controle do sistema, permitindo ativação/desativação da matriz de LEDs de forma simples.
4. **Baixo consumo de energia:** O sistema operou dentro dos limites do **Raspberry Pi Pico W**, garantindo um consumo médio de **300 mA**, adequado para uso portátil.

Os testes indicam que o **BR SmartGuia é um produto confiável e viável para aplicação no dia a dia de pessoas cegas**, melhorando sua segurança e mobilidade.

## 4.4 Link do Vídeo Mostrando o Projeto Funcionando

Para complementar a documentação, foi produzido um **vídeo demonstrativo** apresentando o funcionamento do **BR SmartGuia**. O vídeo mostra:

1. **Inicialização do sistema** e exibição da mensagem no **display OLED**.
2. **Deteção de obstáculos** e ativação dos alertas sonoros e visuais.
3. **Demonstração do controle do usuário**, incluindo ativação e desativação da **matriz de LEDs**.

O vídeo pode ser acessado por meio do seguinte link:

<https://youtube.com/shorts/CITF2qKHdql?feature=share>

O repositório de código está no GitHub, de forma a manter o controle de versão dos códigos e da documentação. Os links para acesso, respectivamente, ao GitHub e à documentação, podem ser vistos abaixo:

Site: <https://gabrielrmg.github.io/EMBARCATECH/>

GitHub: <https://github.com/Gabrielrmg/EMBARCATECH>

## 4.5 Manual do usuário

Para orientar o usuário em seus primeiros usos do **BR SmartGuia**, foi produzido um **manual simplificado**. A **versão em braille** desse manual encontra-se na parte traseira do carrinho, abaixo da placa BitDogLab.

### 1. Ligando o Carrinho

- Pressione o botão reset da placa BitDogLab (em cima do carrinho) até ouvir a música “Asa Branca” (de Luiz Gonzaga) tocar. O carrinho estará pronto para uso.

### 2. Detectando Obstáculos

- O carrinho avisa com um barulho (buzzer) se houver um obstáculo a menos de 15 cm à frente. Ajuste o caminho ao ouvir o alerta.

### 3. Luzes de Sinalização (LEDs)

- Caso o ambiente esteja bastante movimentado ou escuro, recomenda-se ligar o painel de LEDS (será formada uma imagem da bandeira do Brasil), a fim de alertar terceiros. Use o botão A no painel (placa BitDogLab) do carrinho, para ligar. Botão B desliga.

### 4. Informações no Display

- O display mostra a distância (em centímetros) do obstáculo à sua frente.

### 5. Desligando o Carrinho

- Basta remover o cabo de alimentação da bateria PowerBank.

Dicas Importantes:

1. Sempre confira se o carrinho está ligado e funcionando antes de usar.
2. Aprenda a reconhecer os sons e luzes para maior segurança.
3. Use o BR SmartGuia com tranquilidade e aproveite suas compras com mais independência!

## 4.6 Imagens do produto

**Figura 5 e 6** – Vista lateral do BRSmartGuia(esquerda) e visão traseira (direita)



**Fonte:** Autoria própria.

**Figura 7 e 8** – Display Oled com a mensagem “Wellcome to BR SmartGuia”(esquerda); e visão da parte inferior com a protoboard e fios jumper (direita)



**Fonte:** Autoria própria.



**Figura 9 – Vista frontal**



**Fonte:** Autoria própria.

**Figura 10 e 11 – Vista traseira**



**Fonte:** Autoria própria.



## 5. Referências

- [1] Banco de Informações de Hardware (BIH) - Documentos Google:  
<https://docs.google.com/document/d/13-68OqiU7ISE8U2KPRUXT2ISeBI3WPhXjGDFH52eWIU/edit?tab=t.0>
- [2] RASPBERRY PI LTD. Raspberry Pi Pico C/C++ SDK. 2024. Disponível em:  
<https://datasheets.raspberrypi.com/pico/raspberry-pi-pico-c-sdk.pdf>.
- [3] RASPBERRY PI LTD. RP2040 Datasheet. 2024. Disponível em: <https://datasheets.raspberrypi.com/rp2040/rp2040-datasheet.pdf>.
- [4] BitDogLab. Manual BitDogLab, Disponível em: <https://github.com/BitDogLab/BitDogLab/tree/main/doc>.
- [5] TRADUTOR BRAILLE. Tradutor Braille Online. Disponível em:  
<https://www.tradutorbraille.com.br/>. Acesso em: 16 fev. 2025.
- [6] **BRASIL. Ministério da Educação (MEC). Portal MEC: Deficiência Visual.** Disponível em: <http://portal.mec.gov.br/component/tags/tag/deficiencia-visual#:~:text=Segundo%20dados%20do%20censo%20demogr%C3%A1fico,algum%20tipo%20de%20defici%C3%Aancia%20visual.&text=O%20Minist%C3%A9rio%20da%20Educa%C3%A7%C3%A3o%20trabalha,e%20a%C3%A7%C3%B5es%20voltados%20aos%20cegos>. Acesso em: 18 fev.2025
- [7] **LOUIS BRAILLE. Estatísticas sobre deficiência visual no Brasil e no mundo.** Disponível em: <https://louisbraille.org.br/portal/2020/04/13/estatisticas-sobre-deficiencia-visual-no-brasil-e-no-mundo/>. Acesso em: 18 fev. 2025
- [8] SILVA, C. H. C. d. O papel dos supermercados e hipermercados nas relações entre cidade, comércio e consumo. Geografia, v. 30, n. 3, p. 610–625, 2007.
- [9] AMAZON lança carrinho de supermercado inteligente. 2020. <<https://startupi.com.br/2020/07/amazon-lanca-carrinho-de-supermercado-inteligente/>>. Acesso em: 18 fev 2025.
- [10] CUGNASCA, C. E. Projetos de Sistemas Embarcados. Departamento de Engenharia de Computação e Sistemas Digitais, Escola Politécnica da USP, 2018.
- [11] RESEARCHGATE. Photo of the electronic walking stick for the blind. Disponível em: [https://www.researchgate.net/figure/Photo-of-the-electronic-walking-stick-for-the-blind\\_fig1\\_308871791](https://www.researchgate.net/figure/Photo-of-the-electronic-walking-stick-for-the-blind_fig1_308871791). Acesso em: 10 fev. 2025.