Nova Página

- Disponibilização do ícone de acessibilidade

Ally - Web Accessibility & Usability adiciona:
botão flutuante
contraste
aumento de fonte
leitura
teclado
acessibilidade visual

- Recursos obrigatórios de acessibilidade visual

toolbar acessível
navegação
melhorias WCAG

- Tradução e acessibilidade para pessoas surdas (Libras)

VLibras serve para tornar o site acessível para pessoas surdas que utilizam Libras (Língua Brasileira de Sinais).
Ele funciona assim:
o usuário acessa o site
clica no botão do VLibras
um avatar/intérprete virtual aparece
o texto da página é traduzido para Libras em animação.
Isso ajuda principalmente:
pessoas surdas
pessoas com deficiência auditiva
usuários que têm Libras como principal forma de comunicação.

- Leitura automática e apoio para pessoas com deficiência visual

Leitor de texto (voz) por Emerson Andrade:
Permite leitura automática
Ideal para deficiência visual

//--- CÓDIGO ---//

<!-- ==
TTS ACESSÍVEL PARA WORDPRESS / ELEMENTOR
- Leitura em PT-BR
- Play / Pause / Stop

```
    - Seleção de voz
    - Velocidade
    - Destaque da palavra lida
    - Botão flutuante moderno
========================================================== -->

<div id="tts-container">

   <!-- BOTÃO -->
   <button
      id="tts-toggle"
      aria-label="Abrir leitor de conteúdo">

      <svg
         xmlns="http://www.w3.org/2000/svg"
         width="28"
         height="28"
         viewBox="0 0 24 24"
         fill="none"
         stroke="currentColor"
         stroke-width="2"
         stroke-linecap="round"
         stroke-linejoin="round">

         <path d="M11 5 6 9H2v6h4l5 4z"/>
         <path d="M15.54 8.46a5 5 0 0 1 0 7.07"/>
         <path d="M19.07 4.93a10 10 0 0 1 0 14.14"/>

      </svg>

   </button>

   <!-- PLAYER -->
   <div id="tts-player">

      <div class="tts-header">
         <strong>Leitor de Conteúdo</strong>
      </div>

      <!-- CONTROLES -->
      <div class="tts-controls">

         <button onclick="iniciarLeitura()">
            ▶
         </button>

         <button onclick="pausarLeitura()">
```

```
            
          </button>

          <button onclick="pararLeitura()">
            
          </button>

        </div>

        <!-- VELOCIDADE -->
        <div class="tts-options">

          <label>
            Velocidade
          </label>

          <input
            type="range"
            id="tts-rate"
            min="0.5"
            max="2"
            step="0.1"
            value="1"
          >

          <!-- VOZ -->

          <label>
            Voz
          </label>

          <select id="tts-voice"></select>

        </div>

      </div>

    </div>

<!-- ========================================================
    CSS
======================================================== -->

<style>

/* CONTAINER */
```

```
#tts-container{
    position:fixed;
    bottom:120px;
    right:20px;
    z-index:999999;
    font-family:Arial,sans-serif;
}

/* BOTÃO FLUTUANTE */

#tts-toggle{
    width:64px;
    height:64px;

    border:none;
    border-radius:50%;

    background:#ffffff;

    display:flex;
    align-items:center;
    justify-content:center;

    cursor:pointer;

    box-shadow:
        0 8px 24px rgba(0,0,0,.18);

    transition:.2s;
}

#tts-toggle:hover{
    transform:scale(1.05);
}

/* ÍCONE */

#tts-toggle svg{
    stroke:#1351b4 !important;
}

/* PLAYER */

#tts-player{
    width:280px;

    background:#ffffff;
```

```
    border-radius:16px;

    padding:16px;

    margin-top:10px;

    box-shadow:
        0 8px 24px rgba(0,0,0,.15);

    display:none;
}

/* HEADER */

.tts-header{
    margin-bottom:15px;
    font-size:16px;
}

/* CONTROLES */

.tts-controls{
    display:flex;
    gap:10px;
    margin-bottom:15px;
}

.tts-controls button{

    flex:1;

    border:none;

    padding:12px;

    border-radius:10px;

    cursor:pointer;

    background:#f3f4f6;

    font-size:18px;

    transition:.2s;
}
```

```
.tts-controls button:hover{
    background:#e5e7eb;
}

/* LABELS */

.tts-options label{
    display:block;
    margin-top:10px;
    margin-bottom:5px;
    font-size:14px;
}

/* INPUTS */

#tts-rate,
#tts-voice{
    width:100%;
}

/* DESTACAR PALAVRA */

.tts-word{
    display:inline-block;
}

/*.tts-highlight{
    display:inline-block !important;
    background:#ffeb3b !important;
    color:#000 !important;
    padding:2px 6px !important;
    border-radius:6px !important;
    line-height:1.4 !important;
    box-decoration-break:clone;
    -webkit-box-decoration-break:clone;

} */

.tts-highlight{
    display:inline-block !important;
    background:#fde68a !important;
    color:#111827 !important;
    padding:2px 6px !important;
    border-radius:6px !important;
}

/* RESPONSIVO */
```

```
@media(max-width:768px){

   #tts-container{
      right:15px;
      bottom:100px;
   }

   #tts-toggle{
      width:58px;
      height:58px;
   }

}

</style>

<!-- ========================================================
    SCRIPT
========================================================= -->

<script>

let utterance;
let voices = [];

let palavras = [];
let palavraAtual = -1;

const toggle =
   document.getElementById('tts-toggle');

const player =
   document.getElementById('tts-player');

const voiceSelect =
   document.getElementById('tts-voice');

const rateInput =
   document.getElementById('tts-rate');

/* ========================================================
  ABRIR / FECHAR PLAYER
========================================================= */

toggle.addEventListener('click', () => {
```

```
    player.style.display =
        player.style.display === 'block'
            ? 'none'
            : 'block';

});

/* ========================================================
   CARREGAR VOZES
======================================================== */

function carregarVozes(){

    voices =
        speechSynthesis.getVoices();

    voiceSelect.innerHTML = '';

    voices
        .filter(v => v.lang.includes('pt'))
        .forEach((voice, index) => {

            const option =
                document.createElement('option');

            option.value = index;

            option.textContent =
                `${voice.name} (${voice.lang})`;

            voiceSelect.appendChild(option);

        });

}

speechSynthesis.onvoiceschanged =
    carregarVozes;

carregarVozes();

/* ========================================================
   OBTER CONTEÚDO
======================================================== */

function obterElementoConteudo(){
```

```
    return (
        document.querySelector('article') ||
        document.querySelector('.entry-content') ||
        document.querySelector('main')
    );

}

/* ========================================================
   PREPARAR PARÁGRAFOS
======================================================== */

function prepararConteudo(){

    const container =
        obterElementoConteudo();

    if(!container) return '';

    const paragrafos =
        container.querySelectorAll('p');

    palavras = [];

    paragrafos.forEach((p) => {

        const texto =
            p.innerText;

        const lista =
            texto.split(/\s+/);

        p.innerHTML =
            lista.map((palavra) => {

                palavras.push(palavra);

                const index =
                    palavras.length - 1;

                return `
                    <span
                        class="tts-word"
                        data-index="${index}">
                        ${palavra}
                    </span>
                `;
```

```
        }).join(' ');

    });

    return palavras.join(' ');

}

/* ========================================================
   LIMPAR DESTAQUE
======================================================== */

function limparDestaque(){

    document
        .querySelectorAll('.tts-highlight')
        .forEach((el) => {

            el.classList.remove(
                'tts-highlight'
            );

        });

}

/* ========================================================
   DESTACAR PALAVRA
======================================================== */

function destacarPalavra(index){

    limparDestaque();

    const elemento =
        document.querySelector(
            `.tts-word[data-index="${index}"]`
        );

    if(elemento){

        elemento.classList.add(
            'tts-highlight'
        );

        elemento.scrollIntoView({
```

```
            behavior:'smooth',
            block:'center'
        });

    }

}

/* ========================================================
   INICIAR LEITURA
======================================================== */

function iniciarLeitura(){

    pararLeitura();

    const texto =
        prepararConteudo();

    utterance =
        new SpeechSynthesisUtterance(texto);

    utterance.lang = 'pt-BR';

    utterance.rate =
        parseFloat(rateInput.value);

    const voz =
        voices[voiceSelect.value];

    if(voz){
        utterance.voice = voz;
    }

    utterance.onboundary =
        function(event){

        if(event.charIndex !== undefined){

            const textoAtual =
                texto.substring(
                    0,
                    event.charIndex
                );

            const indice =
                textoAtual
```

```
            .split(/\s+/)
            .length - 1;

        if(indice !== palavraAtual){

            palavraAtual = indice;

            destacarPalavra(
                palavraAtual
            );

        }

    }

};

utterance.onend =
    function(){

    limparDestaque();

};

speechSynthesis.speak(
    utterance
);

}

/* ========================================================
  PAUSAR
======================================================== */

function pausarLeitura(){

    if(speechSynthesis.speaking){

        if(speechSynthesis.paused){

            speechSynthesis.resume();

        }else{

            speechSynthesis.pause();

        }
```

```
    }

}

/* ========================================================
   PARAR
======================================================== */

function pararLeitura(){

    speechSynthesis.cancel();

    limparDestaque();

}

</script>
```

Revision #3
Created 2026-05-13 13:24:56 UTC by WILIAM THIAGO BRAZ DA CUNHA
Updated 2026-05-19 14:58:21 UTC by Emerson Andrade