<?php

// Exit if accessed directly

if ( !defined( 'ABSPATH' ) ) exit;

// BEGIN ENQUEUE PARENT ACTION

// AUTO GENERATED - Do not modify or remove comment markers above or below:

if ( !function\_exists( 'chld\_thm\_cfg\_locale\_css' ) ):

function chld\_thm\_cfg\_locale\_css( $uri ){

if ( empty( $uri ) && is\_rtl() && file\_exists( get\_template\_directory() . '/rtl.css' ) )

$uri = get\_template\_directory\_uri() . '/rtl.css';

return $uri;

}

endif;

add\_filter( 'locale\_stylesheet\_uri', 'chld\_thm\_cfg\_locale\_css' );

if ( !function\_exists( 'chld\_thm\_cfg\_parent\_css' ) ):

function chld\_thm\_cfg\_parent\_css() {

wp\_enqueue\_style( 'chld\_thm\_cfg\_parent', trailingslashit( get\_template\_directory\_uri() ) . 'style.css', array( 'bootstrap' ) );

}

endif;

add\_action( 'wp\_enqueue\_scripts', 'chld\_thm\_cfg\_parent\_css', 10 );

function lector\_de\_noticias\_shortcode() {

// Reproductor visual con controles y estilos

$output = '<div id="lector-player" class="lector-player">

<div class="lector-progress-container">

<span id="lector-current-time">0:00</span>

<input type="range" id="lector-progress" min="0" max="100" value="0"/>

<span id="lector-total-time">4:03</span>

</div>

<div class="lector-controls">

<button id="lector-shuffle" class="lector-btn" title="Aleatorio"><i class="fas fa-random"></i></button>

<button id="lector-prev" class="lector-btn" title="Anterior"><i class="fas fa-step-backward"></i></button>

<button id="lector-rewind" class="lector-btn" title="Rebobinar"><i class="fas fa-backward"></i></button>

<button id="lector-play" class="lector-btn lector-play-main" title="Reproducir"><i class="fas fa-play"></i></button>

<button id="lector-forward" class="lector-btn" title="Avanzar"><i class="fas fa-forward"></i></button>

<button id="lector-next" class="lector-btn" title="Siguiente"><i class="fas fa-step-forward"></i></button>

<button id="lector-repeat" class="lector-btn" title="Repetir"><i class="fas fa-redo"></i></button>

</div>

</div>

<style>

.lector-player { max-width:380px; margin:40px auto 20px auto; padding:14px 18px; background:#fff; border-radius:16px; box-shadow:0 4px 14px #0001;}

.lector-controls { display:flex; gap:12px; justify-content:center; margin-top:16px;}

.lector-btn { background:none; border:none; font-size:1.6rem; color:#232323; cursor:pointer; transition:color .16s;}

.lector-play-main { background:#232323; color:#fff; border-radius:50%; width:38px; height:38px; display:flex; align-items:center; justify-content:center;}

.lector-btn:hover, .lector-play-main:hover { color:#0072ff; background:#eaeaea;}

.lector-progress-container { display:flex; align-items:center; gap:10px;}

#lector-progress { flex:1; accent-color:#0072ff; margin:0 5px;}

#lector-current-time, #lector-total-time { width:38px; text-align:center;}

</style>

<script>

// Obtiene el texto visible desde todos los elementos debajo del reproductor

function obtenerTextoDebajo() {

var reproductor = document.getElementById("lector-player");

var texto = "";

var node = reproductor.nextElementSibling;

while (node) {

// Excluye script/style y lee solo texto visible

if (node.nodeType === 1 && node.style && node.style.display !== "none" && node.innerText) {

texto += node.innerText + " ";

}

node = node.nextElementSibling;

}

return texto.trim();

}

var synth = window.speechSynthesis;

var utterance = null;

var intervalo = null, tiempoInicio = 0, duracionTotal = 0;

function contarPalabras(text) {

return text.split(/\s+/).length;

}

function estimarDuracion(text) {

return Math.ceil(contarPalabras(text)/2.5); // segundos aproximados

}

function iniciarProgreso() {

clearInterval(intervalo);

tiempoInicio = 0;

var currentTime = document.getElementById("lector-current-time");

var totalTime = document.getElementById("lector-total-time");

var progress = document.getElementById("lector-progress");

totalTime.textContent = duracionTotal > 60 ? Math.floor(duracionTotal/60)+":"+(duracionTotal%60).toString().padStart(2,"0") : "0:"+duracionTotal.toString().padStart(2,"0");

intervalo = setInterval(function() {

tiempoInicio++;

var m = Math.floor(tiempoInicio/60);

var s = (tiempoInicio%60).toString().padStart(2,"0");

currentTime.textContent = m+":"+s;

progress.value = Math.min(100, parseInt((tiempoInicio/duracionTotal)\*100));

if (tiempoInicio >= duracionTotal) pararLectura();

}, 1000);

}

function pararLectura() {

clearInterval(intervalo);

tiempoInicio = 0;

document.getElementById("lector-current-time").textContent = "0:00";

document.getElementById("lector-progress").value = 0;

}

function reproducirLectura() {

var texto = obtenerTextoDebajo();

if (!texto) return;

duracionTotal = estimarDuracion(texto);

utterance = new SpeechSynthesisUtterance(texto);

utterance.lang = "es-ES";

utterance.onstart = iniciarProgreso;

utterance.onend = pararLectura;

synth.speak(utterance);

}

document.addEventListener("DOMContentLoaded", function() {

var btnPlay = document.getElementById("lector-play");

if (btnPlay) btnPlay.onclick = reproducirLectura;

document.getElementById("lector-rewind").onclick = function(){

pararLectura();

synth.cancel();

reproducirLectura();

};

document.getElementById("lector-forward").onclick = function(){

synth.pause();

clearInterval(intervalo);

};

document.getElementById("lector-repeat").onclick = function(){

pararLectura();

synth.cancel();

reproducirLectura();

};

document.getElementById("lector-prev").onclick = pararLectura;

document.getElementById("lector-next").onclick = pararLectura;

document.getElementById("lector-shuffle").onclick = pararLectura;

});

</script>';

return $output;

}

add\_shortcode('lector\_de\_noticias', 'lector\_de\_noticias\_shortcode');

// END ENQUEUE PARENT ACTION