<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Yo.fan Views Booster</title>

<style>

\* {

box-sizing: border-box;

margin: 0;

padding: 0;

font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

}

body {

background: linear-gradient(135deg, #1a2a6c, #b21f1f, #fdbb2d);

min-height: 100vh;

display: flex;

justify-content: center;

align-items: center;

padding: 20px;

}

.container {

background-color: rgba(255, 255, 255, 0.95);

border-radius: 15px;

box-shadow: 0 10px 30px rgba(0, 0, 0, 0.3);

width: 100%;

max-width: 500px;

padding: 30px;

text-align: center;

}

h1 {

color: #333;

margin-bottom: 20px;

font-weight: 600;

background: linear-gradient(to right, #6e8efb, #a777e3);

-webkit-background-clip: text;

-webkit-text-fill-color: transparent;

}

.description {

color: #666;

margin-bottom: 30px;

line-height: 1.5;

}

.input-group {

margin-bottom: 25px;

text-align: left;

}

label {

display: block;

margin-bottom: 8px;

color: #444;

font-weight: 500;

}

input {

width: 100%;

padding: 12px 15px;

border: 2px solid #ddd;

border-radius: 8px;

font-size: 16px;

transition: border-color 0.3s;

}

input:focus {

border-color: #6e8efb;

outline: none;

}

button {

background: linear-gradient(to right, #6e8efb, #a777e3);

color: white;

border: none;

padding: 14px 25px;

border-radius: 8px;

font-size: 16px;

cursor: pointer;

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

font-weight: 600;

width: 100%;

margin-bottom: 15px;

}

button:hover {

transform: translateY(-2px);

box-shadow: 0 5px 15px rgba(0, 0, 0, 0.2);

}

button:active {

transform: translateY(0);

}

.secondary-btn {

background: linear-gradient(to right, #f5a434, #f5d042);

}

.disclaimer {

margin-top: 25px;

padding: 15px;

background-color: #f8f9fa;

border-radius: 8px;

font-size: 14px;

color: #666;

text-align: left;

}

.success-message {

margin-top: 20px;

padding: 10px;

background-color: #d4edda;

color: #155724;

border-radius: 5px;

display: none;

}

.tabs-counter {

margin-top: 20px;

font-weight: 500;

color: #555;

font-size: 18px;

}

.error-message {

margin-top: 20px;

padding: 10px;

background-color: #f8d7da;

color: #721c24;

border-radius: 5px;

display: none;

}

.instructions {

margin-top: 25px;

text-align: left;

padding: 15px;

background-color: #e9f7fe;

border-radius: 8px;

}

.instructions h3 {

margin-bottom: 10px;

color: #1a73e8;

}

.instructions ol {

padding-left: 20px;

}

.instructions li {

margin-bottom: 10px;

}

.progress-container {

width: 100%;

height: 20px;

background-color: #f0f0f0;

border-radius: 10px;

margin: 20px 0;

overflow: hidden;

display: none;

}

.progress-bar {

height: 100%;

background: linear-gradient(to right, #6e8efb, #a777e3);

border-radius: 10px;

width: 0%;

transition: width 0.3s ease;

}

.speed-control {

margin: 20px 0;

text-align: left;

}

.speed-control label {

display: block;

margin-bottom: 10px;

}

.speed-control input {

width: 100%;

}

</style>

</head>

<body>

<div class="container">

<h1>Yo.fan Views Booster</h1>

<p class="description">

Enter your yo.fan URL and the number of views you want to simulate.

</p>

<div class="input-group">

<label for="url-input">Yo.fan URL:</label>

<input type="url" id="url-input" placeholder="https://yo.fan/your\_page" value="https://yo.fan/">

</div>

<div class="input-group">

<label for="views-input">Number of views:</label>

<input type="number" id="views-input" placeholder="Enter number of views" min="1" max="50" value="5">

</div>

<div class="speed-control">

<label for="speed">Opening Speed (ms delay between tabs):</label>

<input type="range" id="speed" min="100" max="5000" value="1000">

<span id="speed-value">1000ms</span>

</div>

<button id="open-tabs">Start Increasing Views</button>

<button id="test-popup" class="secondary-btn">Test Popup Permission</button>

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

<div class="progress-bar" id="progress-bar"></div>

</div>

<div class="tabs-counter" id="tabs-counter">

Views attempted: <span id="count">0</span>/<span id="total">0</span>

</div>

<div class="success-message" id="success-message">

Process started. Please allow pop-ups for this site if they are blocked.

</div>

<div class="error-message" id="error-message">

Error: Popups are being blocked. Please allow popups and try again.

</div>

<div class="instructions">

<h3>Important Instructions:</h3>

<ol>

<li>You must allow popups for this site when your browser asks</li>

<li>For best results, use this tool on a desktop computer</li>

<li>Keep this tab open until the process completes</li>

<li>You may need to interact with each tab for it to fully load</li>

<li>Some browsers may still block multiple popups</li>

</ol>

</div>

<div class="disclaimer">

<strong>Note:</strong> This tool simulates views by opening multiple tabs.

Due to browser security restrictions, there are limitations to how many tabs can be opened automatically.

Results may vary based on your browser and settings.

</div>

</div>

<script>

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

const urlInput = document.getElementById('url-input');

const viewsInput = document.getElementById('views-input');

const speedInput = document.getElementById('speed');

const speedValue = document.getElementById('speed-value');

const openTabsButton = document.getElementById('open-tabs');

const testPopupButton = document.getElementById('test-popup');

const successMessage = document.getElementById('success-message');

const errorMessage = document.getElementById('error-message');

const countElement = document.getElementById('count');

const totalElement = document.getElementById('total');

const progressContainer = document.getElementById('progress-container');

const progressBar = document.getElementById('progress-bar');

// Update speed value display

speedInput.addEventListener('input', function() {

speedValue.textContent = this.value + 'ms';

});

// Set maximum views limit

viewsInput.addEventListener('change', function() {

if (this.value > 50) {

this.value = 50;

alert('Maximum allowed views is 50 due to browser limitations.');

}

});

// Test popup permission

testPopupButton.addEventListener('click', function() {

const testPopup = window.open('', 'testPopup', 'width=100,height=100');

if (testPopup) {

testPopup.document.write('<h3>Popup test successful!</h3>');

setTimeout(() => {

testPopup.close();

alert('Popup test successful! You can now use the view booster.');

}, 1000);

} else {

errorMessage.style.display = 'block';

errorMessage.textContent = 'Popups are blocked. Please allow popups for this site and try again.';

}

});

openTabsButton.addEventListener('click', function() {

const url = urlInput.value.trim();

let views = parseInt(viewsInput.value);

const speed = parseInt(speedInput.value);

// Validate inputs

if (!url) {

alert('Please enter a valid URL');

return;

}

if (isNaN(views) || views < 1) {

alert('Please enter a valid number of views');

return;

}

// Ensure it's a yo.fan URL

if (!url.includes('yo.fan')) {

if (!confirm('This does not appear to be a yo.fan URL. Continue anyway?')) {

return;

}

}

// Limit the number of views to 50

views = Math.min(views, 50);

// Hide any previous messages

successMessage.style.display = 'none';

errorMessage.style.display = 'none';

// Reset counter and show progress

let viewsAttempted = 0;

countElement.textContent = viewsAttempted;

totalElement.textContent = views;

progressContainer.style.display = 'block';

progressBar.style.width = '0%';

// Show success message

successMessage.style.display = 'block';

// Change button text

openTabsButton.textContent = 'Processing...';

openTabsButton.disabled = true;

// Function to open tabs with delay

function openTabWithDelay() {

if (viewsAttempted < views) {

try {

// Try to open in new tab

const newWindow = window.open(url, '\_blank' + viewsAttempted);

if (newWindow) {

viewsAttempted++;

countElement.textContent = viewsAttempted;

const progress = (viewsAttempted / views) \* 100;

progressBar.style.width = progress + '%';

// Close the tab after a short delay to avoid too many open tabs

setTimeout(() => {

try {

newWindow.close();

} catch (e) {

// Tab might not be closable due to browser restrictions

}

}, 2000);

} else {

errorMessage.style.display = 'block';

errorMessage.textContent = 'Popup blocked after ' + viewsAttempted + ' views. Please allow popups.';

openTabsButton.textContent = 'Start Increasing Views';

openTabsButton.disabled = false;

return;

}

} catch (e) {

console.error('Error opening tab:', e);

errorMessage.style.display = 'block';

errorMessage.textContent = 'Error: ' + e.message;

openTabsButton.textContent = 'Start Increasing Views';

openTabsButton.disabled = false;

return;

}

// Schedule next tab opening with the specified delay

setTimeout(openTabWithDelay, speed);

} else {

// Process completed

successMessage.textContent = 'Process completed! ' + viewsAttempted + ' views attempted.';

openTabsButton.textContent = 'Start Increasing Views';

openTabsButton.disabled = false;

}

}

// Start opening tabs

openTabWithDelay();

});

});

</script>

</body>

</html>