<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Professional Wellbeing Radar Survey</title>

<script src="https://cdn.jsdelivr.net/npm/chart.js"></script>

<style>

body { font-family: Arial, sans-serif; padding: 20px; }

.slider-container { margin: 10px 0; }

label { font-weight: bold; }

</style>

</head>

<body>

<h2>Professional Wellbeing Radar Survey</h2>

<div id="sliders"></div>

<button onclick="generateChart()">Generate Radar Chart</button>

<canvas id="radarChart" width="400" height="400"></canvas>

<script>

const dimensions = [

"Perspective",

"Self-management",

"Supports",

"Meaningfulness",

"Self-care",

"Practice competency",

"Professional development"

];

const slidersDiv = document.getElementById("sliders");

// Create sliders for each dimension

dimensions.forEach((dimension, index) => {

const container = document.createElement("div");

container.className = "slider-container";

container.innerHTML = `

<label for="slider${index}">${dimension}</label><br>

<input type="range" id="slider${index}" min="1" max="5" value="3" step="1">

<span id="value${index}">3</span>

`;

slidersDiv.appendChild(container);

// Update value display when slider is moved

const slider = container.querySelector(`#slider${index}`);

const valueDisplay = container.querySelector(`#value${index}`);

slider.oninput = () => {

valueDisplay.textContent = slider.value;

};

});

// Function to generate radar chart

function generateChart() {

const values = dimensions.map((\_, i) => parseInt(document.getElementById(`slider${i}`).value));

const ctx = document.getElementById('radarChart').getContext('2d');

if (window.radarChartInstance) {

window.radarChartInstance.destroy();

}

window.radarChartInstance = new Chart(ctx, {

type: 'radar',

data: {

labels: dimensions,

datasets: [{

label: 'Your Wellbeing Map',

data: values,

backgroundColor: 'rgba(153, 102, 255, 0.2)',

borderColor: 'rgba(153, 102, 255, 1)',

pointBackgroundColor: 'rgba(153, 102, 255, 1)',

borderWidth: 2

}]

},

options: {

scale: {

ticks: {

beginAtZero: true,

min: 0,

max: 5,

stepSize: 1

}

}

}

});

}

</script>

</body>

</html>