<!DOCTYPE html>

<html lang="es">

<head>

<meta charset="UTF-8">

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

<title>Panel de Control de Trekking</title>

<script src="https://cdn.tailwindcss.com"></script>

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

<style>

@import url('https://fonts.googleapis.com/css2?family=Inter:wght@400;500;600;700;800&display=swap');

body {

font-family: 'Inter', sans-serif;

background-color: #f8f9fa;

}

.modal {

transition: opacity 0.3s ease;

}

.modal-content {

transition: transform 0.3s ease;

}

.chart-container {

position: relative;

margin: auto;

height: 280px;

width: 100%;

max-width: 450px;

}

@media (min-width: 768px) {

.chart-container {

height: 320px;

}

}

::-webkit-scrollbar {

width: 8px;

}

::-webkit-scrollbar-track {

background: #f1f1f1;

}

::-webkit-scrollbar-thumb {

background: #888;

border-radius: 10px;

}

::-webkit-scrollbar-thumb:hover {

background: #555;

}

</style>

<!-- Chosen Palette: Warm Neutrals (Stone, Muted Green/Blue) -->

<!-- Application Structure Plan: A single-page dashboard design. The main view displays a grid of trek summary cards and key statistics/charts. A prominent 'Add Trek' button opens a modal form for creating/editing treks. Each card has buttons to view full details (in another modal) or delete the entry. This structure is chosen for its efficiency, keeping all CRUD operations accessible from a single screen without page reloads, which is ideal for a management-style application. -->

<!-- Visualization & Content Choices:

- Trek List: Goal: Read summary. Method: HTML/CSS cards. Interaction: Click buttons for details/edit/delete. Justification: Scannable and standard UI for lists.

- Trek Form: Goal: Create/Update. Method: Grouped HTML form fields in a modal. Interaction: Standard form input. Justification: Clear data entry workflow.

- Statistics (Total Treks, Avg. Distance): Goal: Inform (quick stats). Method: Dynamic text in HTML. Interaction: Updates automatically. Justification: High-level overview.

- Treks by Difficulty Chart: Goal: Compare proportions. Method: Chart.js Doughnut Chart on Canvas. Interaction: Hover tooltips. Justification: Easy visual breakdown of activity types. NO SVG used.

- Treks per Month Chart: Goal: Show change over time. Method: Chart.js Bar Chart on Canvas. Interaction: Hover tooltips. Justification: Clear representation of trekking frequency. NO SVG/Mermaid used. -->

<!-- CONFIRMATION: NO SVG graphics used. NO Mermaid JS used. -->

</head>

<body class="bg-stone-50 text-stone-800">

<div id="app" class="max-w-7xl mx-auto p-4 sm:p-6 lg:p-8">

<header class="flex flex-col sm:flex-row justify-between items-center mb-8 pb-4 border-b border-stone-200">

<div>

<h1 class="text-3xl md:text-4xl font-extrabold text-stone-900">Panel de Trekking</h1>

<p class="mt-1 text-stone-500">Organiza y analiza tus aventuras en la montaña.</p>

</div>

<button id="add-trek-btn" class="w-full sm:w-auto mt-4 sm:mt-0 flex items-center justify-center gap-2 px-5 py-3 bg-green-600 text-white font-semibold rounded-lg shadow-md hover:bg-green-700 focus:outline-none focus:ring-2 focus:ring-green-500 focus:ring-opacity-75 transition-all duration-300 transform hover:scale-105">

<span class="text-2xl leading-none">＋</span>

<span>Agregar Salida</span>

</button>

</header>

<main>

<section id="dashboard" class="mb-10">

<h2 class="text-2xl font-bold mb-6 text-stone-700">Tu Resumen de Actividad</h2>

<div class="grid grid-cols-1 md:grid-cols-3 gap-6 mb-6">

<div class="bg-white p-6 rounded-xl shadow-sm border border-stone-200/50 flex flex-col justify-center items-center text-center">

<h3 class="text-stone-500 font-semibold">Salidas Totales</h3>

<p id="total-treks" class="text-4xl font-extrabold text-green-600 mt-2">0</p>

</div>

<div class="bg-white p-6 rounded-xl shadow-sm border border-stone-200/50 flex flex-col justify-center items-center text-center">

<h3 class="text-stone-500 font-semibold">Distancia Total</h3>

<p class="text-4xl font-extrabold text-green-600 mt-2"><span id="total-distance">0</span> km</p>

</div>

<div class="bg-white p-6 rounded-xl shadow-sm border border-stone-200/50 flex flex-col justify-center items-center text-center">

<h3 class="text-stone-500 font-semibold">Promedio Personas</h3>

<p id="avg-people" class="text-4xl font-extrabold text-green-600 mt-2">0</p>

</div>

</div>

<div class="grid grid-cols-1 lg:grid-cols-2 gap-6">

<div class="bg-white p-6 rounded-xl shadow-sm border border-stone-200/50">

<h3 class="text-lg font-bold text-stone-700 text-center mb-4">Salidas por Dificultad</h3>

<div class="chart-container">

<canvas id="difficulty-chart"></canvas>

</div>

</div>

<div class="bg-white p-6 rounded-xl shadow-sm border border-stone-200/50">

<h3 class="text-lg font-bold text-stone-700 text-center mb-4">Salidas por Mes (Último Año)</h3>

<div class="chart-container">

<canvas id="monthly-chart"></canvas>

</div>

</div>

</div>

</section>

<section id="trek-list-section">

<h2 class="text-2xl font-bold mb-6 text-stone-700">Tus Salidas Registradas</h2>

<div id="trek-list-container" class="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-6">

</div>

</section>

</main>

</div>

<div id="form-modal" class="modal fixed inset-0 bg-black bg-opacity-60 flex justify-center items-center p-4 z-50 opacity-0 pointer-events-none">

<div class="modal-content bg-white w-full max-w-2xl rounded-xl shadow-2xl max-h-[90vh] overflow-y-auto transform scale-95">

<form id="trek-form" class="p-8">

<h2 id="form-title" class="text-2xl font-bold text-stone-800 mb-6"></h2>

<input type="hidden" id="trek-id">

<fieldset class="mb-6">

<legend class="text-lg font-semibold text-green-700 mb-2">🏞️ Datos del Lugar</legend>

<div class="grid grid-cols-1 sm:grid-cols-2 gap-4">

<div>

<label for="placeName" class="block text-sm font-medium text-stone-600">Nombre del lugar</label>

<input type="text" id="placeName" required class="mt-1 block w-full border-stone-300 rounded-md shadow-sm focus:border-green-500 focus:ring-green-500">

</div>

<div>

<label for="trekDifficulty" class="block text-sm font-medium text-stone-600">Dificultad</label>

<select id="trekDifficulty" required class="mt-1 block w-full border-stone-300 rounded-md shadow-sm focus:border-green-500 focus:ring-green-500">

<option>Fácil</option>

<option>Moderada</option>

<option>Difícil</option>

<option>Extrema</option>

</select>

</div>

<div>

<label for="distanceKm" class="block text-sm font-medium text-stone-600">Distancia (km)</label>

<input type="number" id="distanceKm" min="0" step="0.1" required class="mt-1 block w-full border-stone-300 rounded-md shadow-sm focus:border-green-500 focus:ring-green-500">

</div>

</div>

</fieldset>

<fieldset class="mb-6">

<legend class="text-lg font-semibold text-green-700 mb-2">🗓️ Horarios</legend>

<div class="grid grid-cols-1 sm:grid-cols-3 gap-4">

<div>

<label for="date" class="block text-sm font-medium text-stone-600">Fecha</label>

<input type="date" id="date" required class="mt-1 block w-full border-stone-300 rounded-md shadow-sm focus:border-green-500 focus:ring-green-500">

</div>

<div>

<label for="startTime" class="block text-sm font-medium text-stone-600">Hora de inicio</label>

<input type="time" id="startTime" class="mt-1 block w-full border-stone-300 rounded-md shadow-sm focus:border-green-500 focus:ring-green-500">

</div>

<div>

<label for="endTime" class="block text-sm font-medium text-stone-600">Hora de fin</label>

<input type="time" id="endTime" class="mt-1 block w-full border-stone-300 rounded-md shadow-sm focus:border-green-500 focus:ring-green-500">

</div>

</div>

</fieldset>

<fieldset class="mb-6">

<legend class="text-lg font-semibold text-green-700 mb-2">🚌 Traslados y Personas</legend>

<div class="grid grid-cols-1 sm:grid-cols-2 gap-4">

<div>

<label for="transportType" class="block text-sm font-medium text-stone-600">Tipo de traslado</label>

<input type="text" id="transportType" placeholder="Ej: Auto, Bus, Contratado" class="mt-1 block w-full border-stone-300 rounded-md shadow-sm focus:border-green-500 focus:ring-green-500">

</div>

<div>

<label for="numPeople" class="block text-sm font-medium text-stone-600">Cantidad de personas</label>

<input type="number" id="numPeople" min="1" required class="mt-1 block w-full border-stone-300 rounded-md shadow-sm focus:border-green-500 focus:ring-green-500">

</div>

</div>

</fieldset>

<fieldset class="mb-6">

<legend class="text-lg font-semibold text-green-700 mb-2">☀️ Clima y Registro</legend>

<div class="grid grid-cols-1 sm:grid-cols-2 gap-4">

<div>

<label for="weather" class="block text-sm font-medium text-stone-600">Clima</label>

<input type="text" id="weather" placeholder="Ej: Soleado, 15°C" class="mt-1 block w-full border-stone-300 rounded-md shadow-sm focus:border-green-500 focus:ring-green-500">

</div>

</div>

<div class="mt-4">

<label for="trekkingLog" class="block text-sm font-medium text-stone-600">Registro de la salida</label>

<textarea id="trekkingLog" rows="4" placeholder="Describe observaciones, incidentes o puntos de interés..." class="mt-1 block w-full border-stone-300 rounded-md shadow-sm focus:border-green-500 focus:ring-green-500"></textarea>

</div>

</fieldset>

<div class="flex justify-end gap-4 mt-8">

<button type="button" id="cancel-btn" class="px-6 py-2 bg-stone-200 text-stone-800 font-semibold rounded-lg hover:bg-stone-300 transition-colors">Cancelar</button>

<button type="submit" class="px-6 py-2 bg-green-600 text-white font-semibold rounded-lg hover:bg-green-700 transition-colors">Guardar Salida</button>

</div>

</form>

</div>

</div>

<div id="details-modal" class="modal fixed inset-0 bg-black bg-opacity-60 flex justify-center items-center p-4 z-50 opacity-0 pointer-events-none">

<div class="modal-content bg-white w-full max-w-2xl rounded-xl shadow-2xl max-h-[90vh] overflow-y-auto transform scale-95">

<div id="details-content" class="p-8"></div>

<div class="p-8 pt-0 flex justify-end">

<button id="close-details-btn" class="px-6 py-2 bg-stone-200 text-stone-800 font-semibold rounded-lg hover:bg-stone-300 transition-colors">Cerrar</button>

</div>

</div>

</div>

<script>

document.addEventListener('DOMContentLoaded', () => {

const treks = [

{ id: 1, placeName: 'Cerro Llao Llao', trekDifficulty: 'Fácil', distanceKm: 6, date: '2025-01-15', startTime: '09:00', endTime: '13:00', transportType: 'Bus Línea 20', numPeople: 4, weather: 'Soleado, 22°C', trekkingLog: 'Vistas espectaculares del lago Nahuel Huapi. Sendero bien marcado y de baja dificultad. Ideal para familias.' },

{ id: 2, placeName: 'Refugio Frey', trekDifficulty: 'Moderada', distanceKm: 20, date: '2025-02-20', startTime: '08:00', endTime: '17:00', transportType: 'Auto particular', numPeople: 2, weather: 'Parcialmente nublado, 18°C', trekkingLog: 'El camino de ida es largo pero gratificante. La llegada al refugio con la laguna y las agujas de granito es impresionante.' },

{ id: 3, placeName: 'Valle del Challhuaco', trekDifficulty: 'Fácil', distanceKm: 8, date: '2025-06-10', startTime: '10:00', endTime: '15:00', transportType: 'Excursión contratada', numPeople: 8, weather: 'Nieve ligera, -2°C', trekkingLog: 'Trekking con raquetas de nieve. El bosque nevado es mágico. Se vieron carpinteros gigantes.' },

{ id: 4, placeName: 'Cerro Catedral (Filo)', trekDifficulty: 'Difícil', distanceKm: 12, date: '2024-12-05', startTime: '07:30', endTime: '16:00', transportType: 'Auto particular', numPeople: 3, weather: 'Viento fuerte, 10°C', trekkingLog: 'Subida exigente con tramos de piedra suelta. Las vistas 360° desde el filo justifican el esfuerzo. Requiere buena condición física.'}

];

let difficultyChart = null;

let monthlyChart = null;

const trekListContainer = document.getElementById('trek-list-container');

const addTrekBtn = document.getElementById('add-trek-btn');

const formModal = document.getElementById('form-modal');

const form = document.getElementById('trek-form');

const cancelBtn = document.getElementById('cancel-btn');

const detailsModal = document.getElementById('details-modal');

const detailsContent = document.getElementById('details-content');

const closeDetailsBtn = document.getElementById('close-details-btn');

const difficultyColors = {

'Fácil': 'rgba(74, 222, 128, 0.8)',

'Moderada': 'rgba(250, 204, 21, 0.8)',

'Difícil': 'rgba(239, 68, 68, 0.8)',

'Extrema': 'rgba(124, 58, 237, 0.8)'

};

const difficultyBorderColors = {

'Fácil': 'rgba(34, 197, 94, 1)',

'Moderada': 'rgba(234, 179, 8, 1)',

'Difícil': 'rgba(220, 38, 38, 1)',

'Extrema': 'rgba(109, 40, 217, 1)'

};

function renderTrekList() {

treks.sort((a, b) => new Date(b.date) - new Date(a.date));

trekListContainer.innerHTML = '';

if(treks.length === 0) {

trekListContainer.innerHTML = `<p class="text-stone-500 col-span-full text-center">No hay salidas registradas. ¡Agrega la primera!</p>`;

return;

}

treks.forEach(trek => {

const card = document.createElement('div');

card.className = 'bg-white rounded-xl shadow-sm border border-stone-200/50 p-6 flex flex-col justify-between hover:shadow-lg hover:-translate-y-1 transition-all duration-300';

card.innerHTML = `

<div>

<div class="flex justify-between items-start mb-2">

<h3 class="text-xl font-bold text-stone-800">${trek.placeName}</h3>

<span class="text-sm font-semibold px-3 py-1 rounded-full" style="background-color: ${difficultyColors[trek.trekDifficulty]}; color: #fff;">${trek.trekDifficulty}</span>

</div>

<p class="text-stone-500 text-sm mb-4">🗓️ ${new Date(trek.date).toLocaleDateString('es-AR', { year: 'numeric', month: 'long', day: 'numeric' })}</p>

<div class="flex items-center text-stone-600 mb-2">

<span class="text-xl mr-2">↔️</span>

<span>${trek.distanceKm} km</span>

</div>

<div class="flex items-center text-stone-600">

<span class="text-xl mr-2">👨‍👩‍👧‍👦</span>

<span>${trek.numPeople} personas</span>

</div>

</div>

<div class="flex justify-end gap-2 mt-6">

<button data-id="${trek.id}" class="details-btn text-sm px-4 py-2 bg-stone-100 text-stone-700 font-semibold rounded-lg hover:bg-stone-200 transition-colors">Detalles</button>

<button data-id="${trek.id}" class="edit-btn text-sm px-4 py-2 bg-blue-100 text-blue-700 font-semibold rounded-lg hover:bg-blue-200 transition-colors">Editar</button>

<button data-id="${trek.id}" class="delete-btn text-sm px-4 py-2 bg-red-100 text-red-700 font-semibold rounded-lg hover:bg-red-200 transition-colors">Borrar</button>

</div>

`;

trekListContainer.appendChild(card);

});

}

function updateDashboard() {

const totalTreks = treks.length;

const totalDistance = treks.reduce((sum, trek) => sum + (Number(trek.distanceKm) || 0), 0);

const totalPeople = treks.reduce((sum, trek) => sum + (Number(trek.numPeople) || 0), 0);

const avgPeople = totalTreks > 0 ? (totalPeople / totalTreks).toFixed(1) : 0;

document.getElementById('total-treks').textContent = totalTreks;

document.getElementById('total-distance').textContent = totalDistance.toFixed(1);

document.getElementById('avg-people').textContent = avgPeople;

updateDifficultyChart();

updateMonthlyChart();

}

function updateDifficultyChart() {

const ctx = document.getElementById('difficulty-chart').getContext('2d');

const difficultyCounts = treks.reduce((acc, trek) => {

acc[trek.trekDifficulty] = (acc[trek.trekDifficulty] || 0) + 1;

return acc;

}, {});

const labels = Object.keys(difficultyCounts);

const data = Object.values(difficultyCounts);

if(difficultyChart) {

difficultyChart.destroy();

}

difficultyChart = new Chart(ctx, {

type: 'doughnut',

data: {

labels: labels,

datasets: [{

label: 'Salidas por Dificultad',

data: data,

backgroundColor: labels.map(label => difficultyColors[label]),

borderColor: labels.map(label => difficultyBorderColors[label]),

borderWidth: 2

}]

},

options: {

responsive: true,

maintainAspectRatio: false,

plugins: {

legend: {

position: 'bottom',

},

tooltip: {

callbacks: {

label: function(context) {

let label = context.label || '';

if (label) {

label += ': ';

}

if (context.parsed !== null) {

label += context.parsed + ' salidas';

}

return label;

}

}

}

}

}

});

}

function updateMonthlyChart() {

const ctx = document.getElementById('monthly-chart').getContext('2d');

const twelveMonthsAgo = new Date();

twelveMonthsAgo.setMonth(twelveMonthsAgo.getMonth() - 11);

twelveMonthsAgo.setDate(1);

const monthlyCounts = Array(12).fill(0);

const monthLabels = Array(12).fill(0).map((\_, i) => {

const d = new Date(twelveMonthsAgo);

d.setMonth(d.getMonth() + i);

return d.toLocaleString('es-AR', { month: 'short', year: '2-digit' });

});

treks.forEach(trek => {

const trekDate = new Date(trek.date);

if (trekDate >= twelveMonthsAgo) {

const monthDiff = (trekDate.getFullYear() - twelveMonthsAgo.getFullYear()) \* 12 + (trekDate.getMonth() - twelveMonthsAgo.getMonth());

if(monthDiff >= 0 && monthDiff < 12) {

monthlyCounts[monthDiff]++;

}

}

});

if(monthlyChart) {

monthlyChart.destroy();

}

monthlyChart = new Chart(ctx, {

type: 'bar',

data: {

labels: monthLabels,

datasets: [{

label: 'Nº de Salidas',

data: monthlyCounts,

backgroundColor: 'rgba(34, 197, 94, 0.6)',

borderColor: 'rgba(22, 163, 74, 1)',

borderWidth: 1,

borderRadius: 5,

}]

},

options: {

responsive: true,

maintainAspectRatio: false,

scales: {

y: {

beginAtZero: true,

ticks: {

stepSize: 1

}

}

},

plugins: {

legend: {

display: false

}

}

}

});

}

function showModal(modalElement) {

modalElement.classList.remove('opacity-0', 'pointer-events-none');

modalElement.querySelector('.modal-content').classList.remove('scale-95');

}

function hideModal(modalElement) {

modalElement.classList.add('opacity-0');

modalElement.querySelector('.modal-content').classList.add('scale-95');

setTimeout(() => {

modalElement.classList.add('pointer-events-none');

}, 300);

}

function showFormModal(trek = null) {

form.reset();

if(trek) {

document.getElementById('form-title').textContent = 'Editar Salida';

document.getElementById('trek-id').value = trek.id;

document.getElementById('placeName').value = trek.placeName;

document.getElementById('trekDifficulty').value = trek.trekDifficulty;

document.getElementById('distanceKm').value = trek.distanceKm;

document.getElementById('date').value = trek.date;

document.getElementById('startTime').value = trek.startTime;

document.getElementById('endTime').value = trek.endTime;

document.getElementById('transportType').value = trek.transportType;

document.getElementById('numPeople').value = trek.numPeople;

document.getElementById('weather').value = trek.weather;

document.getElementById('trekkingLog').value = trek.trekkingLog;

} else {

document.getElementById('form-title').textContent = 'Agregar Nueva Salida';

document.getElementById('trek-id').value = '';

}

showModal(formModal);

}

function showDetailsModal(trek) {

detailsContent.innerHTML = `

<h2 class="text-3xl font-bold text-stone-800 mb-4">${trek.placeName}</h2>

<div class="grid grid-cols-1 md:grid-cols-2 gap-x-8 gap-y-4 text-stone-700">

<p><strong>🗓️ Fecha:</strong> ${new Date(trek.date).toLocaleDateString('es-AR', { year: 'numeric', month: 'long', day: 'numeric' })}</p>

<p><strong>🕒 Horario:</strong> ${trek.startTime || 'N/A'} - ${trek.endTime || 'N/A'}</p>

<p><strong>⛰️ Dificultad:</strong> ${trek.trekDifficulty}</p>

<p><strong>↔️ Distancia:</strong> ${trek.distanceKm} km</p>

<p><strong>👨‍👩‍👧‍👦 Personas:</strong> ${trek.numPeople}</p>

<p><strong>🚌 Traslado:</strong> ${trek.transportType || 'No especificado'}</p>

<p><strong>☀️ Clima:</strong> ${trek.weather || 'No especificado'}</p>

</div>

<div class="mt-6">

<h4 class="text-lg font-semibold text-green-700 mb-2">📝 Registro de la Salida</h4>

<p class="text-stone-600 bg-stone-100 p-4 rounded-lg whitespace-pre-wrap">${trek.trekkingLog || 'No hay notas para esta salida.'}</p>

</div>

`;

showModal(detailsModal);

}

addTrekBtn.addEventListener('click', () => showFormModal());

cancelBtn.addEventListener('click', () => hideModal(formModal));

closeDetailsBtn.addEventListener('click', () => hideModal(detailsModal));

form.addEventListener('submit', (e) => {

e.preventDefault();

const id = document.getElementById('trek-id').value;

const newTrek = {

placeName: document.getElementById('placeName').value,

trekDifficulty: document.getElementById('trekDifficulty').value,

distanceKm: parseFloat(document.getElementById('distanceKm').value),

date: document.getElementById('date').value,

startTime: document.getElementById('startTime').value,

endTime: document.getElementById('endTime').value,

transportType: document.getElementById('transportType').value,

numPeople: parseInt(document.getElementById('numPeople').value),

weather: document.getElementById('weather').value,

trekkingLog: document.getElementById('trekkingLog').value,

};

if(id) {

const index = treks.findIndex(t => t.id == id);

treks[index] = { ...treks[index], ...newTrek };

} else {

newTrek.id = Date.now();

treks.push(newTrek);

}

hideModal(formModal);

renderTrekList();

updateDashboard();

});

trekListContainer.addEventListener('click', (e) => {

if(e.target.closest('.edit-btn')) {

const id = e.target.closest('.edit-btn').dataset.id;

const trek = treks.find(t => t.id == id);

showFormModal(trek);

}

if(e.target.closest('.delete-btn')) {

const id = e.target.closest('.delete-btn').dataset.id;

if(confirm('¿Estás seguro de que quieres eliminar esta salida?')) {

const index = treks.findIndex(t => t.id == id);

treks.splice(index, 1);

renderTrekList();

updateDashboard();

}

}

if(e.target.closest('.details-btn')) {

const id = e.target.closest('.details-btn').dataset.id;

const trek = treks.find(t => t.id == id);

showDetailsModal(trek);

}

});

renderTrekList();

updateDashboard();

});

</script>

</body>

</html>