# System Design - UV Exposure Tracker

Project: UV Exposure & Weather Tracker

Client: South Skin Cancer Treatment Center of America

Date: 2025-03-28

## Interface Components

- Input field for city or ZIP  
- UV index chart (Chart.js)  
- Daily forecast list with UV classification  
- Progress bar with color indicator  
- Warning message for overexposure  
- Educational notes and tooltips

## Logic & Flow

1. User enters location or allows GPS.  
2. APIs fetch UV and weather data.  
3. UV chart and forecast are rendered.  
4. Progress bar shows total UV exposure.  
5. Alert appears if user exceeds safe limits.

## Code Implementation

The following code is a snapshot of the core implementation:

<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>South Skin Cancer Treatment Center of America | UV Exposure & Weather Tracker</title>  
 <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>  
 <style>  
 body {  
 font-family: Arial, sans-serif;  
 margin: 20px;  
 }  
 #weather {  
 margin-top: 30px;  
 }  
 label, input, button {  
 margin: 5px 0;  
 }  
 .low { color: green; }  
 .moderate { color: orange; }  
 .high { color: red; }  
 .very-high { color: purple; }  
 .extreme { color: maroon; }  
 </style>  
</head>  
<body>  
 <h1>South Skin Cancer Treatment Center of America</h1>  
 <p><strong>Headquarters:</strong> Dallas, TX</p>  
 <h2>UV Exposure Tracker (Available to All Users)</h2>  
 <p>This tool is provided by the South Skin Cancer Treatment Center of America and is freely available to the public. No login is required. Just enter a city or U.S. ZIP code to see the forecast!</p>  
  
 <canvas id="uvChart" width="600" height="400"></canvas>  
  
 <h2 id="weather">Weather Forecast</h2>  
 <label for="city">Enter City or ZIP Code:</label>  
 <input type="text" id="city" placeholder="e.g. London or 90210">  
 <button onclick="getWeather()">Check Weather</button>  
 <p id="weatherResult"></p>  
 <ul id="forecastList"></ul>  
  
 <p style="font-size: 0.85em; color: #444; margin-top: 5px;">  
 <em>\*Approximate UV absorbed is based on UV Index × 15 minutes.   
 <span title='This estimate assumes average skin exposure and sunlight conditions. Always consult your doctor for personalized advice.' style='cursor: help; text-decoration: dotted underline;'>What's this?</span> This gives a rough estimate of the time your skin is exposed to UV based on intensity.   
 <span title='Your UV exposure depends on factors like skin tone, altitude, time spent outdoors, and use of protection like sunscreen or clothing. Consider speaking with a healthcare provider for personal UV safety recommendations.' style='cursor: help; text-decoration: dotted underline;'>More info</span></em>  
 </p>  
  
 <button onclick="window.print()" style="margin-top: 15px;">Print Forecast</button>  
  
 <script>  
 // Author: [Shequila Sledge]  
 // Description: This custom UV and weather tracker was built for the South Skin Cancer Treatment Center of America.  
 // It uses Open-Meteo APIs to fetch weather forecasts and render a UV chart using Chart.js.  
  
 const uvData = {  
 labels: ['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday'],  
 datasets: [{  
 label: 'UV Index Level',  
 data: [3, 5, 6, 4, 7, 8, 5],  
 borderColor: 'rgba(255, 99, 132, 1)',  
 backgroundColor: 'rgba(255, 99, 132, 0.2)',  
 fill: true,  
 tension: 0.3  
 }]  
 };  
  
 const config = {  
 type: 'line',  
 data: uvData,  
 options: {  
 scales: {  
 y: {  
 title: {  
 display: true,  
 text: 'UV Index'  
 },  
 beginAtZero: true,  
 suggestedMax: 10  
 },  
 x: {  
 title: {  
 display: true,  
 text: 'Day of the Week'  
 }  
 }  
 },  
 plugins: {  
 title: {  
 display: true,  
 text: 'South Skin Cancer Treatment Center - Weekly UV Exposure Levels'  
 },  
 tooltip: {  
 callbacks: {  
 label: function(context) {  
 return `UV Index: ${context.parsed.y}`;  
 }  
 }  
 }  
 }  
 }  
 };  
  
 new Chart(document.getElementById('uvChart'), config);  
  
 async function getWeather() {  
 const city = document.getElementById('city').value;  
 try {  
 const geoResponse = await fetch(`https://geocoding-api.open-meteo.com/v1/search?name=${encodeURIComponent(city)}`);  
 const geoData = await geoResponse.json();  
 const location = geoData.results[0];  
 const weatherUrl = `https://api.open-meteo.com/v1/forecast?latitude=${location.latitude}&longitude=${location.longitude}&daily=weathercode,uv\_index\_max&timezone=auto`;  
 const response = await fetch(weatherUrl);  
 const data = await response.json();  
  
 const todayCode = data.daily.weathercode[0];  
 const todayUV = data.daily.uv\_index\_max[0];  
 const description = interpretWeatherCode(todayCode);  
 const risk = interpretUVRisk(todayUV);  
 document.getElementById('weatherResult').innerHTML = `Today in <strong>${city}</strong>: ${description}, Max UV Index: <span class="${risk.class}">${todayUV}

## Future Improvements

- Add multilingual support  
- Introduce dark mode  
- Alert scheduling or notifications  
- Admin dashboard for usage stats