FINAL PROJECT REPORT

UV INDEX CHART TOOL FOR PATIENTS

SOUTHERN SKIN CANCER TREATMENT CENTERS OF AMERICA
123 MAIN STREET
DALLAS, TX 00000
PROJECT MANAGER: MELISSA ECKER

DECEMBER 1, 2025

TABLE OF CONTENTS

CLIENT INTRODUCTION/ORGANIZATION BACKGROUND

Southern Skin Cancer Treatment Centers of America (SSCTCoA) is a dedicated group of clinics that provide medical services for patients with or at high risk of developing skin cancer. This organization prioritizes preventative care and patient education. A patient portal is provided where clients can access their individualized treatment plans and log their daily UV exposure data.

SSCTCoA is currently undergoing a significant website redesign to accommodate their growing clientele and employee base. This project was initiated to integrate a crucial tool—the UV Index Chart Tool—into the new website design, thereby enhancing the value of the patient portal and supporting the organization's commitment to patient health. The Southern Skin Cancer Treatment Center of America is a privately-owned corporation that is governed by a board of directors.

The overall budget for this project was $20,000, and the target completion date was December 1, 2025.

ANALYSIS

The Problem

The primary challenge facing SSCTCoA and its patients was the lack of an integrated, easy-to-use tool for accurately tracking and predicting UV exposure. Patients are instructed to manually log their exposure in the patient portal without a quick reference tool to access localized UV index data. The lack of such a tool placed an undue burden on patients to find the requisite data from external sources which leaves the potential for inconsistent tracking and possible non-compliance with their respective treatment plans.

Patients of Southern Skin Cancer Treatment Centers of America currently lack an integrated, user-friendly tool within the patient portal to quickly view historical and forecasted local UV index data, which hinders their ability to accurately track and manage their sun exposure relative to their individualized treatment plans, thereby potentially impacting patient outcomes.

Current State

- **Data:** Patient UV exposure is tracked and input manually into the patient portal. UV index data is sourced externally and is inconsistent as patients search for it independently online.

- **Process:** Patients manually seek out local UV index data via third-party apps or websites then attempt to estimate their safe exposure levels. They then input that information into the patient portal. This process is not efficient and is susceptible to errors.

- **People:** Patients are solely responsible for finding the relevant data and inputting it correctly. Medical staff at SSCTCoA then rely upon that data to recommend treatment.

- **Technology:** The organization utilizes an existing patient portal and website which is undergoing a major redesign. The current platform does not include an integrated UV index tracking tool.

Desired Outcome

The desired future state involves a seamless, integrated digital solution that automates the data retrieval process and provides immediate UV index information within the patient portal.

- **Data:** Structured UV index data is retrieved directly from a reliable API (openmeteo.com) and presented in an easy-to-read chart. The system uses city or zip code input by the patient to source data specific to location.

- **Process:** Patients access their patient portal on SSCTCoA website, input their location via city or zip code and will immediately be delivered a chart showing a 10-day view (five days before and five days forward) from the date of input. This simplifies their obligation to track their UV exposure and aid in making health decisions.

- **People:** Being able to check for UV index data at any time and any place, will empower patients to make better decisions and be more involved in their care and treatment. Medical staff will receive reliable data from the patients that shows their compliance with the treatment plan.

- **Technical Improvements:** A responsive client-side web application built with HTML5, CSS, and JavaScript that is integrated into the new website's patient portal. The application adheres to current IT platform and technology standards.

Options and Solutions

The primary goal was to integrate a new UV index tracking tool into the patient portal of the newly designed website. Three main options were considered for reaching that desired outcome.

The first option was to buy a pre-built commercial UV tracking widget to embed into the new website's patient portal. This would have been the quickest option. However, we considered the potential high cost of licensing and/or subscription maintenance and decided that a custom build would be more economical. We also took into account the limitations on customization when using a product built for generic use. Finally, we did not want to risk compatibility issues with the new website. It made more sense to design an application from scratch that we knew would work.

Second, we discussed the option of improving the existing manual entry method in the patient portal. This would take some time but would save money on design and development costs. This option does not address the main issue of patients shouldering the responsibility to find accurate UV index information to input. It also does not address the problems surrounding patient care being planned by that input if it is inaccurate and/or inconsistent.

The third option was to build the application from scratch using an open-source API. This involved developing a client-side tool using standard programming languages (HTML5, CSS, Javascript) and integrating it with a reliable weather API like openmeteo.com. This allows control over the design and customization and lower costs to maintain and operate the application. Compatibility is guaranteed. The downside is that it will take longer to design and develop and will have higher upfront costs.

The team recommended option three to build the application from scratch.

Project Plan

The project was executed following the phased approach detailed in the Project Charter as follows: The project was executed following a phased approach as documented in the Project Charter.

Project Milestone Schedule	**Target Date**	**Actual Date**	**Status**
Project Start	10/10/2025	10/10/2025	Complete
Complete Solution Design	10/25/2025	10/23/2025	Complete
Acquire Hardware and Software	10/30/2025	10/28/2025	Complete
Complete Solution Simulation with New Hardware/Software	11/05/2025	11/04/2025	Complete
Complete Solution Simulation and Testing	11/15/2025	11/15/2025	Complete
Deploy Solution	**11/25/2025**	**11/25/2025**	**Complete**
Project Complete	**12/01/2025**	**12/01/2025**	**Complete**

- **Budget/Resource Constraints:** The project was constrained by a total effort budget of 110 hours of labor time, which was successfully met. Hardware and software were constrained to be compatible with current organization IT platforms and required no new costs, adhering to the budget.

- **Skills/Training Needs:** The project knowledge of frontend web development (HTML, CSS, JavaScript) and experience with API integration and data visualization (Chart.js). These skills were managed internally by the Project Manager and the IT support team.

Design & Development Work Performed

Project Design

The solution utilizes a client-side web application architecture which has minimal impact on the organization's backend infrastructure.

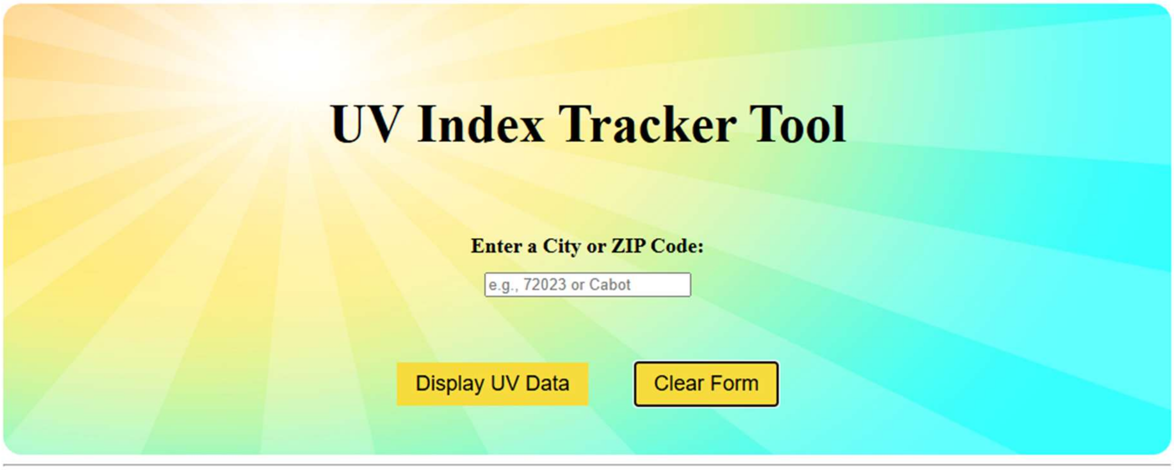
- **IT Infrastructure:** The solution is designed to run independently on the client's web browser, hosted on the SSCTCoA website's existing web server.
    - **Hardware:** Standard end-user devices (computers, laptops, tablets, and mobile devices) were used. No new server hardware was needed or purchased.
    - **Software (Development):** Github (repository and hosting service for development), Notepad (code editor), and Git (version control).
    - **Software (Runtime):** Modern web browsers supporting HTML5, CSS, and JavaScript.
    - **Environments:** Development/testing was conducted on a separate, independent webpage before final deployment to the patient portal on the new website.

Project Development

Data Architecture: The UV index tool does not store patient information or records in compliance with non-functional and ethical considerations. Data is handled temporarily on the client side:

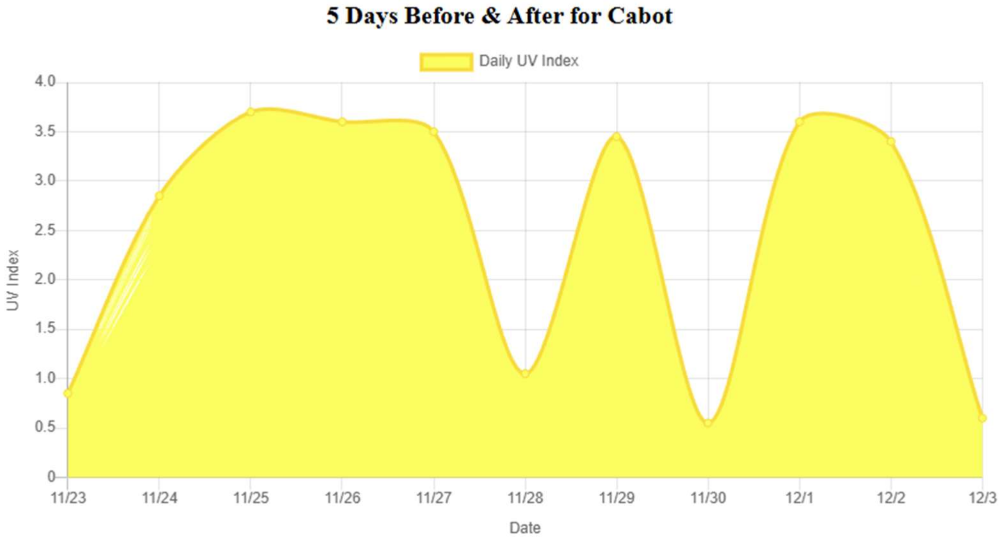
- **Input Data:** City name or zip code.
- **External Data:** UV index data retrieved from openmeteo.com API which returns structured data including dates and UV index values.
- **Application's Interface/Interactions:** The user interface was designed for simplicity and ease of use. It is intuitive and functions responsively on various web browsers.

    Interface Elements:
    - Input box for the user to enter a city or zip code.
    - "Find UV Index" button to initiate the API call.
    - "Clear Form" button to start over or check other locations.
    - An easy-to-read line chart via Chart.js to display 11 days of UV data (five days before and five days after current date).



UV Index Tracker Tool

Enter a City or ZIP Code:



5 Days Before & After for Cabot

- **Code/Application Development:** The solution was developed using a standard web development stack:
    - **HTML5:** Structure and organization of the user interface.
    - **CSS:** Styling to ensure the tool integrates seamlessly with the newly designed website's appearance and functionality.
    - **JavaScript:** Core logic for:
        - Capturing user input of city or zip code.

        - Making the API request to openmeteo.com to UV data including five days prior and five days in the future.
        - Processing the returned JSON data.
        - Generating the line chart using the Chart.js library.

Project Testing

The testing strategy focused on validating the data flow, functionality, and integration of the application.

- **Testing Methods Used:**

    - **Unit Testing:** Validating individual JavaScript functions including the API call and the data parsing logic.
    - **Integration Testing:** Making sure that the openmeteo.com API call successfully retrieves data and passes it accurately to the Chart.js library for display.
    - **System and Performance Testing:** Simulating the full user experience from inputting the city or zip code to displaying the chart output, ensuring the required non-functional requirement of returning the UV indices within three seconds was met.
    - **User Acceptance Testing (UAT):** Testing on separate independent webpage to confirm that the tool was intuitive and met the needs of the end-users (patients) and the organization (SSCTCoA).
    - **Browser Compatibility Testing:** Ensuring functionality on widely used browsers as required by non-functional requirements.

- **Results:** Testing achieved a simulated solution without errors and was confirmed ready to deploy.

Project Implementation

The solution was successfully **deployed** and **integrated** into the patient portal on the SSCTCoA website on **11/25/2025**. This deployment involved integrating the HTML, CSS, and JavaScript files into the new website structure and confirming connectivity to the external API from the live server environment.

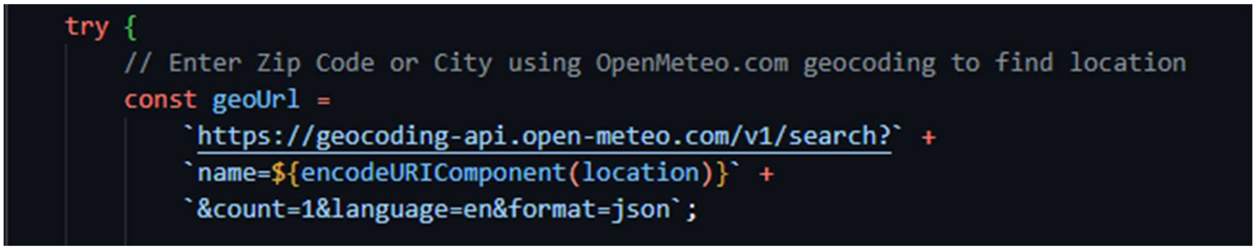
Professional Responsibilities

Ethics, Legality, and Security

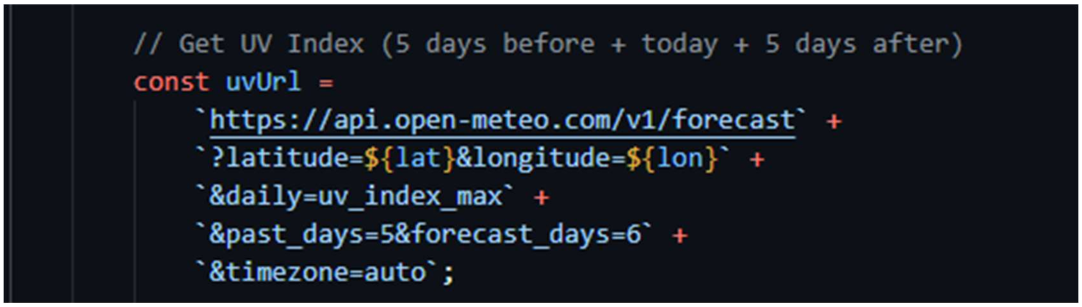
The development and deployment of the UV Index Chart Tool adhered to the highest standards of professional conduct and legal compliance.

- **Ethical Self-Awareness:** The team followed the guidelines and principles of the ACM Code of Ethics and Professional Conduct with a particular focus on public good, honesty, and data privacy. The promise was to provide a high-quality, reliable, and secure tool that benefits patient health.

- **Ethical and Legal Issues Addressed:**
  - **Data Privacy and Minimization (HIPAA):** A major design consideration was to develop a tool that does not store any patient information or records. The tool is client-sided and only retrieves public weather data based on user-provided city or zip code. This design ensures that the tool does not handle Protected Health Information (PHI) which mitigates HIPAA compliance risks.
  - **Intellectual Property/Licensing:** The terms of service for the use of the openmeteo.com API were reviewed for compliance. The use of the Chart.js library was confirmed to be permissible under its open-source license.
  - **Security:** Security risks are minimal with client-side tools, but secure coding practices were used to ensure that API calls are only made over secure protocols (HTTPS).



```
try {
    // Enter Zip Code or City using OpenMeteo.com geocoding to find location
    const geoUrl =
        `https://geocoding-api.open-meteo.com/v1/search?` +
        `name=${encodeURIComponent(location)}` +
        `&count=1&language=en&format=json`;
```



```
    // Get UV Index (5 days before + today + 5 days after)
    const uvUrl =
        `https://api.open-meteo.com/v1/forecast` +
        `?latitude=${lat}&longitude=${lon}` +
        `&daily=uv_index_max` +
        `&past_days=5&forecast_days=6` +
        `&timezone=auto`;
```

Support and Integration

Ongoing Training

- **Employee Training:** The existing SSCTCoA IT department staff received training on the structure, deployment environment, and maintenance procedures for the new tool, including troubleshooting common API or chart issues for both functionality and patient assistance.

- **Patient Tutorial:** A video demonstration/tutorial for patient end-users was created and integrated near the tool interface to ensure patients understand how to input their city or zip code and interpret the results in the UV index chart.

Long-Term Support and Maintenance

- **Alignment:** The UV Index Chart Tool is a crucial jigsaw piece that aligns perfectly with the organization's goal of improving patient outcomes and utilizing the redesigned website.

- **Maintenance and Support:**
  - The organization's IT department is responsible for monitoring the UV tracking tool's performance and connectivity.
  - The primary maintenance concern is API stability. A secondary plan should be in place in case the openmeteo.com API becomes corrupt or non-functional.
  - Regular reviews should occur to ensure optimal performance.

- **Future Recommendations:** Future upgrades could include integrating the UV index data directly into the patient's existing UV exposure data for immediate risk analysis.

SUMMARY/LESSONS LEARNED

Project Takeaways

- **Positive Takeaways:**
  - **Schedule Adherence:** The project was completed on time and within the estimated labor hour budget, demonstrating effective project planning and execution by the project manager and the team.
  - **Technical Simplicity:** The choice of a client-side solution with an API minimized integration risks and ensured rapid development and deployment.
  - **Privacy:** By designing the tool to avoid storing PHI, the project avoided major ethical and legal hurdles (HIPAA).

- **Lessons Learned:**
  - **External Dependency Risk:** The project depends heavily on the external openmeteo.com API. Alternative strategies for API failure or unavailability should have been considered during the planning phase.

Website Compatibility: While compatibility with current IT platforms was a constraint, the "Potential compatibility issues based on a website designed by a different company" risk (from the project charter) required careful consideration during development, which added a small amount of extra labor.

List of Reference Sources

https://www.acm.org/code-of-ethics
https://www.hhs.gov/hipaa/index.html
https://gdpr.eu/what-is-gdpr/
https://en.wikipedia.org/wiki/Social_impact_assessment
https://www.projectmanagementdocs.com/
https://www.w3schools.com/js/js_api_intro.asp
https://www.w3schools.com/js/js_json.asp
https://openmeteo.com/
https://open-meteo.com/en/docs
https://github.com/chartjs/Chart.js/blob/master/LICENSE.md
https://www.tutorialspoint.com/system_analysis_and_design/index.htm
https://project-management.com/software-development-methodologies/
https://www.tutorialspoint.com/management_concepts/index.htm
https://git-scm.com/book/en/v2/Getting-Started-About-Version-Control
https://opensource.com/business/15/5/write-better-docs

Appendices

Hours Worked

Project Component	**Component Time (Hours)**	**Actual Time (Hours)**	**Status**	**Team Member(s)**
Concept Development	20	20	Complete	
Code Development	60	60	Complete	
Testing	20	20	Complete	
Deployment	10	10	Complete	
Total	**110 hours**	**110 hours**	**Complete**	

Documentation

- **Technical Documentation:** Comprehensive guide for the IT department on code structure, API keys, and deployment procedures.

- **User Tutorial:** Video demonstration/written instructions for end users (patients).

- **Testing Report:** Summary of testing cycles including any bugs found and corrected during the process.

- **Source Code**: https://melissae50.github.io/portfolio/uvIndexTracker/index.html