Solar Scape

Functional Specifications Document:

1. Introduction

* Solar Scape is a solar potential analysis tool. Used for analyzing solar potential for specified location based on user input parameters form Google MERRA-2 Single Level Diagnostics data set, and Calculate Solar Photovoltaic from the solar irradiance by considering parameters like GHI, DNI, DFI.

2. User Roles

* Public Users (homeowners, businesses, policymakers etc.) are clients and only interact with application for solar assessment. Can Create account and store their interaction with the application.
* Admin (Super User) has access all over the application and its management.

3. Core Functionalities

* Address/Location Input:
  + Allow users to enter their address or pinpoint their location on an interactive map.
  + Geo-search the address to retrieve accurate latitude and longitude coordinates.
* PV Potential Calculation:
  + Access and process solar irradiance data from Google MERRA-2 Single Level Diagnostics.
  + Calculate mean PV output (average daily, monthly, and yearly) based on location.
  + Display estimated PV output in clear and understandable units (e.g., kWh, MWh).
* User Management:
  + Manage User, create account, Store and download analysis.
  + User account management and authentication.
* Data Acquisition and Processing:
* Download and pre-process MERRA-2 data (GHI, DNI, DHI) for user-defined locations and time ranges.
* Integrate with Google Earth Engine for advanced geospatial analysis and data manipulation.
* Implement quality checks and validation procedures for downloaded data.
* Reporting and Downloading:
* Generate comprehensive reports summarizing the analysis results with maps, charts, and data tables.
* Allow users to download reports and raw data for further analysis.

5. Data Management and Security

* User data will be collected, stored, and protected.

6. Non-Functional Requirements

* **Performance:**
* System should handle large datasets efficiently with minimal lag.
* Work on optimization for efficient performance.
* API responses should be returned quickly for a responsive user experience.
* **Scalability:**
* System should be able to accommodate increased user load and data volume.
* **Security:**
* User data and system access must be protected with secure authentication and authorization mechanisms.
* Implementing microservices architecture for scalability and expansion.
* **Accessibility:**
* System should be accessible to users with different abilities and devices.
* Usability: Implement accessibility, Tool tips, Clear and Concise Graphs and Focus on UI/UX.