Hackathon Evaluation Report - Round 1

1. Team Information

- **Team Name:**
  - **MATLAB Maverick**
- **Team Members (Names & Roles):**
  - **Shukla Kushang Akshay- Team Leader (Roles -Project Lead & Algorithm Development, IoT & Sensor Integration, AI & Deep Learning Specialist)**
- **Institution/Organization:**
  - **Gujarat Technological University**
- **Contact Email:**
  - **kushangashukla1@gmail.com**

2. Problem Statement Understanding

- **Problem Statement:**
  - Solar energy has become a major force in sustainable energy due to the growing demand for renewable energy sources. However, the location of solar panels affects their effectiveness, taking into account elements including terrain, shade, and sunshine exposure. Inadequate positioning lowers return on investment and energy output.

- **Key Challenges Identified:**
  - Sunlight exposure analysis considering seasonal variations.
  - Shading effect assessment to minimize energy loss.
  - Integration of IoT for real-time sensor data acquisition.
  - Dynamic panel rotation system for maximum efficiency.
  - AI-based prediction for energy output forecasting (Future Enhancement).

3. Proposed Solution Approach

- **Solution Overview:**
  - Our system is a MATLAB-based optimization program that uses AI-based energy forecasting, IoT sensor integration, and GIS mapping to determine the optimal location for solar panels. In order to optimize energy output, it examines environmental parameters in real time.

- **Technologies/Tools Planned to be Used:**
  - MATLAB (Optimization Toolbox, Image Processing Toolbox, Mapping Toolbox)
  - IoT Sensors (Simulated for Real-Time Irradiance & Temperature Monitoring)

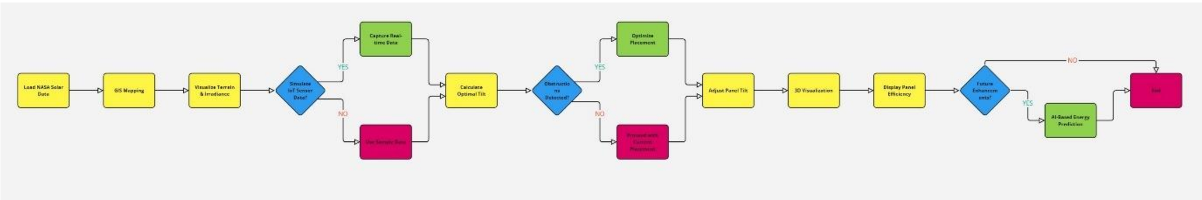
➢ GIS Mapping for Terrain Analysis
➢ Smart Tracking System for Dynamic Panel Rotation
➢ AI-Based Prediction Models (Future Enhancement)

- **Unique Selling Proposition (USP) or Innovation in the Approach:**

  ➢ Combining GIS, IoT, and Smart Tracking for enhanced solar energy optimization.
  ➢ Real-time sensor data processing for accurate tracking and efficiency improvements.
  ➢ Machine Learning predictions for seasonal energy adjustments (Future Enhancement).

4. Block Diagram / Architecture Design

- **Diagram illustrating the high-level design or architecture:**



- **Explanation of Key Components & Flow:**

  ➢ **Load NASA Solar Data** – Collects solar radiation data.
  ➢ **GIS Mapping** – Maps terrain for visualization.
  ➢ **Visualize Terrain & Irradiance** – Displays sunlight exposure.
  ➢ **Validate IoT Sensor Data?** – Decides between real-time or sample data.
  ➢ **Capture Real-time Data** (if YES) – Uses live solar data.
  ➢ **Use Sample Data** (if NO) – Uses stored data.
  ➢ **Calculate Optimal Tilt** – Determines the best panel angle.
  ➢ **Obstructions Detected?** – Checks for physical barriers.
  ➢ **Optimize Placement** (if YES) – Adjusts panel position.
  ➢ **Proceed with Current Placement** (if NO) – Keeps existing position.
  ➢ **Adjust Panel Tilt** – Fine-tunes tilt angle.
  ➢ **3D Visualization** – Displays layout.
  ➢ **Display Panel Efficiency** – Shows energy output.
  ➢ **Future Enhancements?** – Decides improvement scope.
  ➢ **AI-Based Energy Prediction** (if YES) – Uses AI for forecasts.
  ➢ **End** (if NO) – Concludes the process.

5. Step-by-Step Execution Plan:

- **Milestones & Expected Timelines:**

  ➢ 1. Week 1: Develop MATLAB-based GIS mapping and shading analysis.

(Completed)

- 2. Week 2: Integrate IoT sensor simulation for real-time irradiance tracking. (Completed)
- 3. Week 3: Implement smart tracking system for dynamic panel adjustment. (Completed)
- 4. Week 4: AI model training for energy prediction. (Future Enhancement)
- 5. Future Expansion: Cloud-based monitoring and weather-based adjustments. (Future Enhancement)

- **Current Status:**
  - GIS-based solar panel placement visualization.
  - IoT sensor data simulation for irradiance & temperature.
  - Smart tracking system to adjust panel tilt dynamically.
- **Potential Bottlenecks & Risks Identified:**
  - Real-time IoT sensor integration for non-simulated data.
  - Optimizing AI model accuracy for different terrains and weather conditions (Future Enhancement).
  - Hardware implementation of tracking system for physical panel rotation.

6. Partial Code & Initial Results:

- **Code Snippets (if any) or Link to Repository:**
  - **Sample Data:**

```
% Load the dataset
data = readtable('nasa_solar_large_data.csv');

% Display first few rows
disp('Sample Data:');
disp(data(1:5, :));

% Extract data columns
latitude = data.Latitude;
longitude = data.Longitude;
solar_irradiance = data.Solar_Irradiance_W_m2;
elevation = data.Elevation_m;
shading_factor = data.Shading_Factor__;
```

  - **Solar Irradiance & Shading Factor Analysis:**

```
%% Step 6: Visualize the Data and Results
figure;
scatter3(latitude, longitude, solar_irradiance, 100,
shading_factor, 'filled');
```

```
▪ xlabel('Latitude'); ylabel('Longitude'); zlabel('Solar Irradiance (W/m²)');
▪ title('Solar Irradiance & Shading Factor Analysis');
▪ % % colorbar;
```

➢ **Optimal Solar Panel Location and Estimated Energy Efficiency:**

```
▪ % Display optimal location for maximum energy efficiency
▪ [~, best_idx] = max(effective_irradiance);
▪ fprintf('Optimal Solar Panel Location: Lat = %.2f, Lon = %.2f\n', latitude(best_idx), longitude(best_idx));
▪
▪ %% Step 7: Compute Final Energy Efficiency at Optimal Location
▪ optimal_energy_efficiency = (1 - shading_factor(best_idx) / 100) * cosd(optimal_tilts(best_idx));
▪ fprintf('Estimated Energy Efficiency at Best Location: %.2f%%\n', optimal_energy_efficiency * 100);
```

➢ **Optimized Solar Panel Locations Based on GIS Data:**

```
▪ % Step 8: GIS Mapping: Visualizing Optimal Solar Panel Locations
▪ figure;
▪ geoscatter(latitude, longitude, 100, effective_irradiance, 'filled'); % Plot points
▪ geobasemap satellite; % Use satellite map for better visualization
▪ title('Optimized Solar Panel Locations Based on GIS Data');
▪ colorbar; % Show color scale for irradiance intensity
```

➢ **IoT Sensor- Solar Irradiance & Temperature:**

```
▪ %% Step 9: IoT Sensor Integration (Simulated)
▪ % Simulate real-time IoT sensor data (Replace with actual sensor input)
▪ iot_solar_irradiance = 850 + randn() * 10; % Simulated irradiance data (W/m²)
▪ iot_temperature = 30 + randn() * 2; % Simulated temperature data (°C)
▪
▪ fprintf('IoT Sensor - Solar Irradiance: %.2f W/m² | Temperature: %.2f°C\n', iot_solar_irradiance, iot_temperature);
```

➢ Adjusted Panel:

```
%% Step 10: Smart Tracking System (Dynamic Panel Rotation)
function new_tilt = adjustTilt(current_irradiance)
    % Adjust panel tilt dynamically based on solar irradiance
    if current_irradiance < 800
        new_tilt = 5; % Increase tilt for better exposure
    elseif current_irradiance > 900
        new_tilt = -5; % Reduce tilt for optimal output
    else
        new_tilt = 0; % Maintain current tilt
    end
end

% Adjust tilt based on simulated IoT sensor data
panel_tilt_adjustment = adjustTilt(iot_solar_irradiance);
fprintf('Adjusted Panel Tilt: %d°\n', panel_tilt_adjustment);
```

➢ GitHub Repository: https://github.com/KushangShukla/OptimizingSolarPanelPlacement

- **Preliminary Results:**

Sample Data:

Latitude	Longitude	Solar_Irradiance_W_m2	Elevation_m	Shading_Factor__
31.112	-122.71	921.65	185.56	18.78
37.184	-122.07	750.84	124.22	13.9
34.928	-116.11	939.68	79.36	8.51
36.539	-121.49	847.84	226.59	11.69
37.297	-120.56	762.94	64.2	18.41

Fig.1-Sample Data

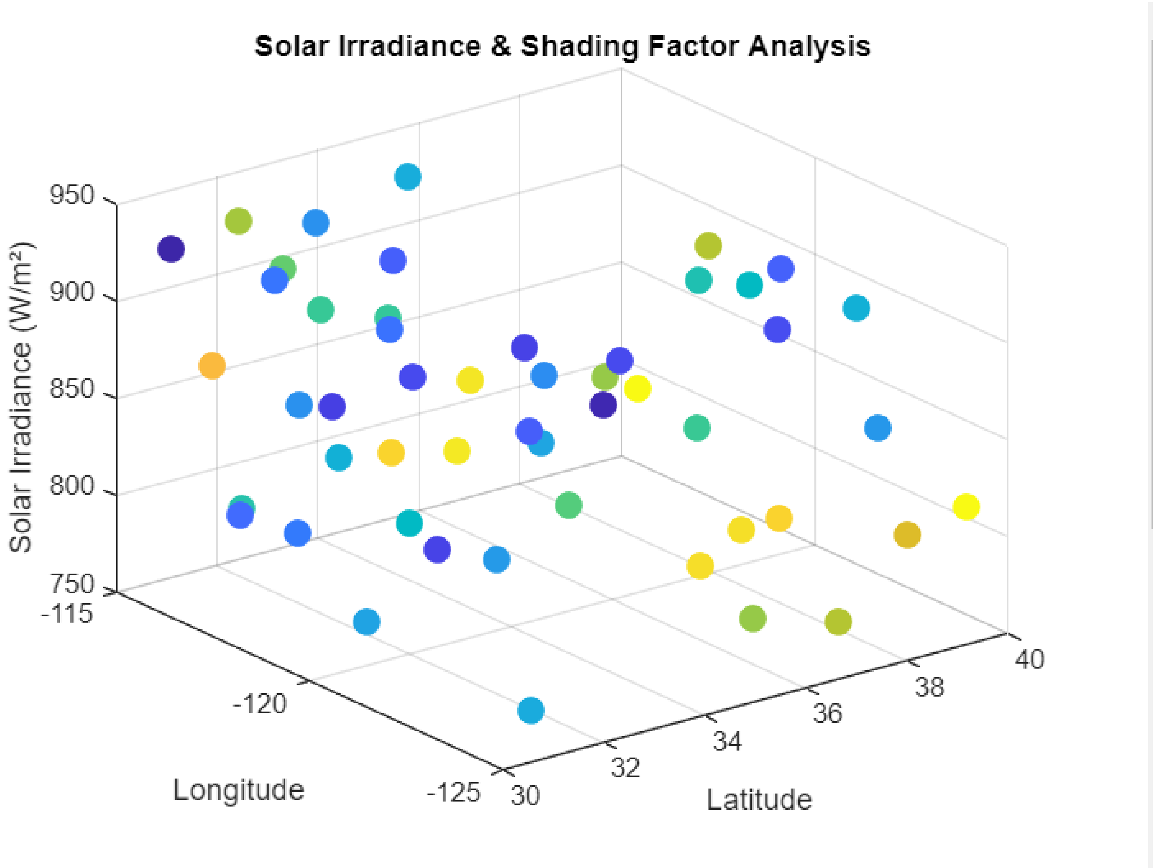


Fig.2- Solar Irradiance & Shading Factor Analysis

```
Optimal Solar Panel Location: Lat = 30.77, Lon = -115.40
Estimated Energy Efficiency at Best Location: 97.06%
```

Fig.3- Optimal Solar Panel Location and Estimated Energy Efficiency

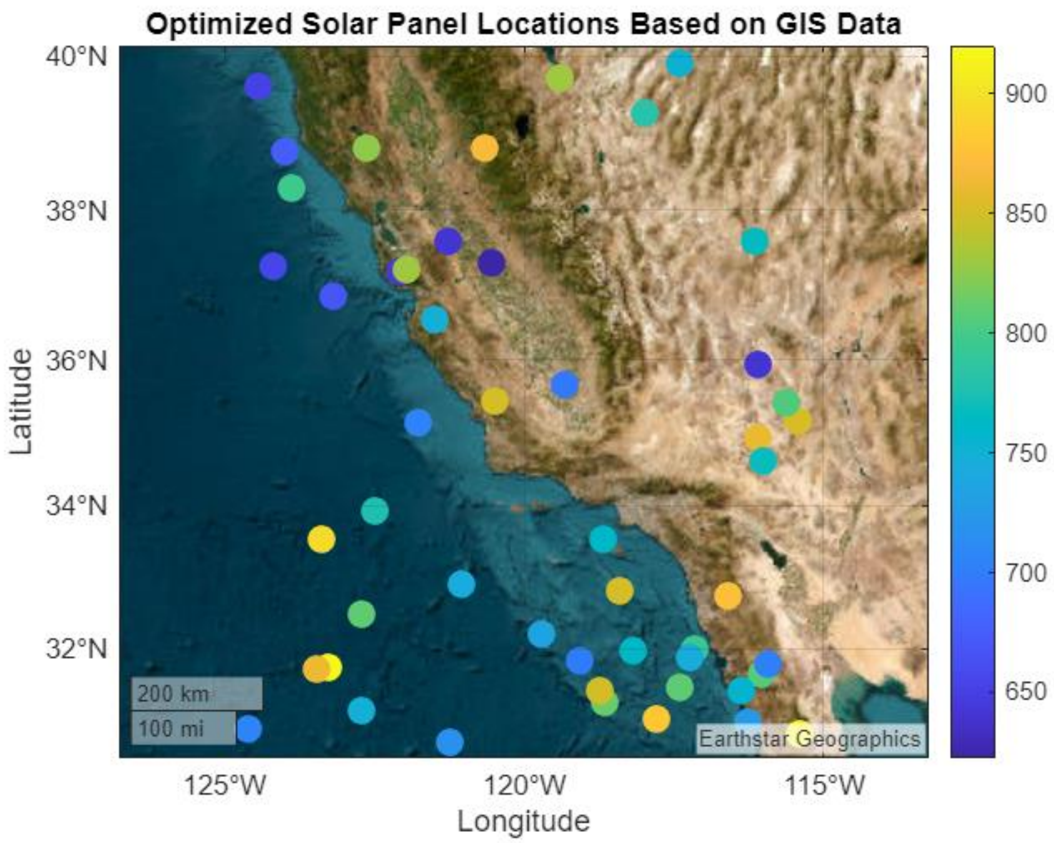


Fig.4- Optimized Solar Panel Locations Based on GIS Data

```
IoT Sensor - Solar Irradiance: 853.19 W/m² | Temperature: 27.38°C
```

Fig.5- IoT Sensor- Solar Irradiance & Temperature

```
Adjusted Panel Tilt: 0°
```

Fig.6-Adjusted Panel

7. Conclusion & Next Steps

- **Conclusion:**

  - By bridging GIS mapping, IoT sensor integration, AI-based forecasting, and smart tracking, this solution improves the efficiency and intelligence of solar panel placement. Future developments will concentrate on cloud connectivity and hardware integration to create a completely self-sufficient smart solar energy system.

- **Next Steps:**
    - Integrate real-time IoT sensor readings from physical hardware.
    - Develop AI-based energy prediction model for seasonal analysis. (Future Enhancement)
    - Optimize smart tracking system for better panel rotation control.
    - Develop cloud-based monitoring for remote access to solar efficiency data. (Future Enhancement)