Problem Statement 2: Next-Gen EMS: Smart Energy Management System (EMS) with Real-Time Monitoring

India’s growing reliance on renewables and EV adoption demands robust microgrid management. While renewable energy (solar, wind) offers sustainability, variability in generation and rising EV charging loads often cause voltage fluctuations, frequency instability, and unplanned downtime.

Technicians often face another barrier: diagnosing faults in EV components (motors, inverters, BMS) or renewable assets (panels, inverters, storage systems) is highly complex. Today, most operators lack end-to-end visibility across the system, forcing reactive fixes instead of proactive management.

A cloud-based EMS platform with real-time IoT-driven monitoring, intelligent diagnostics, and adaptive decision-making can change this landscape enabling operators to track energy flows, detect issues early, and respond to dynamic system conditions with confidence.

Objective

Design a cloud-enabled Smart EMS that provides real-time monitoring, diagnostic insights, and adaptive scheduling for renewables, storage, grid, and EV subsystems. The backend should ingest IoT data streams, analyze them for performance and health, and deliver actionable alerts through a unified dashboard. Reinforcement Learning (RL) can be used to recommend dispatch and charging/discharging strategies under uncertain conditions.

Data to be Used

- IoT Time-Series Data: Simulated or open datasets for voltage, current, frequency, harmonics, temperature across renewables, batteries, and EVs.
- Solar/Wind Generation Data: Open datasets (e.g., NREL, UCI).
- Load Demand Profiles: Household/industrial demand datasets with EV charging peaks.
- Cloud Data Streaming Framework: Teams may simulate data ingestion from MQTT, Kafka, or similar.
- https://data.mendeley.com/datasets/y58jknpgs8/2
- https://www.kaggle.com/datasets/anikannal/solar-power-generation-data
- https://en.wikipedia.org/wiki/Global_Solar_Atlas
- https://data.mendeley.com/datasets/msxs4vj48g/1
- https://zenodo.org/records/6473455

- https://www.data.gov.in/catalogs/?ministry=Ministry%20of%20New%20and%20Renewable%20Energy

What Participants Need to Do:

- Build a backend system capable of ingesting **real-time IoT data streams** with all APIs are fully synchronized and integrated.
- Build pipelines and ensure scalable, real-time data storage and processing.
- Provide **clear diagnostic outputs** (e.g., “Battery nearing end of life” or “Inverter overload”) rather than just anomaly flags.
- Enable health indices for each subsystem: renewable, storage, EV charger, EV motor.
- Implement **real-time alerts** for critical issues (over-voltage, SoC too low, motor overheating).
- Alerts should include recommended action (e.g., “Switch load to grid,” “Schedule maintenance”).
- Formulate EMS scheduling as an RL problem with states (SoC, demand, solar output, costs).
- Train an RL agent to minimize cost & emissions while ensuring reliability.
- The advisory panel should show RL-based scheduling suggestions.
- All project work, including version control and team collaboration, should be managed through a GitHub repository.
- Cloud dashboard that displays:
  - Real-time status of all subsystems.
  - Health indices and diagnostic recommendations.
  - Alerts & recommended actions.
  - Energy dispatch flows (current vs. suggested by RL agent)

Expected Deliverables

1. **Cloud Monitoring Backend** – capable of real-time IoT data ingestion and processing.
2. **Diagnostics Module** – clear subsystem health and diagnostic insights.
3. **Alerts & Advisory System** – real-time alerts with suggested actions.
4. **RL Scheduler (Optional Bonus)** – adaptive dispatch and optimization engine.
5. **Unified Dashboard** – visualization of system status, health indices, alerts, and dispatch flows.
6. **Technical Report** – methodology, datasets, system architecture, diagnostic logic, and validation.
7. **Demonstration** – working prototype showing real-time or simulated IoT data streaming into the EMS dashboard.