

Application 2: Real-Time Data Processing System for Weather Monitoring

Objective:

Develop a real-time data processing system to monitor weather conditions and provide summarized insights using rollups and aggregates. The system will utilize data from the OpenWeatherMap API (<https://openweathermap.org/>).

Data Source:

The system will continuously retrieve weather data from the OpenWeatherMap API. You will need to sign up for a free API key to access the data. The API provides various weather parameters, and for this assignment, we will focus on:

- main: Main weather condition (e.g., Rain, Snow, Clear)
- temp: Current temperature in Centigrade
- feels_like: Perceived temperature in Centigrade
- dt: Time of the data update (Unix timestamp)

Processing and Analysis:

- The system should continuously call the OpenWeatherMap API at a configurable interval (e.g., every 5 minutes) to retrieve real-time weather data for the metros in India (Delhi, Mumbai, Chennai, Bangalore, Kolkata, Hyderabad).
- For each received weather update:
 - Convert temperature values from Kelvin to Celsius (user preference is possible).

Rollups and Aggregates:

1. Daily Weather Summary:

- Roll up the weather data for each day.
- Calculate daily aggregates for:
 - Average temperature

- Maximum temperature
- Minimum temperature
- Dominant weather condition (with reason)
- Store the daily summaries in a database or persistent storage.

2. Alerting Thresholds:

- Define user-configurable thresholds for temperature or specific weather conditions.
- Track the latest weather data and compare it with thresholds.
- Trigger alerts if thresholds are breached (displayed on console or through notifications).

Test Cases:

1. System Setup:

- Verify system starts and connects to OpenWeatherMap API.

2. Data Retrieval:

- Simulate API calls at intervals and check response.

3. Temperature Conversion:

- Test temperature conversion from Kelvin to Celsius or Fahrenheit.

4. Daily Weather Summary:

- Simulate multiple days and verify daily summaries.

5. Alerting Thresholds:

- Configure thresholds and trigger alerts based on data.

Bonus:

- Support additional weather parameters from the API like humidity, wind speed.
- Explore weather forecast retrieval and summaries for predicted conditions.

Evaluation:

- Functionality, correctness, and accuracy of data processing.
- Efficiency and completeness of test cases and scenarios.
- Code clarity and maintainability.
- (Bonus) Additional feature implementation.

Artifacts To be Submitted For the Assignments:

- Codebase (preferably GitHub)
- Readme with build instructions and design choices
- Readme should contain dependencies for running the application. E.g., webserver, Database containers (Docker or Podman).