* **IoT Sensor Simulation:**

During research, I discovered that **real IoT sensors can generate hundreds to millions of data points per second**, depending on the type (temperature, humidity, air pressure, wind speed, etc.).

* + **Current Approach**
    - I chose to simulate sensor readings using setInterval() in **Node.js**.
    - Each interval generates random values and pushes them to the backend/dashboard.
    - This works well for small-scale testing and visualization.
  + **Limitations**
    - With many sensors or a high number of registered clients, setInterval() quickly becomes inefficient.
    - Each sensor runs independently, which can cause performance bottlenecks.
    - Scaling this approach in real-world IoT systems would not handle **high throughput**.
  + **Future Enhancement**
* **WebSockets** can be used to establish persistent two-way communication.
  + Instead of polling at intervals, the server pushes updates instantly to all connected clients.
  + More efficient for handling multiple sensors and live dashboards.
* **Public Weather API:**

Initially, I used the OpenWeatherMap API to fetch current weather data. However, when I started considering data visualization and displaying multi-day forecasts, I realized that OpenWeatherMap was not ideal, as it lacks accurate and detailed forecast data for these purposes. So I used WeatherAPI for that purpose.

* **Charts:**

I used recharts library (AreaChart) for visualizing different metrics and comparing between IOT Sensor data and Public API.