Dante Berouty & Thomas Nguyen

Professor Luti

CECS 327

December 10th 2024

Assignment 8 Report

System Architecture and Approach

The system is modular, designed for efficient collection, processing, and analysis of IoT data. It includes two smart refrigerators and a smart dishwasher, each with sensors for data collection. The dishwasher uses an ammeter for electricity consumption and a Water Flow Sensor for water usage, while the refrigerators feature an ammeter, thermistor, and moisture meter for monitoring electricity, temperature, and humidity. Sensor data is routed through a broker to a MongoDB database. Users can interact with the system via a TCP server hosted on a Google Cloud VM, which processes three predefined queries to retrieve and analyze relevant data. The client can communicates with the server using its public IP.

Key components

- Google Cloud Virtual Machines: Host both the server and client applications.
- TCP Socket Server: Facilitates querying of IoT data by processing user defined queries.
- MongoDB Database: Stores IoT data and metadata.
- Dynamic Device Integration: Devices and sensors are managed dynamically using metadata, allowing for scalability.
- Predefined Queries: Three specific queries related to moisture, water usage, and electricity consumption.
- Algorithms: Implemented to calculate averages, perform unit conversions, and evaluate thresholds as needed.

Research Findings on IoT Sensors and Data

Sensor Name	Description	Sensor Range	Desired Value	Unit of Measurement
Moisture Meter (from Dataniz)	Measures humidity levels to track the fridge's internal environment.	N/A	N/A	Relative Humidity (%)
YF-S201 Water Flow Sensor	Digital pulse sensor that measures water flow rate using a turbine wheel.	1–30 L/min (Flow Rate)	0–10 L/min	Liters per Minute (L/min)
PZEM-004T AC Power Meter	Measures AC voltage, current, power, and energy consumption for appliances.	0–9999 kWh	0–2 kWh per cycle	Kilowatt-Hours (kWh)

Metadata Usage

The metadata from Dataniz is essential for organizing and querying the IoT system by providing a structured map of devices and sensors. Each device document includes unique identifiers (assetUid), names, sensor types, units, and hierarchical relationships, allowing the system to identify which sensors belong to which devices.

By linking sensor UIDs from metadata to corresponding readings in the database, the system can flexibly handle user queries without hardcoding relationships. Metadata also supports accurate data analysis and conversions, such as interpreting moisture as relative humidity or converting water flow into gallons. In summary, metadata transforms static IoT data into a scalable system that dynamically organizes and processes user queries.

Challenges Faced

One of the main challenges was understanding and setting up metadata for the project. It was difficult at first to see how metadata could help manage devices and sensors or be used to make the system scalable.

Another issue was optimizing database queries to make them faster and more efficient. We had to improve the way data was retrieved and processed to ensure the system could handle user queries smoothly.

Feedback for Dataniz.com

Strengths:

- The platform makes it easy to create and test IoT devices and sensors.
- Metadata management tools are helpful for organizing and keeping track of devices.
- Generate and view visualization tool is easy to understand.

Areas for Improvement:

- It's hard to edit sensor, board, and device names once they are created.
- The free version limits us to three devices and three sensors per device, which makes it hard to build more realistic, complex setups.
- You can’t delete metadata once it’s created.
- The longitude and latitude fields don’t match real-world locations, which creates inaccuracies.