MINISTRY OF EDUCATION AND RESEARCH

TECHNICAL UNIVERSITY
OF CLUJ-NAPOCA, ROMANIA

DISTRIBUTED SYSTEMS

ASSIGNMENT 3

Online Energy Utility Platform

Remote Procedure Call (RPC)

Reckerth Daniel Peter

30444

Contents

I. Assignment Objective

The aim of this project is to create a system for an online energy utility platform designed to manage clients and their associated devices which are equipped with smart sensors monitoring energy consumption.

In the third assignment we suppose that clients have intelligent home appliances that can be remotely controlled using RPC. Each device can communicate with the server to compute a time when the device will start to produce an optimal energy consumption.

The main abilities are as following:

- Retrieving the client hourly historical energy consumption over D days – user input
- Compute averaged/baseline energy consumption for the client over the past week (7 days)

$$Baseline(h) = \frac{1}{7}\sum_{d=1}^{7} E_{client}^{d}(h), \qquad h = 1, \dots, 24$$

- Allows the selection of a program with duration
- Computes the best time to be started considering the baseline and the program duration to avoid energy peaks from client

II. Functional Requirements

The function requirements are the following:

- Display a chart with client historical energy consumption
- The client application displays the client baseline – reference consumption
- The client application allows the selection of a program with duration
- The client application asks the server for the best start time in the next day to minimize peaks of energy consumption

III. Design and Implementation

For developing this system we have decided to use the following technologies:

- REST services for backend application – Java Spring
- Frontend – React
- For database – Postgres
- Deployment – Heroku and Docker
- RPC – using JSON-RPC 2.0

While working on the application I had to design and implement a configuration class and service class where the RPC will take function. In the configuration class common RPC topics are taken care of while in the service we implement the following requirements having methods for computing the historical energy consumption and the baseline one.

In frontend we have a new component called HistoricalEC where we render and display the two charts and we also implement the RPC call.

IV. Conclusions

This improved to be a good assignment learning about RPC and difference between controller endpoints and them.

Bibliography

- Provided documentation for Docker and CI/CD on Heroku
- http://solarcitynotes.blogspot.com/p/json-rpc.html