SYP Documentation

Introduction

We are Energy Abacus. Our application monitors plugs and displays given information in our Ionic Frontend. So far, we have implemented the receiving of information from our plugs in our Frontend. But how does that work?

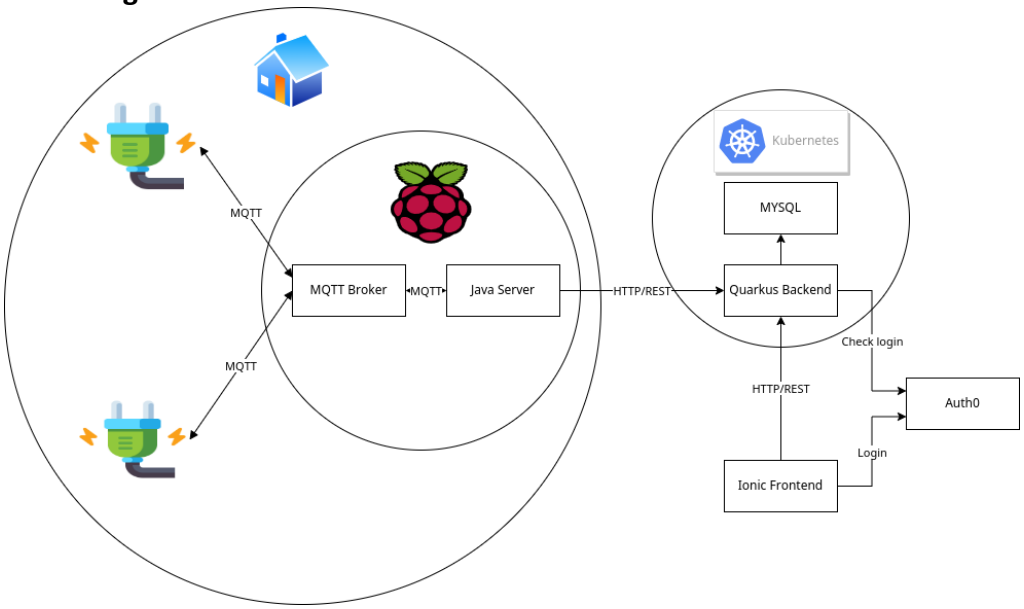
Explanation

Both plug and MQTT Broker must be connected via settings. After that the MQTT system works and can be used by other software such as the Java Server in our example. The Java Server subscribes to the Plugs via MQTT Broker and receives the information (temperature, wattPerMinute, etc..) each time the Plug posts to the Broker. The Java Server regulates all plugs which allows sending proper information to our Quarkus Backend. The Quarkus application offers REST endpoints for the Frontend and stores the information in a MySQL.

Security Measurements

Our main branches are protected. To change something, you need to create your own branch, make a pull request, wait for the automated checks to complete and someone of the team to do a code review.

From Plug to Ionic Frontend:



Links and Information

Scrum Documentation: https://vm81.htl-leonding.ac.at/agiles/99-337/current
Auth0: https://manage.auth0.com/dashboard/us/dev-3adocs3bconafo8d/
Quarkus Backend: https://student.cloud.htl-leonding.ac.at/e.gstallnig/abacus/main/
Swagger: https://student.cloud.htl-leonding.ac.at/e.gstallnig/abacus/main/q/swagger-ui/

LeoCloud Pods:

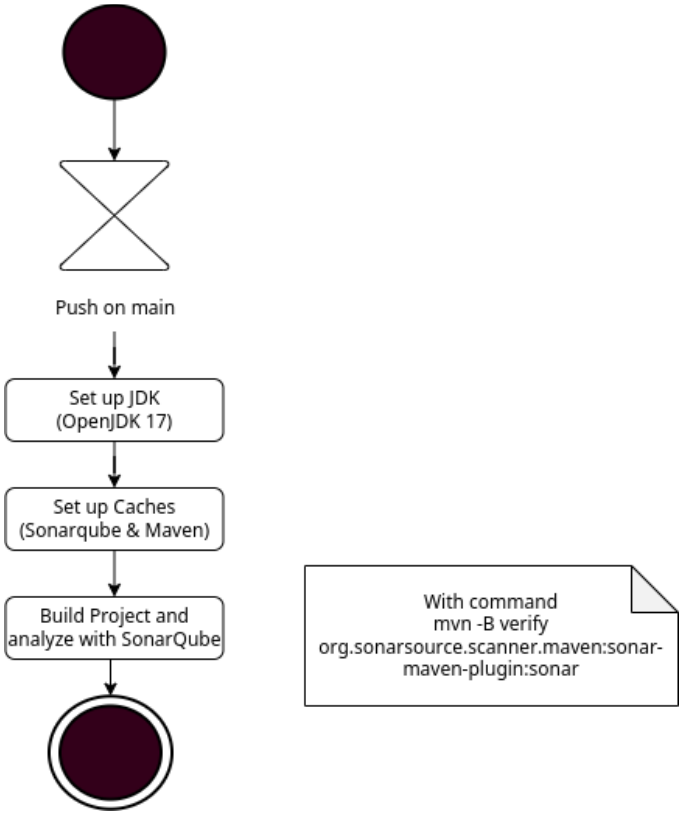
- MySQL
- Quarkus (Backend)

How to setup Hub:

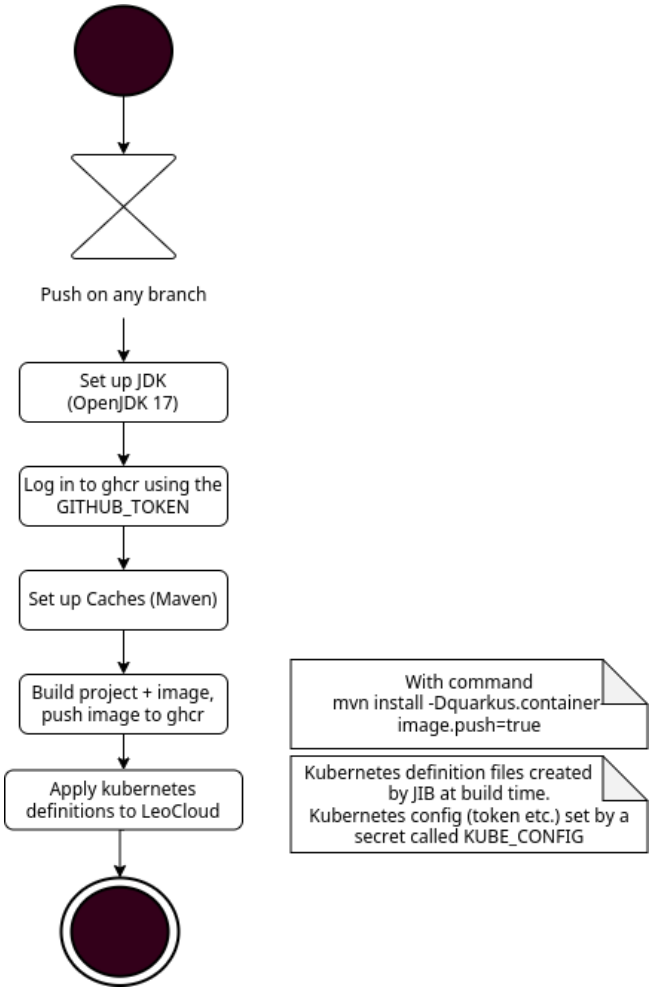
1. Setup a new Raspberry Pi
2. sudo apt-get update
3. sudo apt-get upgrade
4. sudo reboot
5. After rebooting run this command:
   curl -sSL https://raw.githubusercontent.com/Energy-Abacus/Hub/main/setup.sh | sudo bash
6. Enjoy!

Github Actions

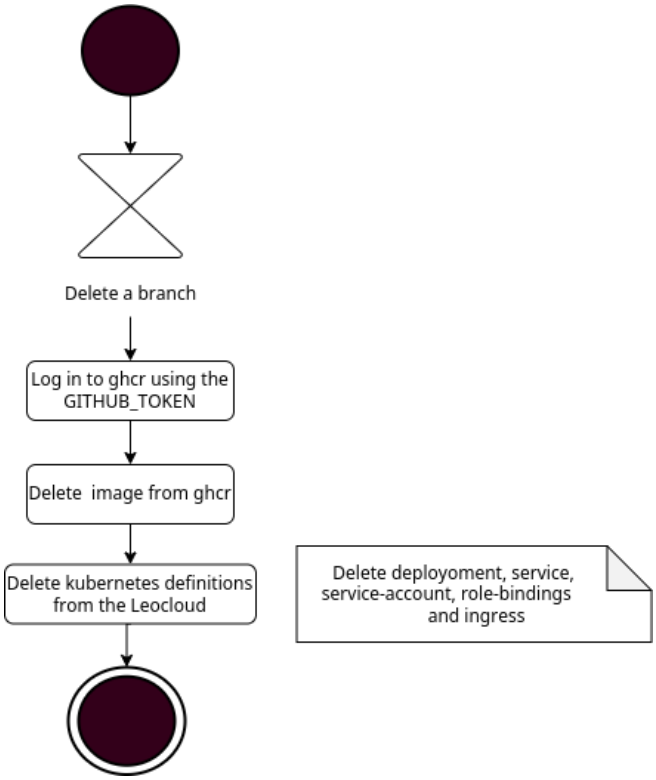
Push on Main:



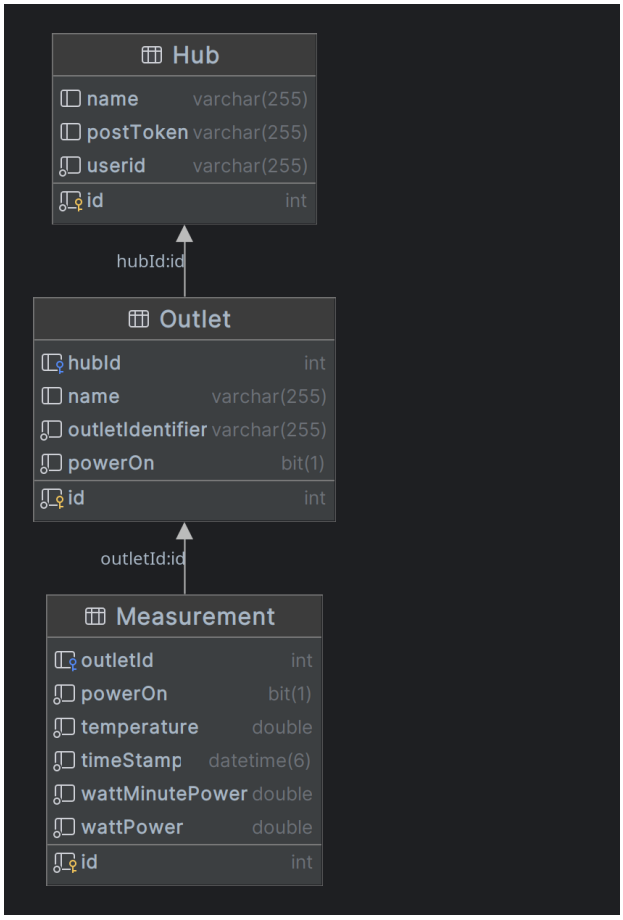
Push on Branch:



Delete Branch:



ERD:


Hub
name varchar(255)
postToken varchar(255)
userid varchar(255)
id int
hubId:id
Outlet
hubId int
name varchar(255)
outletIdentifier varchar(255)
powerOn bit(1)
id int
outletId:id
Measurement
outletId int
powerOn bit(1)
temperature double
timeStamp datetime(6)
wattMinutePower double
wattPower double
id int