

Github ứng dụng

<https://github.com/dqhieuu/pub-sub-java-cli>

Trong này chứa mã nguồn, hướng dẫn cài đặt, ảnh minh hoạ cho ứng dụng.

Giao thức MQTT

Publisher	Broker	Subscriber
HELO AS PUB →	200 OK	← HELO AS SUB
PUB topic/location/sensor <message> →	Gửi đến sub phù hợp: topic/location/sensor <message> →	Không response
	201 OK / 401 Invalid Topic	← SUB topic/location/sensor
	202 OK / 402 Invalid Topic	← UNSUB topic/location/sensor
	100 Bye Bye (xóa subscriber khỏi danh sách subscribe)	← QUIT

Design 1 smart home

Giá trị beacon truyền đi (có thể từ 2-60HZ) (tức là truyền từ 2 lần/giây)

- Công tắc (cho bóng đèn, tv, gara, ...): giá trị {0,1} **TYPE 1**

- Nhiệt độ:

~~Cách 1: Ký hiệu kèm hệ đo, vd: "15.234 F", "20.123 C",~~

Cách 2: Dùng chung 1 hệ là Kelvin, không ghi kèm hệ đo, VD: 298.152 **TYPE 2**

- Âm lượng, độ ẩm, độ sáng bóng đèn: giá trị [0.000 -> 1.000] **TYPE 3**

~~-Bóng đèn led rgb: (200,100,253)~~

~~-Bóng đèn led chỉnh màu theo độ lumen: 5000, 7000....~~

Tên các topic các loại thiết bị

- Type 1: light_switch, tv_switch, garage_switch

- Type 2: temp

- Type 3: volume_range, humidity_range, light_range

Cách subscriber (ứng dụng theo dõi smart home) lưu giá trị:

Nhận tất cả thông tin từ broker gửi đến, lưu vào từ điển tên các thiết bị khác nhau, kèm các thông số như:

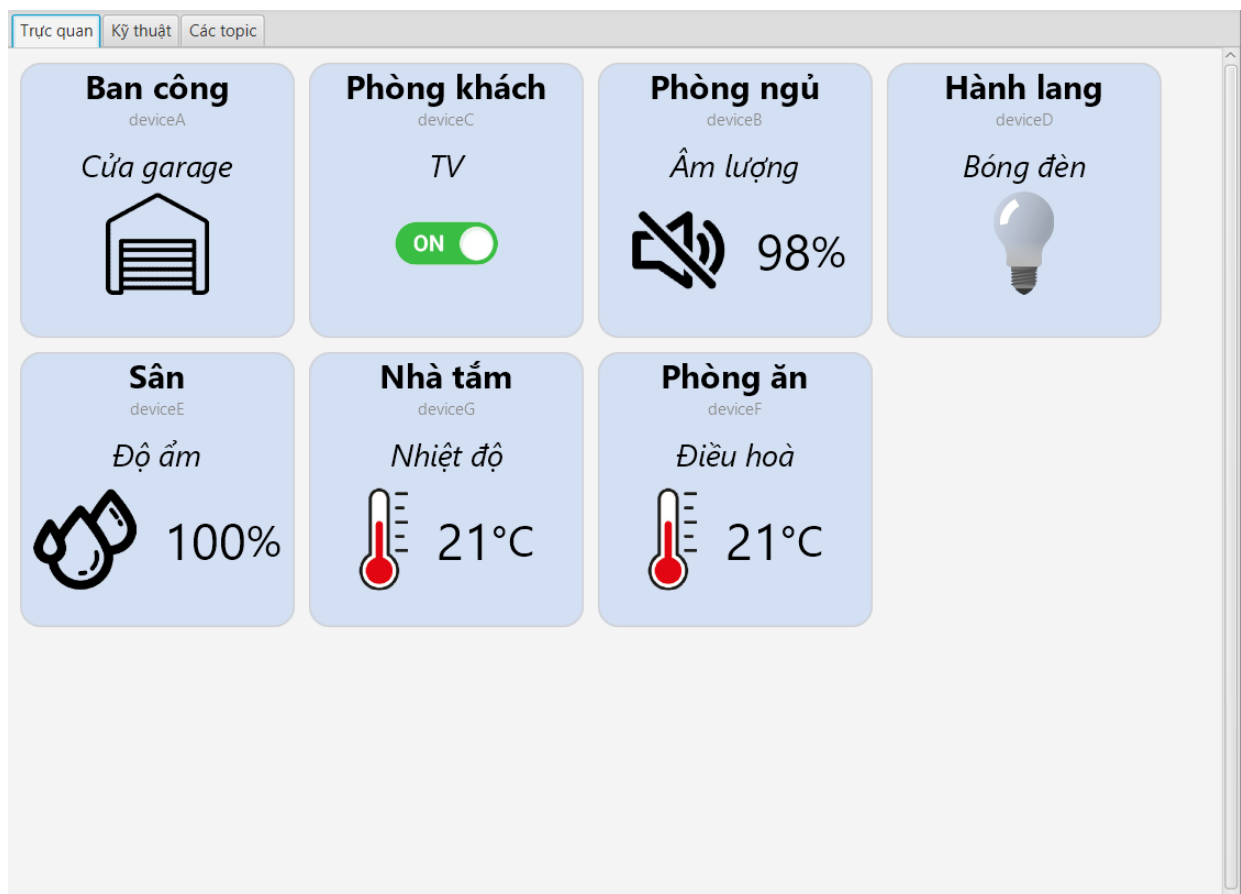
```
{  
    heartBeat: timestamp lần nhận giá trị mới nhất từ thiết bị này  
    value: giá trị mới nhất nhận được  
}
```

Sử dụng 1 luồng riêng, refresh **mỗi giây (hoặc 60hz)** để đọc các giá trị heartBeat từ từ điển và update render MVC:

Nếu heartBeat của thiết bị nào vượt quá **60 giây** so với timestamp bây giờ, loại thiết bị khỏi từ điển.

Ảnh giao diện

Subscriber



Trực quan

Kỹ thuật

Các topic

Tên topic

b/a/d

Thêm topic mới

Tên topic

+ / + / +

Xoá topic

Publisher

