



Hình - Sơ đồ triển khai mô hình IoT.

Mô tả tổng quan

Server contents: dùng để truyền các thông điệp (messages) giữa nhiều Client (Publisher và Subscriber). Trong đó nó là điểm giao của tất cả các kết nối đến từ client. Nhiệm vụ của Server Contents là nhận message từ publisher và xếp các message theo hàng đợi rồi chuyển chúng tới một địa chỉ cụ thể.

Publisher và Subscriber làm một trong 2 việc là publish các message lên một topic cụ thể hoặc subscribe một topic nào đó để nhận message từ topic này.

- **M_1:** là các thiết bị IoT gửi/nhận các message lên một topic cụ thể
- **App_1:** là các ứng dụng dùng cho việc gửi/nhận dữ liệu từ các thiết bị IoT.

Mô tả chi tiết

Nhóm x sẽ gồm 1 ứng dụng có mã là App_x và tương tác với 1 mạch điều khiển M_y, trên M_y sẽ gồm các thiết bị sensor và thiết bị điều khiển gọi tắt lần lượt là các thiết bị M_y_i với i từ 1 tới n tùy hệ thống. Mỗi nhóm có App_x và M_y sẽ tương tác với server trên 2 topic có mã là T_(2*x) và T_(2*x-1) với x là số thứ tự nhóm (x từ 1). Trong đó T_(2*x-1) là topic để nhận dữ liệu từ cảm biến và T_(2*x) để nhận dữ liệu điều khiển thiết bị.

Giả sử nhóm 1 (x=1), có ứng dụng App_1 và mạch điều khiển M_1, tương tác qua Topic 1 và Topic 2 của server MQTT. Trong đó Topic 1 có chiều từ M_1 publish và Topic 2 có chiều từ App_1 publish. Cụ thể, định thời 3 phút, mạch M_1 gửi dữ liệu cho ứng dụng, sẽ publish Topic 1 payload (Data_D1) có chuỗi chứa các cặp key:value1, value2, value3, ... vd id1:1, 20, 3, v.v thể hiện value1 là trạng thái on, of, value2 thể hiện chỉ số tốc độ quạt, value3 ... của thiết bị quạt có mã id1.

Tùy loại thiết bị của các thiết bị bên trong mà có bao nhiêu loại value. Có thể định nghĩa lại payload theo định dạng JSON tùy cách thống nhất. Khi đó, App_1 subscribe Topic 1 để nhận dữ liệu về và xử lý chuỗi để gán lại các giá trị hiển thị cho từng thiết bị trên ứng dụng.

Chiều ngược lại, trên ứng dụng App_1, nếu cần tắt 1 thiết bị nào đó, thì App_1 sẽ publish lên Topic 2 payload (Data_D2) có nội dung như id1:0, bên thiết bị subscribe Topic 2 và nhận chuỗi trả về, phân tích và điều khiển thiết lập id1 tắt.

Lưu ý: 1 mạch được sử dụng cho nhiều nhóm CS, 1 nhóm CS chỉ tương tác với 1 mạch

Vậy cần thống nhất:

1. Quy tắc đặt Topic và format của payload giữa bên CS và CE để cùng làm việc (sử dụng JSON)

Ví dụ

Nhóm CS	Tên đề tài	Nhóm CE (tên nhóm, sdt liên hệ, email)	Mã số mạch điều khiển tương tác	Topic dùng	Thông tin thiết bị gồm: Mã thiết bị, tên thiết bị, loại thiết bị (out chỉ trừ dữ liệu, in có thể điều khiển), tên các thông số đo/lấy được	Format payload
1	...		M1: mạch điều khiển đo độ ẩm, nhiệt độ và điều khiển thiết bị quạt trong nhà	T1, T2	id1, sensor độ ẩm 1, out, value1 với value1 là giá trị độ ẩm ghi nhận được id1, quạt 1, in/out, value1, value2 với value1 là trạng thái quạt tắt hay mở, value2 là mức độ quạt	<p>quy định như sau payload cho T1 (chiều gửi dữ liệu từ device lên server)</p> <pre>[{ "device_id": "id2", "values": ["1", "3"] }, { "device_id": "1", "values": ["3.24"] }]</pre> <p>⇒ thiết bị id1 có giá trị độ ẩm là 3.24, thiết bị quạt có trạng thái là bật, cường độ 3</p> <p>payload cho T2 (chiều gửi dữ liệu từ mobile app lên server)</p> <pre>[{ "device_id": "id2", "values": ["0"] }]</pre> <p>⇒ yêu cầu tắt thiết</p>

						bị quạt
2	...		1	T3, T4	...	

Ghi chú: cách chia trên theo quy tắc mỗi nhóm chỉ cần tương tác với 1 mạch điều khiển trong đồ án. Với quy mô nhiều mạch thì cần thống nhất lại. Vd nếu đề tài là quản lý điều khiển các thiết bị cho nhiều toà nhà thì mỗi toà nhà là 1 mạch điều khiển. Do đó, cần xem lại tên đề tài của các nhóm để hoặc giới hạn quy mô, hoặc quy định thêm topic cho các nhóm.