

4

CÔNG NGHỆ
INTERNET OF THINGS
HIỆN ĐẠI

Sử dụng Dashboard và hoàn thiện cơ bản mô hình IoT

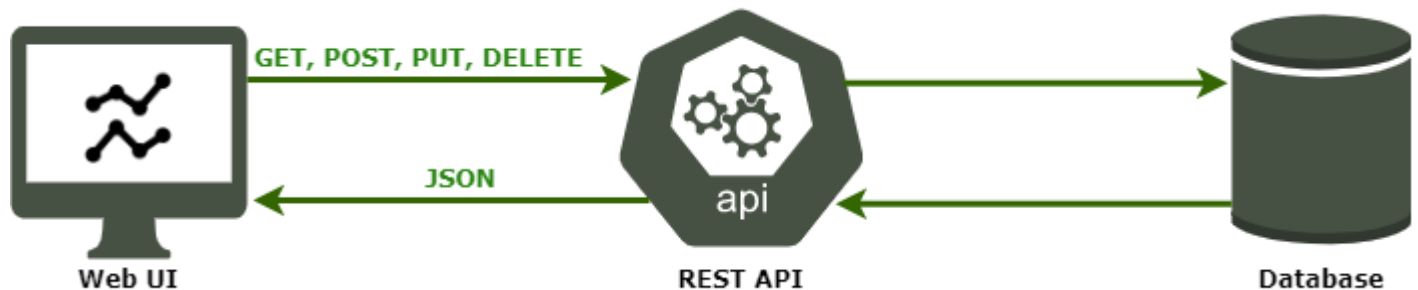
Lưu hành nội bộ

A. MỤC TIÊU

- Giới thiệu tổng quan về các nền tảng mã nguồn mã để lưu trữ, quản lý và hiển thị dữ liệu IoT.
- Giới thiệu các thành phần Web UI, API, Database trong mô hình IoT.
- Tổng quan về việc xây dựng, kết hợp các thành phần để tạo nên một mô hình IoT cơ bản hoàn chỉnh.

B. GIỚI THIỆU

1. Tổng quan mô hình thực hiện



Hình 1. Mô hình tổng quan

Tại bài thực hành này, sinh viên xây dựng Website hiển thị và điều khiển trong mô hình IoT gồm có 3 thành phần chính:

- **Web UI:** thành phần dùng để thể hiện các giá trị cảm biến, thực hiện các tác vụ liên quan đến điều khiển thiết bị và hiển thị logs cho người quản trị theo dõi.
- **API:** tương tự như ở bài thực hành trước đã giới thiệu, đây là thành phần trung gian để người dùng có thể tương tác với cơ sở dữ liệu, xử lý các tác vụ tính toán,...
- **Database:** cơ sở dữ liệu của hệ thống, giúp cấu trúc các dữ liệu, lưu trữ để phục vụ nhu cầu khai thác thông tin của nhiều người sử dụng từ một hoặc nhiều ứng dụng cùng lúc với nhiều mục đích khác nhau.

2. IoT platform

Các nền tảng IoT (Internet of Things) được sử dụng để quản lý và kiểm soát các thiết bị kết nối IoT, thu thập dữ liệu từ chúng, và thực hiện các hoạt động tự động hóa dựa trên dữ liệu thu thập. Một số chức năng chính của các nền tảng IoT:

- **Quản lý thiết bị:** Các nền tảng IoT cho phép quản lý, theo dõi và điều khiển các thiết bị IoT từ xa. Có thể thêm mới thiết bị, xóa thiết bị, cập nhật phần mềm của thiết bị, và theo dõi tình trạng của chúng.
- **Thu thập dữ liệu:** Các nền tảng IoT có khả năng thu thập dữ liệu từ các thiết bị IoT. Dữ liệu này có thể bao gồm thông tin về nhiệt độ, độ ẩm, áp suất, vị trí, tình trạng hoạt động, và nhiều thông tin khác.
- **Trực quan hóa dữ liệu:** Các nền tảng IoT cung cấp công cụ để biểu đồ, biểu đồ và hiển thị dữ liệu thu thập dưới dạng giao diện trực quan. Điều này giúp người dùng dễ dàng theo dõi và phân tích dữ liệu IoT.
- **Bảo mật:** Các nền tảng IoT cung cấp các tính năng bảo mật để bảo vệ dữ liệu và thiết bị khỏi các mối đe dọa trực tuyến. Điều này bao gồm xác thực, mã hóa dữ liệu và kiểm soát quyền truy cập.
- **Hỗ trợ giao thức:** Các nền tảng IoT hỗ trợ nhiều giao thức kết nối, bao gồm MQTT, CoAP, HTTP, và nhiều giao thức khác để giao tiếp với các thiết bị IoT.

Một số IoT platform mã nguồn mở:



3. API

API là viết tắt của **Application Programming Interface** – phương thức trung gian kết nối các ứng dụng và thư viện khác nhau. **REST (Representational State Transfer)** là một dạng chuyển đổi cấu trúc dữ liệu, một kiểu kiến trúc để viết API.

RESTful API là một tiêu chuẩn dùng trong việc thiết kế các API cho các ứng dụng web để quản lý các resource. RESTful là một trong những kiểu thiết kế API được sử dụng phổ biến ngày nay để cho các ứng dụng (web, mobile...) khác nhau giao tiếp với nhau.

Chức năng quan trọng nhất của REST là quy định cách sử dụng các phương thức HTTP (như GET, POST, PUT, DELETE...) và cách định dạng các URL cho ứng dụng web để quản các resource ¹.

4. Database

Giới thiệu cơ sở dữ liệu SQL

SQL là viết tắt của **Structured Query Language** có nghĩa là ngôn ngữ truy vấn có cấu trúc, là một ngôn ngữ máy tính để lưu trữ, thao tác và truy xuất dữ liệu được lưu trữ trong một cơ sở dữ liệu quan hệ.

SQL là ngôn ngữ chuẩn cho hệ cơ sở dữ liệu quan hệ. Tất cả các hệ thống quản lý cơ sở dữ liệu quan hệ (RDMS) như MySQL, MS Access, Oracle, Sybase, Informix, Postgres và SQL Server đều sử dụng SQL làm ngôn ngữ cơ sở dữ liệu chuẩn.

Giới thiệu cơ sở dữ liệu NoSQL

NoSQL là một hệ thống quản lý dữ liệu không quan hệ (**non-relational Data Management System**) dễ dàng mở rộng. Mục đích chính của việc sử dụng cơ sở dữ liệu NoSQL là dành cho các kho dữ liệu phân tán với nhu cầu lưu trữ dữ liệu lớn. NoSQL được sử dụng cho Dữ liệu lớn và ứng dụng web thời gian thực.

¹ Xem thêm tại: <https://topdev.vn/blog/restful-api-la-gi/>

Hiện nay, trên thị trường có khá nhiều NoSQL Database Management System: MongoDB, RavenDB, Neo4j,... Trong đó nổi bật có 4 loại như sau:

- Key-Value Database
- Document Database
- Column-Family Database
- Graph Database

C. YÊU CẦU & NỘI BÀI

1. Yêu cầu

Xây dựng ứng dụng IoT để giám sát chất lượng không khí và cung cấp dữ liệu theo thời gian thực trên nền tảng opensource IoT platform (Thingsboard, Openremote...)

- Tại phần thiết bị, sinh viên thực hiện sử dụng 2 loại thiết bị cảm biến DHT22/DHT11 và MQ-135 để lấy 3 loại dữ liệu cảm biến là nhiệt độ, độ ẩm, chất lượng không khí. Trên Wemos D1 sử dụng cảm biến MQ-135, đồng thời lập trình để gửi các giá trị cảm biến đến hệ thống thông qua giao thức HTTP. Trên Raspberry sử dụng DHT22/DHT11, đồng thời lập trình để gửi các giá trị cảm biến đến hệ thống thông qua giao thức MQTT.
- Biểu diễn dữ liệu: Cấu hình dashboard trên IoT platform để hiển thị dữ liệu theo thời gian thực từ Wemos D1 và Raspberry. Sử dụng các biểu đồ (charts/graphs) để thể hiện dữ liệu theo thời gian.

2. Yêu cầu nộp bài

- Sinh viên tìm hiểu và thực hành theo hướng dẫn. Thực hiện **nhóm**.
- Sinh viên báo cáo kết quả thực hiện và nộp bài bằng file. Trong đó:
 - Trình bày chi tiết quá trình thực hành và trả lời các câu hỏi nếu có (kèm theo các ảnh chụp màn hình tương ứng).
 - Giải thích các kết quả đạt được.

- Tải mẫu báo cáo thực hành và trình bày theo mẫu được cung cấp.

Nén tất cả các file và đặt tên file theo định dạng theo mẫu:

Mã lớp-LabX_MSSV1_MSSV2

Ví dụ: NT532.M21.1-Lab01_19520001_19520002

- Nộp báo cáo trên theo thời gian đã thống nhất tại website môn học.
- Các bài nộp không tuân theo yêu cầu sẽ **KHÔNG** được chấm điểm.

HẾT

Chúc các bạn hoàn thành tốt bài thực hành!