

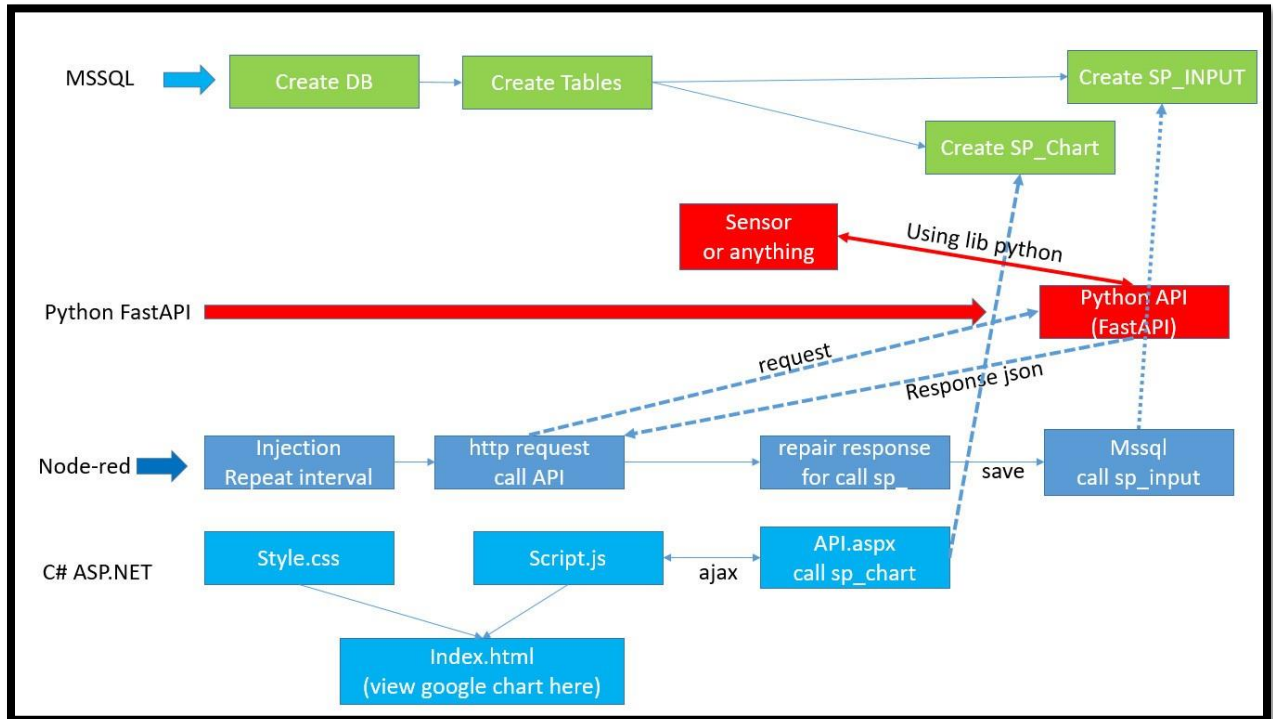
Họ và tên: Đỗ Văn Hiếu

Bài tập môn: Lập trình python

Mã SV: K205480106044

Giáo viên: Đỗ Duy Cốp

Đề bài: Em hãy thiết kế mô hình sau:



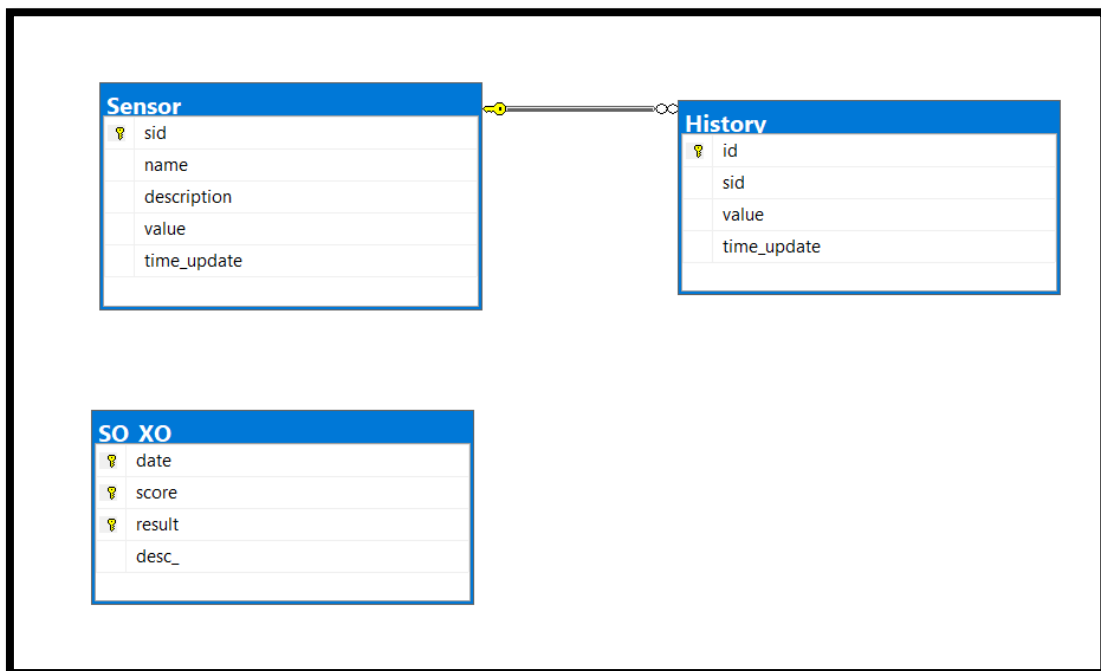
BÀI LÀM

Bài tập em thực hiện gồm 2 API chính từ fastAPI của python gồm hai route GET.

1. Route thứ nhất em trả về 1 giá trị random từ 1 -> 1000
2. Route GET thứ hai trả về kết quả số số độc đắc khu vực miền bắc mới nhất (crawling data từ trang: <https://www.kqxs.vn>)

1. MSSQL – SQL Server

Sql server là nơi lưu trữ dữ liệu để node-red đẩy data vào, cấu trúc các bảng:



Cấu trúc gồm 3 bảng, hai bảng trên có mục đích cho quá trình lưu sensor, bảng SO_XO sẽ lưu giá trị số xổ được lấy về.

2. Python fastAPI

Sử dụng thư viện **fastAPI** python chạy ứng dụng qua cổng 8000, server python sẽ nhận request từ node-red và trả về dữ liệu.

2.1 API Sensor random

```
# Server Python sẽ làm gì đó trả về data cho node-red lưu db
@app.get("/demo")
def demo():
    # Bây h là random
    random_number = random.randint(1, 10000)
    # Trả về số ngẫu nhiên dưới dạng JSON
    return {"data": random_number}
```

Khi client gọi tới server python sẽ trả về 1 con số ngẫu nhiên thể hiện cho 1 sensor nào đó! (vì em k có phần cứng ạ!)

2.2 Crawling data website.

Em sử dụng một module do em tạo ra để lấy dữ liệu từ trang web:

<https://www.kqxs.vn>

```
api.php  index.css  index.html  xoso.py API_python 2  test.js  arr_scheduce.json  xoso.py
API_python > xoso.py > play_so_xo > get_so_xo
1  import requests
2  from bs4 import BeautifulSoup
3
4  class play_so_xo:
5      def get_so_xo(target_query):
6          # target_query : 13-05-2024;
7          # Bước 1: Gửi yêu cầu HTTP để lấy HTML của trang web
8          baseurl = 'https://www.kqxs.vn/?yesterday=true&date='
9          url_request = baseurl + target_query
10         response = requests.get(url_request)
11
12         # Kiểm tra xem yêu cầu có thành công hay không
13         if response.status_code == 200:
14             html_content = response.text
15
16             # Bước 2: Sử dụng BeautifulSoup để phân tích HTML
17             soup = BeautifulSoup(html_content, 'html.parser')
18
19             # Bước 3: Tìm thẻ <span> với class "number" và lấy giá trị từ thuộc tính data-value
20             span_tag = soup.find('span', class_='number', attrs={'data-prize': '1'})
21
22             if span_tag:
23                 data_value = span_tag['data-value']
24                 print(f"Giá trị số số {target_query} là: {data_value}")
```

Khi client request đến, server sẽ lấy date gửi lên và cho module thực hiện quá trình tìm dữ liệu và trả về kết quả!

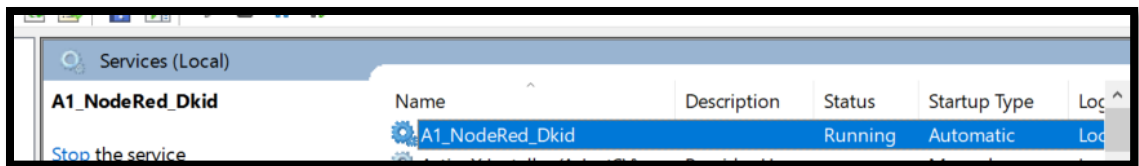
```
@app.get("/soxo")
def xoso(data_query: str):
    print("data_query", data_query)
    kq = play_so_xo.get_so_xo(data_query)
    return {"kq_sx": kq, "date": data_query}
```

3. Node-red

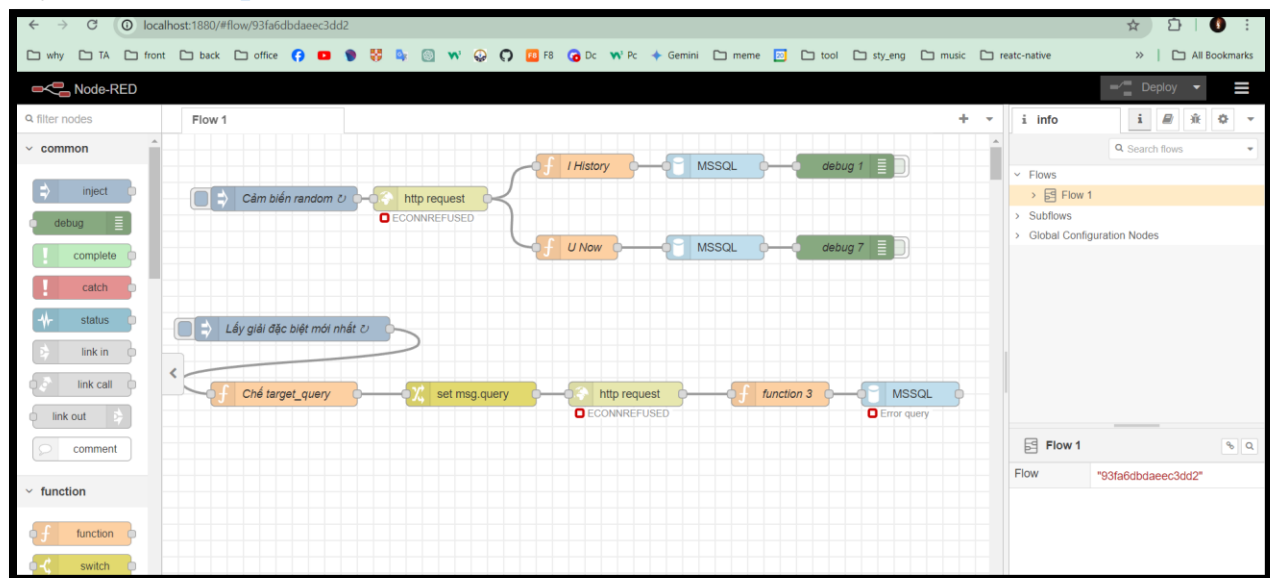
Node-red là một nền tảng chủ yếu cho các dự án IOT nó rất nhỏ gọn. Trong bài tập này node-red được chạy để kết nối giữa python và cơ sở dữ liệu!

Cài đặt node-red:

1. Cài nodejs: khi cài nodejs chúng ta có môi trường chạy javascript và công cụ quản lý gói npm
2. Cài thư viện node-red qua npm: `npm install -g --unsafe-perm node-red`
3. Sử dụng NSSM: chạy node-red như 1 service



Sau khi cài và chạy node-red, nó sẽ chạy mặc định qua cổng 1880, đến trình duyệt và vào <http://localhost:1880>



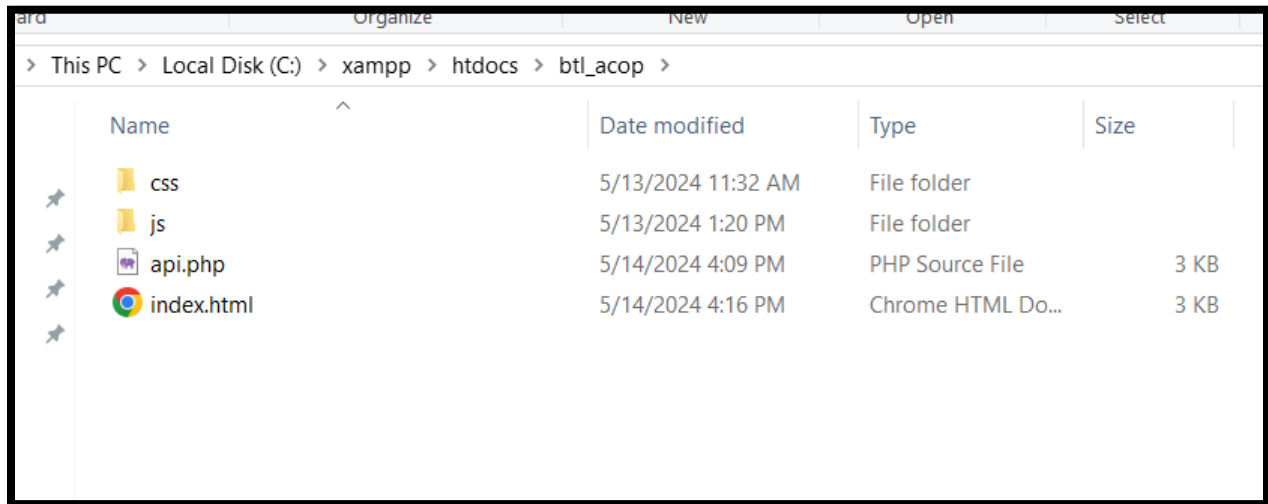
Giao diện lập trình node-red, sử dụng các node:

1. Inject: Xác định vòng lặp tạo xung để thực hiện request tới api
2. Function: chỉnh sửa dữ liệu trước, sau request và dữ liệu lưu database
3. http request: request tới server qua http và nhận về json
4. MSSQL: lưu dữ liệu vào SQL Server

4. Website hiển thị dữ liệu (XAMPP = Apache + PHP)

Xử dụng ứng dụng XAMPP để chạy server có thể dễ dàng chạy máy chủ bằng PHP.

Mã nguồn đơn giản gồm 1 file index.html và các file css để làm đẹp, file js để có AJAX qua API.



File **api.php**:

Kết nối server lấy dữ liệu mà node-red đã lưu trả về kết quả dữ liệu cho js client vẽ giao diện

```
api.php | index.css | index.html | server.py 4 | test.js | arr_sched
btl_acop > api.php
17 // Method GET
18 if ($_SERVER["REQUEST_METHOD"] == "GET") {
19     // check parameter first
20     if(isset($_GET["action"])) {
21         // code GET here!!!!
22         $action = $_GET["action"];
23         if($action=="get_all_ramdom"){
24             // Thực thi truy vấn SQL
25             $stmt = $conn->query("select * from History");
26             // Xử lý kết quả
27             $array_kq = [];
28             while ($row = $stmt->fetch(PDO::FETCH_ASSOC)) {
29                 $array_kq[] = $row; // Thêm dòng hiện tại vào mảng
30             }
31             $json_result = json_encode($array_kq);
32             echo $json_result;
```

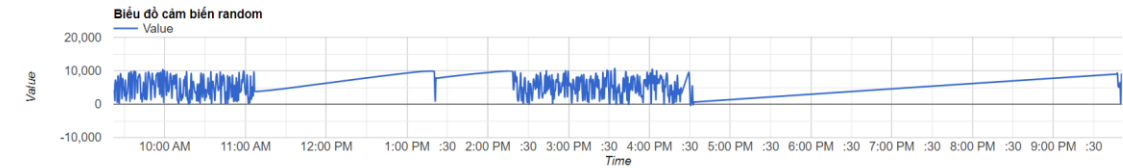
File index.js:

Call về server PHP để nhận dữ liệu tạo giao diện là kết quả hiện tại, biểu đồ lịch sử sensor random và kết quả số xố mới nhất

```
api.php • index.css index.html server.py 4 test.js arr_scheduce.json xoso.py
) bt1_acop > js > index.js > random_all > then() callback > drawChart > chart
38 });
39
40 function randomNow(){
41   fetch(`/bt1_acop/api.php?action=get_random`)
42   .then((response) => {
43     // Xử lý phản hồi từ máy chủ
44     if (!response.ok) {
45       throw new Error("Network response was not ok");
46     }
47     return response.json();
48   })
49   .then((json_data) => {
50     // Chế table
51     console.log("datatanow: ", json_data)
52     let kq = `
53       <table class="table table-bordered table-success">
54         <thead>
55           <tr>
56             <th>STT</th>
57             <th>Sid</th>
58             <th>Name</th>
59             <th>Value</th>
60             <th>Time</th>
61           </tr>
62         </thead>
63         <tbody>
64           <tr>
65             <td>${</td>
66             <td>${json_data[0].sid}</td>
67             <td>${json_data[0].name}</td>
68             <td>${json_data[0].value}</td>
69             <td>${json_data[0].time_update}</td>
70           </tr>
71         </tbody>
72       </table>
73     `;
74   });
```

Kết quả website:

STT	Sid	Name	Value	Time
\$	demo	Cảm biến random	9223	2024-05-14 21:50:35.003



Xem số số

```
[{"date":"14-05-2024","score":"1","result":"16880","desc_":null}]
```