**REST API**

1. **Định nghĩa:**

* REST API cho bạn tìm và trả về 1 list kết quả từ dịch vụ mà bạn request đến.
* API là interface của lập trình ứng dụng. Nó là một tập hợp các rules cho phép ứng dụng trao đổi với nhau. Và developer tạo các API trên server để cho khách hàng giao tiếp với nó.
* REST quyết định API sẽ trông như thế nào. “REST” là viết tắt của Representational State Transfer. REST API cũng là 1 tập hợp các rules nhưng là để cho developer theo khi tạo nên API của họ. Một trong những rules đó là bạn sẽ có thể lấy được 1 phần của data khi link đến 1 URL củ thể nào đó.
* Mỗi URL là một request còn mỗi data trả về là một response.

1. **Định nghĩa một yêu cầu:**

* **Endpoint**
* **Method**
* **Headers**
* **Data hoặc body.**

1. **Endpoint :**

* **Là một yêu cầu do bạn request với cấu trúc :**
  + **Root-endpoint/?**
* **VD :** 
  + **api.twitter.com là root-endpoint củaTwitter API**
  + **api.github.com là root end-point của Github API**
* **Path sẽ quyết định tài nguyên mà bạn request. Như là máy trả lời tự động với các lựa chọn và kết quả khác nhau.**
* **Truy caaoj vào các đường dẫn đó tương đương vào các phần của website. Ví dụ như là để lấy list tất cả các post tag “JavaScript” thì đường link đến là :**

***https://www.smashingmagazine.com/tag/javascript/. https://www.smashingmagazine.com/ is the root-endpoint and /tag/javascript is the path.***

set the request method in cURL by writing -X or --request, followed by the request method.

VD: curl -X POST https://api.github.com/user/repos

You can debug the status of a response with the verbose option (-v or --verbose) or the head option (-I or --head).

1. **200+** means the request has **succeeded**.
2. **300+** means the request is **redirected** to another URL
3. **400+** means an **error that originates from the client** has occurred
4. **500+** means an **error that originates from the server** has occurred
5. To perform a basic authentication with cURL, you can use the -u option, followed by your username and password, like this:

curl -x POST -u "username:password" https://api.github.com/user/repos

1. The data (sometimes called “body” or “message”) contains information you want to be sent to the server. This option is only used with POST, PUT, PATCH or DELETE requests.
2. To send data through cURL, you can use the -d or --data option:

curl -X POST <URL> -d property1=value1

1. To send multiple data fields, you can create multiple -d options:

curl -X POST <URL> -d property1=value1 -d property2=value2

1. If it makes sense, you can break your request into multiple lines \ to make it easier to read:

curl -X POST <URL> \

-d property1=value1 \

-d property2=value2

1. If you know how to spin up a server, you can make an API and test your own data. If you don’t know, but feel courageous enough to try, you can follow [this article](https://zellwk.com/blog/crud-express-mongodb/) to learn to create a server with Node, Express, and MongoDB.

Example:

