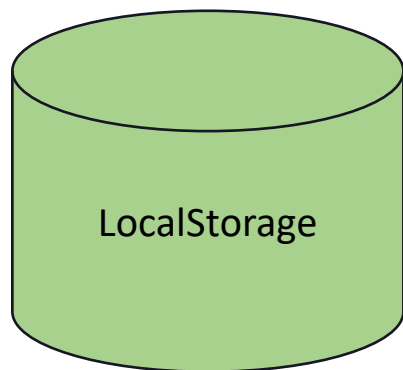


❖ LocalStorage HTML5



Phương thức	Thuộc tính
setItem(name,value)	Dùng để lưu dữ liệu vào storage
getItem(name)	Dùng để lấy dữ liệu ra từ storage
remove(name)	Dùng để xóa dữ liệu trong storage

❑ LocalStorage chứa danh sách các item dưới dạng

```
[  
  {name:value},  
  {name:value},  
]
```

Trong đó:

name: là tên của 1 item localStorage

value: là giá trị của item đó (giá trị trong javascript được hiểu theo là kiểu chuỗi nó chỉ lưu được chuỗi)

❑ Phương thức chuyển đổi kiểu dữ liệu thông dụng trong Javascript

Object Javascript (Đối tượng)

```
{  
  MaSV: '01',  
  HoTen: 'Khai',  
  ....  
  Key: Value  
}
```

Đối tượng có thể truy xuất được thông qua cú pháp **object.Key (object.thuộc tính)**

➔ Lưu ý: Khi dùng hàm **JSON.stringify** để chuyển giá trị từ object sang chuỗi chỉ lưu được object, array, các kiểu dữ liệu cơ sở trừ function là không chuyển được.

JSON.stringify

JSON.parse

JSON (Chuỗi dữ liệu có cấu trúc)

"{MaSV:'01',HoTen:'Khai',...,Key:Value}"

Chuỗi thì không thể truy xuất được đến thuộc tính vì đơn giản nó chỉ là chuỗi string thông thường

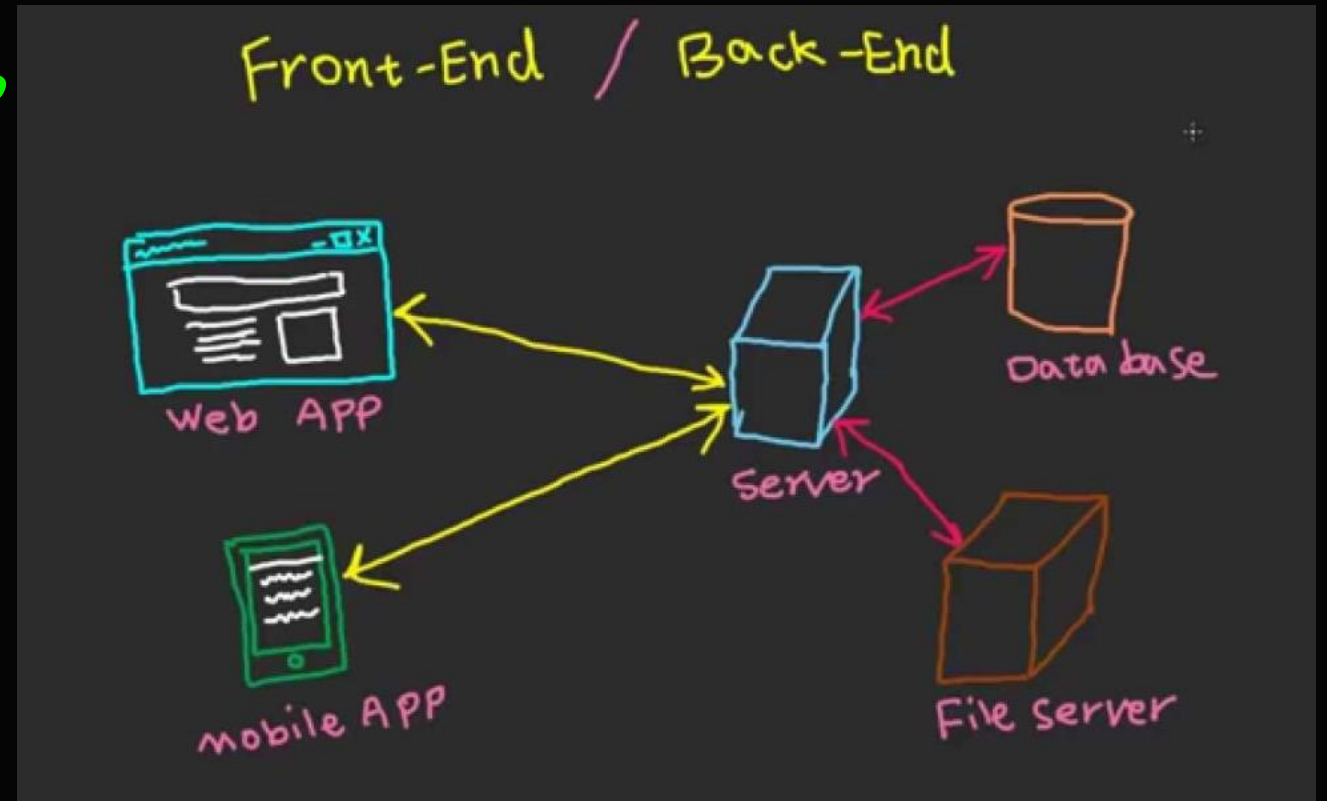
**AXIOS
AJAX**

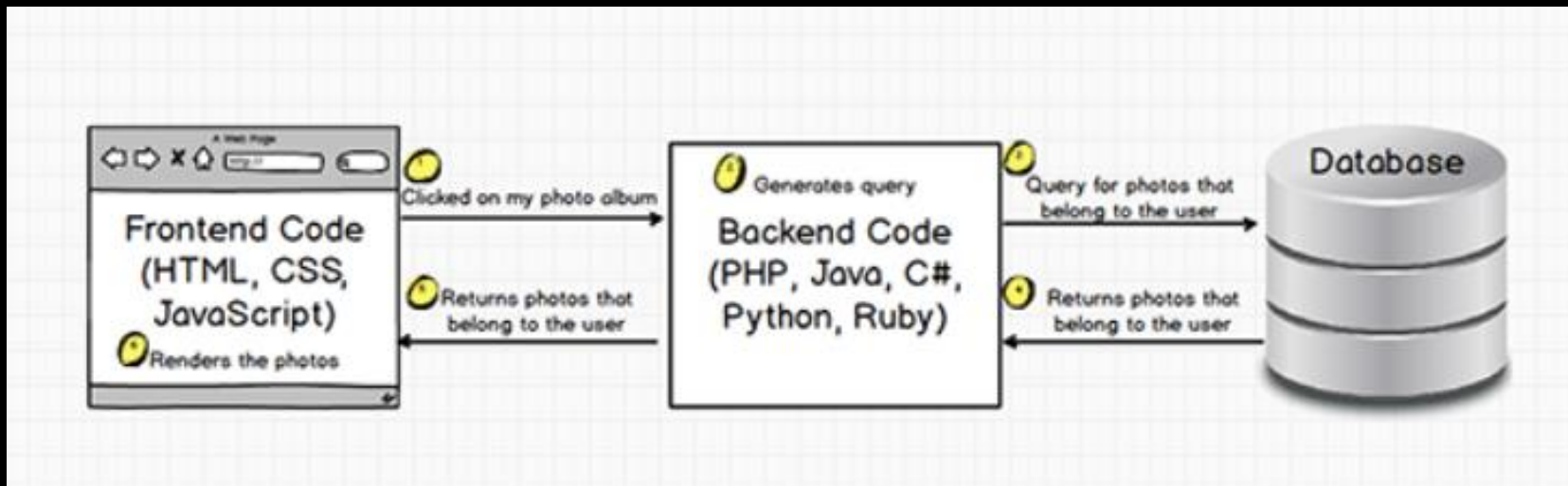
!! **Lập trình FrontEnd là gì?**

!! **Cơ sở dữ liệu là gì?**

!! **Backend làm gì?**

!! **FrontEnd làm gì?**





- Trước đây các lập trình viên web để viết 1 trang web cần thông thạo về cơ sở dữ liệu và ngôn ngữ lập trình để dựng 1 trang web động. Hiểu nôm na là mấy dev fullstack bây giờ 1 mình cân cả thế giới. (Code backend và frontend = 1 ngôn ngữ cách code này người ta gọi là Template Engines).
→ Lúc đó công việc của mấy bạn frontend là cắt photoshop làm html css vài ba hiệu ứng js nhỏ nhoi (lượng không cao như bây giờ vì công việc cũng khá đơn giản ít phải dựng tới logic code)
- Tuy nhiên sự bùng nổ công nghệ thông tin vài năm trở lại đây khi 4.0 bắt đầu được áp dụng trên hầu hết mọi mặt trận ảnh hưởng tới tất cả các ngành nghề → Nhu cầu tăng cao → để kiếm 1 thằng nhân sự best từ đầu tới cuối không hề dễ.
- Lúc đó các nhà phát triển web mới tách web ra làm 2 mảng độc lập là FrontEnd (FE developer) và Backend (BE Developer). Thay vì để làm 1 trang web từ đầu đến cuối thì công việc được chia ra.
 1. Backend đảm nhiệm công việc xử lý dữ liệu thô cho ra json thông qua API.
 2. Frontend đảm nhiệm công việc xây dựng giao diện ứng dụng và kết nối với api từ backend thông qua công nghệ AJAX (Asynchronous JavaScript and XML) → Như vậy FE không cần phải quan tâm đến phần phía dữ liệu là gì xử lý bằng ngôn ngữ nào. Chỉ cần biết html css js (vài thư viện js như react hoặc angular hoặc vue ...) là có thể làm được trang web độc lập.
 3. Còn backend cũng chả cần phải ngồi dàn layout hay css nữa

Cybersoft.edu.vn – Hướng dẫn thực hiện: Trương Tấn Khải

AJAX LÀ GÌ ?

1. Ajax là một kỹ thuật (viết tắt của chữ *AJAX = Asynchronous JavaScript and XML*).
2. Ajax là một công nghệ giúp chúng ta tạo ra những trang web động mà hoàn toàn không cần reload lại trang nên rất mượt và đẹp.
3. Ajax được viết bằng ngôn ngữ javascript nên nó chạy trên client, tức mỗi máy user chạy độc lập mà không hoàn toàn ảnh hưởng đến nhau.
4. Hiện nay có nhiều thư viện javascript như jQuery hay Angular, React, Vue, ... đều hỗ trợ kỹ thuật này.

DÙNG AJAX VỚI THƯ VIỆN AXIOS

✓ ***Một số phương thức ajax thông dụng.***

1. Phương thức get

2. Phương thức post

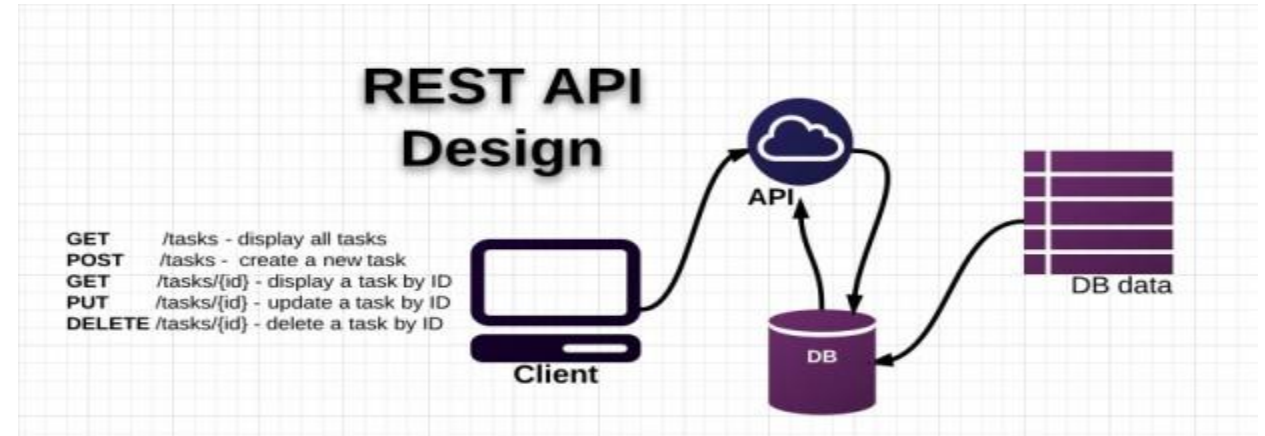
3. Phương thức put

4. Phương thức delete

❑ Lý thuyết về các thứ trên (POST, PUT, GET, DELETE)

(đọc chơi cho biết, đi phỏng vấn có khi hỏi biết đường trả lời 😊😊😊)

- ✓ **Http là gì? Http** là một thành phần của request gọi từ client đến server để tương tác với backend. Tại sao sử dụng POST, PUT, GET, DELETE. Vì đơn giản sử dụng các phương thức theo chuẩn RESTful. Chuẩn RESTful là gì đơn giản chỉ cần hiểu (Vì là phần của backend nên nếu bạn nào muốn tìm hiểu có thể keyword google).
- ✓ **RESTful API** là một tiêu chuẩn dùng trong việc thiết kế API cho các ứng dụng web (thiết kế Web services) để tiện cho việc quản lý các resource. Nó chú trọng vào tài nguyên hệ thống (tệp văn bản, ảnh, âm thanh, video, hoặc dữ liệu động...), bao gồm các trạng thái tài nguyên được định dạng và được truyền tải qua HTTP. (Nguồn: <https://topdev.vn/blog/restful-api-la-gi/>)
- ✓ **API (Application Programming Interface)** là một tập các quy tắc và cơ chế mà theo đó, một ứng dụng hay một thành phần sẽ tương tác với một ứng dụng hay thành phần khác. API có thể trả về dữ liệu mà bạn cần cho ứng dụng của mình ở những kiểu dữ liệu phổ biến như [JSON](#) hay XML.
- ✓ **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.



- ✓ **POST Create:** Dùng để tạo dữ liệu mới (Chậm hơn nhưng bảo mật hơn)
- ✓ **GET Read:** Lấy dữ liệu về (Tốc độ nhanh nhất nhưng bị giới hạn 2048 ký tự ở 1 số browser – cache được)
- ✓ **PUT Update:** Cập nhật dữ liệu
- ✓ **DELETE Delete:** Xóa dữ liệu

- ❑ **Thực hành với axios (Lưu ý chạy project dưới môi trường server)**

Nhúng cđn:

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/axios/0.19.2/axios.min.js"></script>
```

Thực hành với api quản lý sinh viên

<http://svcy.myclass.vn/swagger/ui/index#/SinhVien>

MOVIES	Show/Hide	List Operations	Expand Operations
QuanLyBanHang	Show/Hide	List Operations	Expand Operations
QuanLyMenu	Show/Hide	List Operations	Expand Operations
QuanLyTrungTam	Show/Hide	List Operations	Expand Operations
SinhVien	Show/Hide	List Operations	Expand Operations
GET	/api/SinhVien/LayDanhSachSinhVien		
GET	/api/SinhVien/LayThongTinSinhVien/{id}		
PUT	/api/SinhVien/CapNhatThongTinSinhVien		
POST	/api/SinhVien/ThemSinhVien		
DELETE	/api/SinhVien/XoaSinhVien/{id}		

Cybersoft.edu.vn – Hướng dẫn thực hiện: Trương Tấn Khải