**Restful API**

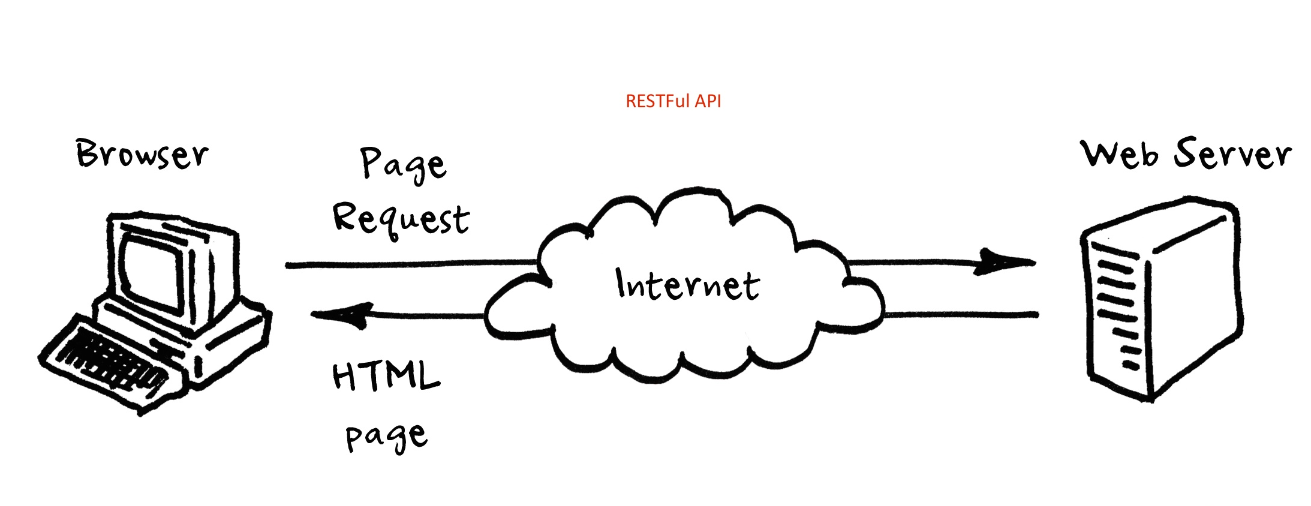
1. **Restful API ?(Rest - Representational State Transfer** **)**

**Restful** API là một tiêu chuẩn dùng trong việc thết kế các thiết kế 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 nhất ngày nay.

Trọng tâm của REST quy định cách sử dụng các HTTP method(GET, POST, PUT, DELETE) và cách định dạng các URL cho ứng dụng web để quản các resource.

Việc quản lý resource của một website bao gồm 4 tác vụ chính:

* Tạo mới một resource (create)
* Lấy thông tin một resource (read)
* Cập nhật một resource (update)
* Xoá một resource (delete).



Có 4 hoạt động chủ yếu ta có thể kể đến khi làm việc với server: lấy dữ liệu ở một định dạng nào đó (JSON), tạo dữ liệu mới, cập nhật dữ liệu, xóa dữ liệu. REST hoạt động chủ yếu dựa vào giao thức HTTP. Các hoạt động cơ bản nêu trên sẽ sử dụng những phương thức HTTP riêng.

* GET: lấy dữ liệu
* POST: tạo mới
* PUT: cập nhật (thay đổi)
* DELETE: Xóa dữ liệu

Những phương thức hay hoạt động này thường được gọi là CRUD tương ứng với Create, Read, Update, Delete – Tạo, Đọc, Sửa, Xóa. Mỗi phương thức trên phải được API call thông quađể gửi chỉ thị cho server phải làm gì. Đại đa số web API chỉ cho phép GETrequest lấy dữ liệu khỏi một externer server.

1. **SOAP** (Simple Object Access Protocol) và **REST**(Representational State Transfer)

**SOAP** là một giao thức truy cập Webservice được sử dụng trong một thời gian và hưởng tất cả các lợi ích của việc sử dụng lâu dai. Được phát triển bởi Micrisoft, **SOAP** không thực sự đơn giản như từ viết tắt của nó. Trong khi đó, **REST** là một chiến binh mới, tìm kiếm để sửa các vấn đề mà **SOAP** gặp phải và cung cấp các phương thức thực sự đơn giản để truy cập Webservice. Tuy nhiên, đôi khi **SOAP** thực sự dễ sử dụng hơn, đôi khi **REST** có vấn đề của riêng nó. Cả hai kỹ thuật đều có vấn đề cần xem xét khi quyết định sử dụng phương thức nào.

**SOAP** chắc chắn là lựa chọn nặng ký để truy cập Web service. Nó cung cấp những ưu điểm sau đây khi so sánh với REST:

* Ngôn ngữ, nền tảng và phương tiện độc lập (REST yêu cầu sử dụng HTTP)
* Hoạt động tốt trong môi trường doanh nghiệp (REST giao tiếp thẳng trực tiếp)
* Tiêu chuẩn hoá
* Cung cấp khả năng mở rộng đáng kể trước khi xây dựng dưới dạng các tiêu chuẩn WS \*
* Tích hợp xử lý lỗi
* Tự động hóa khi sử dụng với một số sản phẩm ngôn ngữ

Bên cạnh đó,***REST***dễ sử dụng hơn và linh hoạt hơn. Nó có những lợithế sau khi so sánh với ***SOAP:***

* Không có tools đắt tiền nào yêu cầu tương tác với Web service
* Smaller learning curve
* Hiệu quả (SOAP sử dụng XML cho tất cả các truyền tin, REST có thể sử dụng định dạng truyền tin ngắn gọn hơn)
* Nhanh (không yêu cầu xử lý rộng rãi) Gần gũi hơn với các công nghệ Web khác trong triết lý design.
* Trên đây là một vài khái niệm cơ bản về SOAP và REST cũng như các đặc điểm của từng phương thức. Hi vọng chúng sẽ giúp ích cho các bạn để lựa chọn phương thức phù hợp nhất với ứng dụng của mình.