

[KHƠI MỎ VÀNG TRAO GIÁ TRỊ](#)[TRANG CHỦ](#)[SẢN PHẨM](#)[TÍNH NĂNG](#)[HỌC MARKETING](#)YOU ARE HERE: [HOME](#) » [BLOG](#) » [TẤT TẦN TẬT VỀ RESTFUL APIS VÀ CÔNG CỤ POSTMAN](#)

Tất tần tât về RESTful APIs và công cụ Postman

[KHƠI MỎ VÀNG TRAO GIÁ TRỊ](#)

Share



Share



Tweet



Share



Share



Pin



api

Hiện **nguyên lý REST và cấu trúc dữ liệu RESTful** được biết đến rất rộng rãi trong giới lập trình, nhưng vẫn có nhiều người hay nhầm lẫn. Bản thân REST không phải là một loại công nghệ, mà là **phương thức tạo API với nguyên lý tổ chức nhất định**. Những nguyên lý này nhằm hướng dẫn lập trình viên tạo môi trường xử lý API request được toàn diện hơn. Chúng ta cùng nhau tìm hiểu về RESTful ở góc nhìn toàn cảnh nhất, đồng thời làm rõ khái niệm REST APIs nhé.

REST For Web Developers

REST là viết tắt của **Representational State Transfer**. Giải thích đơn giản, REST là một loại **hướng dẫn và dạng cấu trúc dùng cho việc chuyển đổi dữ liệu**. Thông thường, REST hay được dùng cho ứng dụng web, nhưng cũng có thể làm việc được với dữ liệu phần mềm.

API là viết tắt của **Application Programming Interface**, phương thức **kết nối với các thư**



KHOI MỎ VÀNG TRAO GIÁ TRỊ

Share



Share



Tweet



Share



Share



Pin

Nhìn chung, RESTful API là những API đi theo cấu trúc REST.

RESTful Là Gì ?

REST là viết tắt của cụm từ **Representational State Transfer** (đôi khi còn được viết là ReST), là một kiểu kiến trúc được sử dụng trong việc giao tiếp giữa các máy tính (máy tính cá nhân và máy chủ của trang web) trong việc quản lý các tài nguyên trên internet. REST được sử dụng rất nhiều trong việc phát triển các ứng dụng Web Services sử dụng giao thức HTTP trong giao tiếp thông qua mạng internet. Các ứng dụng sử dụng kiến trúc REST này thì sẽ được gọi là ứng dụng phát triển theo kiểu RESTful.

Trong từ **RESTful**, thì từ *ful* (đọc là phù, phồ, phò hay phùn đều được) chính là *suffix* trong tiếng Anh, giống như từ *help* có nghĩa là giúp đỡ thì từ *helpful* là rất hữu ích.

Cấu trúc REST

Thật khó giải thích sao cho cụ thể. Tuy nhiên, vẫn có một số **quy luật bất biến**, như:

- » **Sự nhất quán** trong cả API
- » **Tồn tại không trạng thái** (ví dụ, không có server-side session)
- » Sử dụng **HTTP status code** khi cần thiết
- » Sử dụng **URL endpoint** với logical hierarchy
- » **Versioning** trong URL chứ không phải trong HTTP header

Hơn nữa, bạn không nhất thiết phải tuân theo những quy luật trên không sai một chữ (di



Share



Share



Tweet



Share

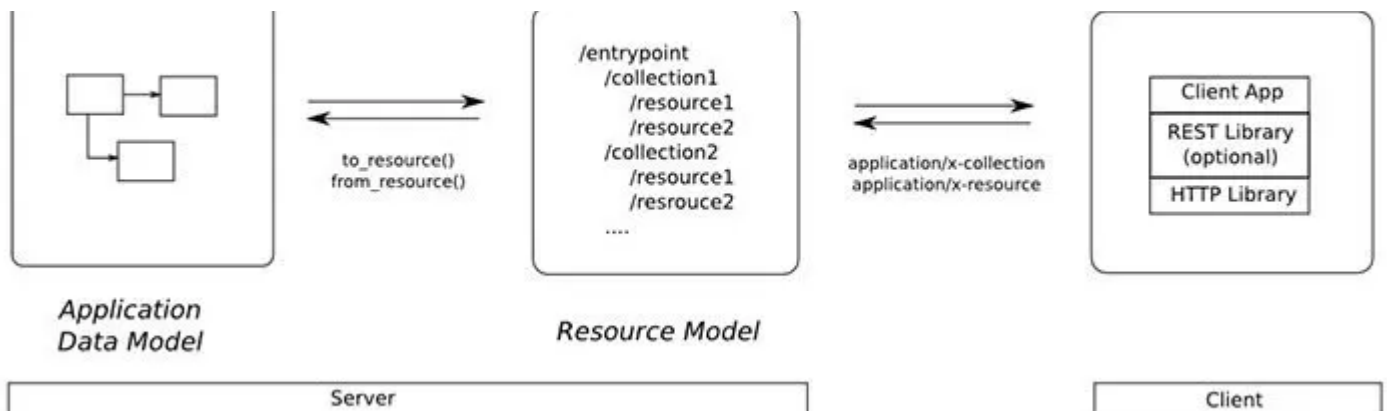


Share



Pin

KHOI MỎ VÀNG TRAO GIÁ TRỊ



Cần chú ý với RESTful APIs

Nhìn chung, có **bốn lệnh** dùng để **truy cập RESTful API**:

1. GET để truy vấn object

2. POST để tạo object mới
3. PUT để sửa đổi hoặc thay thế một object
4. DELETE để loại bỏ một object

Mỗi phương thức trên phải **được API call thông qua** để gửi chỉ thị cho server phải làm gì.

Đại đa số web API **chỉ cho phép GET request** lấy dữ liệu khỏi một external server.

Authentication không bắt buộc, nhưng nên có khi ta cho phép các lệnh khá “nguy hiểm” như PUT hay DELETE.

Tuy nhiên, rất ít thấy RESTful API nào cho phép các lệnh này. Ví dụ như <http://pokeapi.co>, Pokemon API database miễn phí, với lượng rate limit công khai khá (rate limit: người dùng bị giới hạn số kiểu API request thực hiện được), nhưng chỉ cho phép phương thức **GET** để truy cập tài nguyên. Trong “dân gian” ta hay gọi kiểu giới hạn này là **consumption-only API**.



Share



Share



Tweet



Share



Share



Pin

KHOI MỎ VÀNG TRAO GIÁ TRỊ

rất quan trọng, chỉ nên dùng cho API request.

Một số URL có thể bao gồm đường dẫn tiền tố như **/v2/** cho phiên bản 2 cập nhật từ API trước đó; hay thấy ở những lập trình viên muốn giữ 1.x API, nhưng vẫn muốn cung cấp cấu trúc mới nhất.

Nên nhớ, **return data ở endpoint sẽ thay đổi** mạnh mẽ dựa vào **phương thức HTTP**. Ví dụ, **GET** trả nội dung, còn **POST** tạo nội dung mới. Request có thể chỉ đến cùng một endpoint, nhưng kết quả có thể rất khác.

Để hiểu rõ khái niệm này, các bạn có thể lên tìm nhiều ví dụ online khác. Ngoài [Pokeapi](http://pokeapi.co), ta còn có:

- » [Reddit API](#)
- » [GitHub API](#)
- » [Flickr API](#)
- » [Pinterest API](#)

Tự build API

Quá trình xây dựng một API riêng không dễ dàng, nhưng cũng không quá phức tạp như nhiều người nghĩ.

Mỗi API phải **kết nối đến server** để trả dữ liệu. Bạn không những phải viết code để làm điều đó, mà còn phải format return data nữa. Một số yêu cầu khác có thể gồm **authentication** và **rate limiting**.

Hãy điểm qua một số nguyên lý cơ bản của cấu trúc API.



Share



Share



Tweet



Share



Share



Pin

KHOI MỎ VÀNG TRAO GIẢI TRỊ

ương để bạn học hỏi, vì qua đó bạn có thể tranh luận với những tập thể khác và hiểu được nguyên lý đằng sau lý thuyết đó.

Kết nối với các API khác trước cũng là một cách học hỏi hay. Hãy tìm hiểu về các điểm cơ bản nhất của kết nối client-side, và từ đó bạn có thể chuyển sang phát triển API, tạo API riêng từ con số không.

Bạn có thể tham khảo một số tài liệu dưới nếu muốn đi theo con đường API này.

Sách

- » [REST API Design Rulebook](#)
- » [RESTful Web APIs](#)
- » [RESTful Web Services Cookbook](#)

» [Undisturbed REST: A Guide to Designing the Perfect API](#)

Bài viết

» [A Beginner's Guide to HTTP and REST](#)

» [Creating a RESTful API](#)

» [RESTful Resource Naming Guide](#)

» [Creating A REST API using the MEAN Stack](#)

Postman là gì ?



KHOI MỎ VÀNG TRAO GIÁ TRỊ

Share

 Share

 Tweet

 Share

 Share

 Pin



KHƠI MỎ VÀNG TRAO GIẢI TRỊ

Share

 Share Tweet Share Share Pin

postman

Postman là một App Extensions, cho phép làm việc với các API, nhất là REST, giúp ích rất nhiều cho việc testing. Hỗ trợ tất cả các phương thức HTTP (GET, POST, PUT, DELETE, OPTIONS, HEAD ...) Postman cho phép lưu lại các lần sử dụng. Sử dụng cho cá nhân hoặc team lớn.

Cài đặt

Truy cập trang chủ <https://www.getpostman.com/> hoặc từ Chrome Web Store.

Cách sử dụng

Sau khi cài đặt, các bạn bấm vào app, chọn Postman, giao diện đăng nhập sẽ hiện ra. Bạn có thể bấm **“Skip This”** để bỏ qua phần đăng nhập, tuy nhiên mình khuyến khích **nên tạo account** và đăng nhập vì Postman có thể đồng bộ dữ liệu của bạn trên nhiều máy, khá tiện lợi.

Bạn có thể sử dụng phương thức đăng nhập bằng Google.

Sau khi bạn đăng nhập thành công giao diện chính sẽ hiện ra



KHOI MỎ VÀNG TRAO GIÁ TRỊ

Share



Share



Tweet



Share



Share



Pin



KHƠI MỞ VÀNG TRAO GIẢI TRỊ

Share



Share



Tweet



Share



Share



Pin

postman-preview

Một số khái niệm

Request: Để sử dụng RestAPI, ta sẽ gửi một HTTP Request tới endpoint của API đó. Ví dụ như để lấy thông tin người dùng github có username là conanak99, ta sẽ gửi HTTP GET

Request tới endpoint: <https://api.github.com/users/conanak99>.

Mỗi request này chứa những thông tin như URL, method (POST, GET), body (trường hợp dùng POST hay PUT), HTTP header.

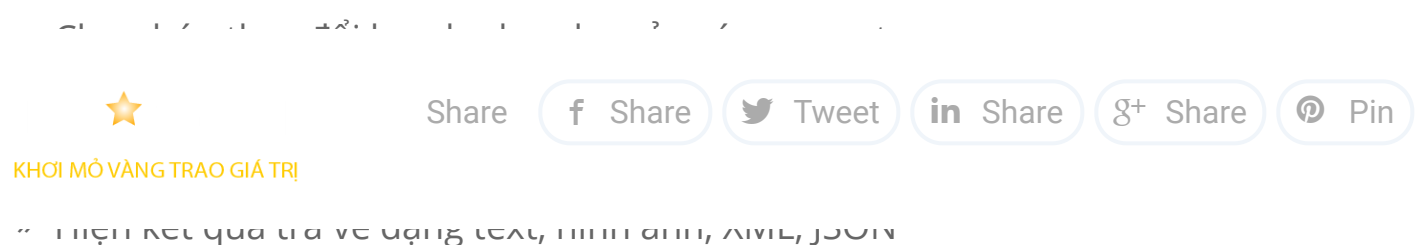
Collection: Collection là một tập hợp của nhiều request. Những request liên quan tới nhau thì ta gom lại vào chung 1 collection để dễ quản lý hơn.

Enviroment và variable: Trong dự án thực tế, người ta thường dùng deploy RestAPI trên nhiều môi trường (local, test, [production](#)). Postman hỗ trợ cài đặt các biến môi trường (ur gốc, API key, ...), thuận tiện hơn khi cần test trên nhiều môi trường

Các chức năng cơ bản của postman

Các chức năng hay được sử dụng của Postman bao gồm:

- » Cho phép gửi HTTP Request với các method GET, POST, PUT, DELETE.



- » Hỗ trợ authorization (Oauth1, 2)

Để gửi 1 request, ta chỉ cần dán URL vào khung text, chọn method, điền header, body, sau đó bấm Send là xong.

Kết

Qua bài hướng dẫn này, bạn đã có cái nhìn tổng quát về của Postman, đồng thời biết cách sử dụng những chức năng chính của nó. Postman rất hữu dụng với các Web Developer khi bạn cần test cách truyền params hoặc xem kết quả trả về từ một RestAPI nào đó.

Ngoài ra, Postman còn có các chức năng khá hay như: Viết và chạy test case cho API, export collection để chia sẻ với người khác, các bạn có thể tự mày mò tìm hiểu thêm.

Chúc các bạn thành công. Nếu thấy bài viết này có ích đừng quên Like Share để ủng hộ [Movan.vn](#) nhé

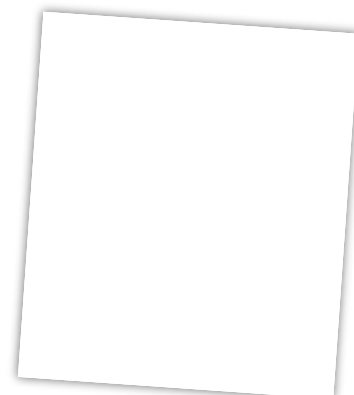
Nếu bạn có câu hỏi hay thắc mắc gì thì cứ comment trong bài nhé.

1.8 (35%) 32 vote[s]

Tác giả Nguyễn Duy Nhật

Học tại Học viện kĩ thuật mật mã Quê quán: Hà Tĩnh

follow me on:   



KHOI MỎ VÀNG TRAO GIÁ TRỊ

Share

 Share

 Tweet

 Share

 Share

 Pin

18 Th5, 2019

CRM là gì-Định nghĩa, tầm quan trọng của việc sử dụng CRM?

16 Th5, 2019

Omnichannel – Nhân tố sống còn của ngành bán lẻ

16 Th5, 2019

KOL's là gì? và tại sao bạn nên quan tâm?

[← Previous Post](#)[Next Post →](#)

Hacker đấu tranh cho tự do internet tự
vẫn, giới công nghệ nổi sóng

Googlebot là gì? Làm sao để chiến thắng
Googlebot

Popular posts

Khách Hàng Tiềm Năng Giúp
Tăng Trưởng Doanh Số Bất
Ngờ

▲ Câu chuyện khách hàng Th8 02, 2017

Phễu Marketing & Chuyển Hóa
Khách Hàng Trên Facebook

▲ Học Marketing ▲ Sale Funnels
Th7 18, 2017

Phễu Bán Hàng
Giúp Tăng Doai

▲ Học Marketing ▲ S
Th7 19, 2017

Copyright text © 2017 by MOVAN JSC. Chính sách và điều khoản [Xem tại đây](#)