RESTFul API's

Karie Funk

What are RESTful APIs

RESTful API’s are an application program interface (API) that uses HTTP requests to GET, PUT, POST and DELETE data.

It is based on REST technology. Which is an architectural style and approach to communications often used in web services development.

History of RESTful API's

In 2000, Roy Fielding and his colleagues had one objective: create a standard so that any server could talk to any other server in the world.

RESTful API design was defined by Dr. Roy Fielding in his 2000 doctorate dissertation.

The first ones to be interested in the phenomenon were the e-commerce giants, Ebay followed by Amazon.

History of RESTful API's cont.

Before RESTful API's SOAP—Simple Object Access Protocol—was used to integrate APIs.

Flickr launched their own REST API in August of 2004. They quickly became an extremely popular platform for images, which bloggers were finally able to easily embed on their sites and social media feeds.

Since 2006, REST APIs has drastically improved and has become extremely popular for software developers.

How are RESTful API's used?

RESTful API's are used with HTTP protocols and a client and a server.

A REST API works by the user searching for something and getting a list of results back from the service you're requesting from.

When you link to a specific URL, you can get a piece of data called a resource. Each URL is called a request while the data sent back to you is called a response.

Website communications

REST relies on HTTP protocols. These protocols consist of GET, PUSH, DELETE, and POST.

What the user is trying to do will determine the HTTP protocol used.

For example, when a user is trying to load a webpage, then a GET request is used. When a user is typing into a form of some sort a POST request is being used.

Advantages of RESTful APIs.

- Flexibility
- Used by sites such as Google, Facebook, and Amazon
- Variety of data formats
- Separation of client and server
- Stateless
- Easily testable

Disadvantages of RESTful APIs.

- Poor implementation
- Confusing to new developers
- Can be difficult to maintain
- Can be expensive
- Doesn’t have a built-in messaging system
- Not the best at confidential data being passed between the client and the server

The differences between a URI and URL

A URI stands for Uniform Resource Identifier

A URL stands for Uniform Resource Locator

A URI can identify a web resource by location, name, or both. Examples of this include images, videos, a CSS file, etc.

For example, say you have this web address: https://foobar/path/to/image.jpg

The location of a URI means the ‘http://’ and the name refers to ‘foobar/path/to/image.jpg’

The differences between a URI and URL cont.

A URL defines where the resource can be obtained.

For example, a URL would correspond to a person's street address.

URL always includes a network protocol e.g. HTTP, HTTPS, FTP etc to retrieve a resource from its location.

While URI, in case of URN just uniquely identifies the resource e.g. ISBN numbers which are a good example of URN is used to identify any book uniquely.

References

What is a restful api? | mulesoft. (n.d.). Retrieved from https://www.mulesoft.com/resources/api/restful-api

Understanding and using rest apis Smashing Magazine. (n.d.). Retrieved from https://www.smashingmagazine.com/2018/01/understanding-using-rest-api/

What is restful api? - definition from whatis.com. (n.d.). Retrieved from https://searchapparchitecture.techtarget.com/definition/RESTful-API

Team, G. (n.d.). Advantages and disadvantages of restapi over soap | geekboots. Retrieved from https://www.geekboots.com/story/advantages-and-disadvantages-of-restapi-over-soap

References cont.

History of rest apis - mobapi. (n.d.). Retrieved from https://www.mobapi.com/history-of-rest-apis/

The history of rest apis. (n.d.). Retrieved from https://blog.readme.io/the-history-of-rest-apis/

Comparing uri vs url - keycdn support. (n.d.). Retrieved from https://www.keycdn.com/support/comparing-uri-vs-url