Difference between a POST request and PUT request

Jay Chow

Software Engineer, Cisco eCommerce, IT

Cisco, San Jose

4th June 2020

Agenda

1 POST

2 PUT

3 POST vs PUT: An example

4 Reference

POST request

- The POST method is used to request that the origin server accept the entity enclosed in the request as a new subordinate of the resource identified by the Request-URI in the Request-Line. It essentially means that POST request-URI should be of a collection URI i.e POST /questions.
- POST is NOT idempotent. So if you retry the request N times, you will end up having N resources with N different URIs created on server.
- Use POST when you want to add a child resource under resources collection.
- Use POST when you want to add a child resource under resources collection.
- Responses to this method are not cacheable, unless the response includes appropriate Cache-Control or Expires header fields. However, the 303 (See Other) response can be used to direct the user agent to retrieve a cacheable resource.
- Always use POST for CREATE operations.

PUT request

- RFC-2616 clearly mention that PUT method requests for the enclosed entity be stored under the supplied Request-URI. If the Request-URI refers to an already existing resource – an update operation will happen, otherwise create operation should happen if Request-URI is a valid resource URI i.e PUT /questions/{question-id}.
- PUT method is idempotent. So if you send retry a request multiple times, that should be equivalent to single request modification
- Use PUT when you want to modify a singular resource which is already a part of resources collection. PUT replaces the resource in its entirety. Use PATCH if request updates part of the resource
- Though PUT is idempotent, the responses is not cached
- Generally, in practice, always use PUT for UPDATE operations

POST vs PUT: An example

- Let’s say we are designing a network application:

GET /device-management/devices : Get all devices

***POST** /device-management/devices : Create a new device*

GET /device-management/devices/{id} : Get the device information identified by "id"

***PUT** /device-management/devices/{id} : Update the device information identified by "id"*

DELETE /device-management/devices/{id} : Delete device by "id"

Reference

https://restfulapi.net/rest-put-vs-post/