**Http method best practices**

1. One software module sends a request to a second software module and waits for a response.
2. The first software module performs the role of the client.
3. The second the role of the server.
4. Browser is the primary http client responsible for load the web application.

http client

* Browser level client
* Application level client

http client is an application library used in client side application to generate request and receive response.

http client libraries varies from platform to platform.

POSTMAN http client

Postman is an http client application used to test request response communication. Postman is widely used for API testing and generating documentation.

Quickly and easily send REST , SOAP and GRAPHQL request directly within postman.

Generate and publish beautiful and machine readable API documentation.

Http request

http request is the first step to initiate web request/response communication. Every request is a combination of request header, body and request URL.

Http request segments

Request area: Standard data type.

Body: Simple string JSON, Download, redirect, XML.

Header: Key pair value

URL parameter: String

Compare Get vs Post

* Back button/reload is harmless for Get but data will be re submitted in Post.
* In Get method can be bookmarked but in Post method bookmarked is not possible.
* Cached can be possible in Get but in Post cash cannot be possible.
* History parameter remain in browser history in Get but in Post history parameters are not saved in browser history.
* In Get there is restriction in data length but in post there are no restrictions.
* In Get only ASCII characters are allowed but in Post there are no restrictions, binary is also allowed in Post.
* Get is less secure compared to Post.
* In Get data is visible to everyone in URL but in Post data is not displayed in the URL.

**Http request throttling**

Throttle request refers to a process in which a user is allowed to hit the application maximum time in per second or per minute. Throttling is also known as request rate limiting.

* Essential component of internet security as dos attacks can tank a server with unlimited request.
* Rate limiting also helps make your API scalable by avoid unexpected spikes in traffic , causing severe lag time.

**HTTP response**

* http response is the final step of request-response communication. Every response is combination of response header, body and cookies.

**http response segments**

* Response Area: Standard data type
* Body: Simple string, JSON, download, XML, Redirect
* Header: Key Pair Value
* Cookies: Key Pair Value

**HTTP response status messages**

* 200: ok
* 201: created
* 202: accepted
* 203: non authoritative information
* 204: no content
* 205: reset content
* 206: partial content
* 400: bad request
* 401: unauthorized
* 403: forbidden
* 404: not found
* 405: method not allowed
* 408: request time out
* 500: internal server error
* 502: bad gateway
* 503: service unavailable