**Métodos HTTP**

O protocolo HTTP define um conjunto de **métodos de requisição** responsáveis por indicar a ação a ser executada para um dado recurso. Cada um deles implementa uma semântica diferente, mas alguns recursos são compartilhados por um grupo deles, como por exemplo, qualquer método de requisição pode ser do tipo safe, idempotent ou cacheable.

Os métodos HTTP são: GET, HEAD, POST, PUT, DELETE, CONNECT, OPTIONS, TRACE, PATCH.

**GET:** Através dessa requisição nós pedimos a representação de um recurso: que pode ser um arquivo html, xml, json, etc.

Se desejamos visualizar a home de um site enviamos a seguinte requisição:

GET / HTTP/1.1

Host: www.google.com.br

...

**HEAD:** solicita os cabeçalhos retornados de um recurso específico que foi requisitado por um método HTTP [GET](https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Methods/GET). Tal solicitação pode ser feita antes de baixar um grande recurso para economizar largura de banda, por exemplo.

**POST:** **É utilizado para enviar dados para o servidor.** Quando usamos POST, os dados vão no corpo da requisição e não na URI.

POST /cadastro.php HTTP/1.1

Host: www.devmedia.com.br

nome=Marlon&sobrenome=Paulo...

**PUT:** atualiza as propriedades literais e as propriedades de recurso local, além de excluir quaisquer propriedades de recurso local que não estão incluídas na solicitação.

O método a seguir atualiza a propriedade literal, taskname:

PUT http://yourserver/oslc/so/WorkTask/123

{

"dcterms:taskname": "Check-out Leaking – Modified for Test"

}

Se a solicitação for processada com sucesso, o cliente OSLC receberá a seguinte resposta:

204 No Content

ETag: 1376596202470

**DELETE:** remove o recurso especificado. Se desejarmos excluir um arquivo enviamos a seguinte requisição:

DELETE /arquivo.html HTTP/1.1

**CONNECT:** Converte a requisição de conexão para um túnel TCP/IP transparente, facilitar a comunicação criptografada com SSL (HTTPS) através de um proxy HTTP não criptografado.geralmente para

CONNECT server.exemplo.com:80 HTTP/1.1

Host: server.exemplo.com:80

Proxy-Authorization: basic aGVsbG86d29ybGQ=

**OPTIONS:** Retorna os métodos HTTP suportados pelo servidor para a URL especificada.

Para descobrir quais tipos de requisição um servidor suporta, é possível utilizar o curl e enviar uma requisição OPTIONS:

curl -X OPTIONS http://example.org -i

A resposta terá um cabeçalho Allow com os métodos permitidos:

HTTP/1.1 200 OK

Allow: OPTIONS, GET, HEAD, POST

Cache-Control: max-age=604800

Date: Thu, 13 Oct 2016 11:45:00 GMT

Expires: Thu, 20 Oct 2016 11:45:00 GMT

Server: EOS (lax004/2813)

x-ec-custom-error: 1

Content-Length: 0

**TRACE:** Devolve a mesma requisição que for enviada veja se houve mudança e/ou adições feitas por servidores intermediários.

**PATCH:** Serve para atualizar **partes** de um recurso, e não o recurso todo.

PATCH /file.txt HTTP/1.1

Host: www.example.com

Content-Type: application/example

If-Match: "e0023aa4e"

Content-Length: 100

[descrição das mudanças]

Uma resposta sucedida é indicada pelo status de resposta 204, visto que a resposta não carrega um corpo de mensagem.

HTTP/1.1 204 No Content

Content-Location: /arquivo.txt

ETag: "e0023aa4f"

**WSDL**

É um a descrição em formato XML de um Web Service que utilizará SOAP / RPC como protocolo. É o acrônimo de Web Services Description Language (Linguagem de Descrição de Serviços Web).

Por meio de um **WSDL** você informa ao cliente como cada serviço em um end-point deve ser invocado: quais os parâmetros e tipo de dados de cada parâmetro é esperado, e qual o tipo de dado do retorno será enviado como resposta.

Além de descrever cada serviço (que pode ser comparado analogamente à um método a ser executado no programa servidor), também descreve como podem ser encontrados.

**Referência bibliográfica**

Métodos de requisição HTTP. Disponível em <https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Methods>. Acesso em 29 nov.2022

FERREIRA, Gabs. Os métodos HTTP: quais são e pra que servem. Disponível em <http://gabsferreira.com/os-metodos-http-e-a-diferenca-entre-eles/>. Acesso em 29 nov.2022

HTTP: GET e POST. Disponível em [https://www.devmedia.com.br/http-get-e-post/41221. Acesso em 29 nov.2022](https://www.devmedia.com.br/http-get-e-post/41221.%20Acesso%20em%2029%20nov.2022)

Método HTTP PUT. Disponível em <https://www.ibm.com/docs/pt-br/tap/3.5.0?topic=resources-http-put-method>. Acesso em 29 nov.2022

SILVA, Alex José. Introdução ao WSDL. Disponível em <https://medium.com/@alexjosesilva/introdu%C3%A7%C3%A3o-ao-wsdl-abece3a8bab5>. Acesso em 29 nov.2022