# HTTP protocol da Internet

HTTP (HyperText Transfer Protocol) é um protocolo de rede usaso na transferência de informações na internet. A sua utilização mais comum é entre os browsers e os servidores web que alojam os websites que visitamos.

A inserção de um endereço na barra de url do browser ou quando se clica num link, o é enviado um pedido HTTP para o servidor web. Este pedido é um pedido para que o servidor envie os dados do site de volta para o seu browser, para que você possa visualizá-lo.

Existem diferentes tipos de pedidos HTTP que podem ser enviados pelos navegadores, designados tipicamente de verbs ou actions, o GET, POST, PUT, DELETE, entre outros. Cada tipo de pedido indica ao servidor a ação que o navegador deseja que ele execute. Por exemplo, um pedido GET é um pedido para que o servidor envie os dados de uma página para o navegador, enquanto um pedido POST associado ao envio de dados para o servidor, por exemplo o envio dos dados de um formulário

Referencias para estudo:

<http://www.steves-internet-guide.com/http-basics/>

<https://kinsta.com/knowledgebase/javascript-http-request/>

# WebAPI

Uma Web API é uma interface de programação que permite que uma aplicação web ou servidor web interaja com aplicações ou serviços externos. Ela permite a comunicação entre diferentes sistemas e dispositivos através da internet usando protocolos web padrão, como HTTP.

As Web APIs são frequentemente usadas para permitir que desenvolvedores terceirizados criem aplicações que interajam com um determinado serviço web ou site. Por exemplo, um site de redes sociais pode fornecer uma Web API que permite aos desenvolvedores criar aplicativos que possam acessar e modificar as informações de perfil de um usuário ou publicar conteúdo no site em nome do usuário.

As Web APIs podem ser implementadas de várias maneiras e podem retornar dados em vários formatos, como XML, JSON ou HTML. Elas são frequentemente baseadas na arquitetura REST (Representational State Transfer), que é um conjunto de princípios para projetar serviços web que são leves, mantíveis e escaláveis.

Referencias para estudo:

<https://medium.com/swlh/api-for-dummies-232a5a48f950>

<https://www.youtube.com/watch?v=OVvTv9Hy91Q>

# Interagir com WebAPi’s através de JavaScript

A função **fetch** é uma função do JavaScript que permite que você envie pedidos HTTP para obter ou enviar dados de um servidor.

A sintaxe básica da função **fetch** é:

fetch(url, options)

Onde **url** é o endereço do servidor que você deseja acessar e **options** é um objeto com as opções para o pedido. Algumas opções comuns incluem:

* **method**: O tipo de pedido HTTP a ser enviado (por exemplo, "GET", "POST", etc.).
* **headers**: Um objeto com os cabeçalhos que devem ser incluídos no pedido.
* **body**: O corpo do pedido, que pode conter os dados a serem enviados para o servidor.

A função **fetch** retorna uma promessa que pode ser usada para obter a resposta do servidor quando ela estiver disponível. Por exemplo:

fetch(url) .then(response => response.json()) .then(data => console.log(data)) .catch(error => console.error(error))

Neste exemplo, o pedido é enviado para o endereço especificado em **url** e, quando a resposta do servidor estiver disponível, o corpo da resposta é convertido em JSON e exibido no console. Se ocorrer algum erro, ele é capturado no bloco **catch** e exibido no console.

Referencias para estudo:

<https://blog.webdevsimplified.com/2022-01/js-fetch-api/>

<https://www.youtube.com/watch?v=cuEtnrL9-H0>

<https://www.youtube.com/watch?v=Oive66jrwBs>