

HTTP

HyperText Transfer Protocol (HTTP) = é um protocolo, um conjunto de regras, que estabelece as regras de comunicação entre o navegador e o servidor para que ambos entendam o que está sendo pedido pelo usuário.

Documentação do HTTP = <https://tools.ietf.org/html/rfc2616>.

Peer-To-Peer (P2P) = outro tipo de comunicação na internet, mas acontece apenas entre clientes, sem servidor. As informações não se concentram em um lugar específico, como no torrent.

Intermediários = locais que os dados mandados passam no caminho entre navegador e servidor. Como roteador, modem, provedor.

HTTPS = o http, só que com segurança, usa também o Secure Sockets Layer (SSL) e o Transport Layer Security (TLS). A identidade online recebe o nome de certificado digital, e ganha uma chave pública que criptografa os dados enviados. Só quem consegue descriptografar é o servidor, que possui a chave privada.

Endereço = <http://www.qualquercoisa.com.br>.

Endereço IP = o mesmo que o endereço, mas como números, como em 216.58.202.238, que o computador entende melhor.

Domínio = www.qualquercoisa.com.br. É o nome bonitinho que a gente lembra, ao invés dos números.

DNS = servidor, banco de dados que diz a correspondência entre endereços de ip e domínios. Se dou um domínio, o navegador manda a informação pro dns e ele manda o ip correto, para eu acessar o site que procuro.

Raiz = o .com.br é a raiz, mostra a origem (comercial, educação, governamental) e nacionalidade.

Recurso = o /páginaespecífica de quando acesso algum caminho no site.



Requisições = o http trabalha como os correios: uma carta ou requisição é independente da outra. Ele é stateless, e manda para o endereço pedido a informação dada. O que precisa ser armazenado durante a sessão fica nos cookies.

Status Code = código da resposta do html para a requisição, como 200 (sucesso), 301 (movido para outro lugar), 404 (recurso não existente), 500 (erro no processamento). O primeiro algarismo dita o conteúdo da mensagem: 2 é sucesso, 3 redirecionamento, 4 erro do cliente, 5 erro do servidor.

Divirta-se = <httpstatusdogs> e <httpcats>.

GET = método para receber dados, como nas barras de pesquisa. Implementa o resultado no url da resposta. Os parâmetros são indicados por ?, e separados por &.

POST = método para enviar dados, como no login e senha. Implementa no corpo da requisição, para confidencialidade.

DELETE = método para excluir dados.

PUT = método para atualizar dados.

Formatos = não existe apenas o html, mas outros como xml ou json.

REST = junção dos métodos, formatos e uri.

HTTP2 = atualização do http, que tomou algumas medidas de segurança e compactação, como dados binários, hpack, gzip e tls.

Usa header stateful, em que é guardado o que será usado novamente, não precisa enviar a cada vez. Usa server push, onde o servidor manda todos os dados que serão necessários de uma vez, não precisa ficar pedindo. É multiplexing, não precisa esperar a resposta de uma requisição para fazer outra.