



01. O que é HTTP

📅 Date	@09/11/2022
📁 Categoria	Java
📖 Curso	HTTP: Entendendo a web por baixo dos panos

Tópicos

- Este curso foi atualizado!
- Introdução
- O que é o HTTP
- Objetivo do treinamento
- Modelo Client-Server
- Papel do HTTP entre Cliente e Servidor
- Para saber mais: Peer-To-Peer
- Para saber mais: Outros protocolos
- Para saber mais: Arquitetura da Alura
- O que aprendemos?

Papel do HTTP entre Cliente e Servidor

O cliente inicia a comunicação e o servidor responde. No entanto, qual é o papel do **HTTP**

entre o cliente e o servidor? Estabelecer regras de comunicação.

Se você compreende este texto, é porque você sabe português! Para que alguém consiga se comunicar com você, esse alguém deverá usar o português (supondo que

você desconheça outro idioma, é claro). Isso significa que, sua regra (protocolo) de comunicação com o mundo é a língua portuguesa, que define a forma com que as informações devem chegar até você (através do vocabulário, regras de gramática e etc). Uma outra pessoa que conheça português irá usar do mesmo formato, já que vocês possuem um idioma em comum.

Na internet, como já vimos, o idioma mais comum é o HTTP. Ele é responsável por definir a forma de como os dados são trafegados na rede através de várias regras. Portanto, todo mundo que conhece o idioma HTTP poderá receber e enviar dados e participar dessa conversa!

Para saber mais: Peer-To-Peer

Você já usou *torrent* para baixar algum arquivo na internet? Caso sim, aproveitou um outro modelo de comunicação, o **P2P** ou *Peer-To-Peer*!

O modelo **Cliente-Servidor** não é o único modelo de comunicação na rede, nem sempre o mais adequado. Por exemplo, imagine que precisemos contar as letras de 20 palavras. No caso do modelo *Cliente-Servidor*, quem fará esse trabalho é o servidor, certo? E se precisar contar as letras de 1 milhão de palavras? Muito trabalhoso para o servidor, não?

O modelo **Cliente-Servidor** tenta centralizar o trabalho no servidor, mas isso também pode gerar gargalos. Se cada Cliente pudesse ajudar no trabalho, ou seja, assumir um pouco da responsabilidade do servidor, seria muito mais rápido. Essa é a ideia do P2P! Não há mais uma clara divisão entre *Cliente-Servidor*, cada cliente também é servidor e vice-versa!

Isto é útil quando você precisa distribuir um trabalho ou necessita baixar algo de vários lugares diferentes. Faz sentido?

Usando algum aplicativo de Torrent, o protocolo utilizado não é o HTTP, e sim o protocolo **P2P**, como *BitTorrent* ou *Gnutella*.

Para saber mais: Outros protocolos

- FTP
 - *File Transport Protocol*, protocolo para transferir arquivos.
- BitTorrent
 - Além de ser um protocolo, também é um aplicativo para troca de arquivos na internet.
- SMTP
 - *Simple Mail Transfer Protocol*, protocolo para enviar e-mails.
- SQL
 - *Structured Query Language*, não é um protocolo para internet, e sim uma **linguagem de consulta para banco de dados**.

Um banco de dados cuida dos dados de uma aplicação, é parecido com uma planilha de Excel. O SQL ajuda muito a acessar esses dados.

Um banco de dados não se preocupa em como os dados serão visualizados, ele só administra os dados! Aqui na Alura, o banco de dados guarda informações sobre os usuários, cursos, perguntas, respostas, etc.

O que aprendemos?

O que você aprendeu nesse capítulo?

- A arquitetura CLiente-Servidor
- Um protocolo é um conjunto de regras
- HTTP é um protocolo que define as regras de comunicação entre cliente e servidor na internet.
- HTTP é o protocolo mais importante da Internet