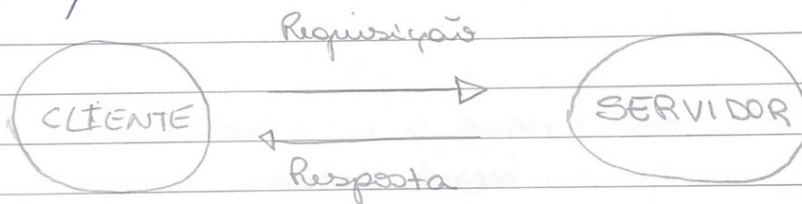


13/09/22

Internet - Entendendo como funciona

- Cliente aplicação conectada à internet. navegador.
- Servidor software específico em execução que irá responder a uma requisição do seu navegador.
 - ↳ armazenar, processar e entregar
- End. IP Internet Protocol. Identificador numérico a um dispositivo
- TCP/IP principal protocolo de envio e recebimento de dados da internet
- ISP Internet Service Provider é o provedor da internet
- DNS Domain Name System dispositivo com uma base de dados distribuída que gerencia os nomes de serviços, pc... conectado à internet.
- Port Number número de porta de 16 bits utilizado p' identificar um porta específico no computador
- Host dispositivo conectado à rede que pode receber ou transmitir pacotes IP
- HTTP Hyper-text Transfer Protocol comunicação entre navegadores e servidores na internet.
- URL Uniform Resource Locator e identifica um recurso web específico



O cliente é responsável por interagir com o usuário. Em uma aplicação web o cliente é responsável por definir a estrutura, a aparência e mecanismos para lidar com as interações do usuário.

ESTRUTURA da página ⇒ linguagem HTML Hyper Text Markup Language

<h1></h1> descreve o título principal (heading)

<p> descreve um parágrafo

 descreve uma lista não ordenada

↳ cada item da lista está num

Aparência da página → linguagem **CSS** Cascading **Style** Sheets

↳ permite alterar fonte, cor, tamanho; adicionar vídeos...

Interação com pessoas usuárias → linguagem **JavaScript**

↳ comportamentos dinâmicos

HTTPS Headers

contém metadados (dados sobre dados) que incluem o tipo de requisição (GET, POST, PUT, DELETE), o caminho URL, o endereço de IP, etc.

- Content-Type
- Server
- Status
- Host
- Cookie

HTTPS Body

corpo da mensagem que está sendo transmitida

Métodos HTTP

são verbos que dizem ao servidor o que fazer com os dados no URL

- GET

mais comum

é inferido encaminhadas pelo servidor para uma URL específico.

leituras → não podem ser modificados

idempotentes → o efeito da requisição no servidor será sempre o mesmo

Status 200 (OK) ou Status 404 (Not Found)

- POST

criar um novo recurso, com um formulário de login.

não é nem seguro e nem idempotente

Status code 201 (CREATED)

- PUT

atualizar o recurso identificado pelo URL

não é seguro, mas é idempotente

Status code 200 (OK) ou 404 (NOT FOUND)

- DELETE

deletar um recurso identificado pela URL

são idempotentes


Status code 200 (OK) ou 404 (NOT FOUND)

- REST

representational state transfer → estilo de arquitetura utilizado no design de aplicações web.

↳ são os dados necessários para que o servidor possa atender a uma ~~meta~~ determinada requisição

- Interface uniforme
- Stateless

 **Curl** serve para verificar conectividade, além de transferir dados via terminal

`curl [OPTION][URL]`

• OPTION

- ↳ `-u` ou `--user`: nome de usuário e senha
- ↳ `-T` ou `--upload-file`: transferir arquivos locais p/ remoto
- ↳ `-s` ou `--silent`: colocar curl em modo silencioso

• URL

endereço de web p/ se referir.