

Como funciona a web

cliente / servidor:

modelo baseado em requisições e respostas

cliente: solicita um serviço através de uma requisição para o servidor

servidor oferece serviços e executa a tarefa do servidor respondendo

rede: comunicação entre cliente e servidor

desvantagens: sobrecarga: se houver muitas mensagens de clientes o servidor pode sobrecarregar

Nó único: somente uma máquina manda mensagem podendo haver falha se a mesma falhar

vantagens: recursos centralizados: gerencia recursos comuns a todos evitando problemas de redundância

facilidade de manutenção: repara o servidor sem afetar o cliente

cliente: browser, html, css, JavaScript

servidor: servidor web (ex.: Google, youtube, etc)

http

protocolo de transferência de dados (páginas html), o cliente solicita uma requisição e o protocolo http envia um texto ao servidor esse pedido, a cada novo pedido é gerado um novo protocolo | Sem criptografia;

cliente — HTTP request —> servidor

←- HTTP response —-

métodos:

get: solicita informações e retorna uma resposta

post: solicitação para ser gravada no servidor

put: alterar dados no servidor -> enviar todos os dados incluindo os que não precisam de atualização

path: altera dados -> envia só os dados que quer atualizar

delete: remove informações no servidor

header: informações adicionais

date: data

tipo de conteúdo: texto, img, json etc...

método: get, post...

cookies: informações gerais, que persistem no geral

status code: 200 -> ok

201 -> criado

404 -> url inválida

403 -> cliente não tem acesso

401 -> sem autorização

500 -> erro

Body: informações do requisições

https: versão do http com criptografia nas requisições, ou seja uma requisição segura (camada 4 no tcp/IP)

HTPPS = HTTP + TLS ou SSL

TLS e SSL: criptografia, camada de segurança