

**Modelo/URL/Protocolo/Request**





01

Modelo  
Server/Client

02

URL, Domínio,  
IP e DNS

03

Protocolo HTTP

04

Request e  
Response



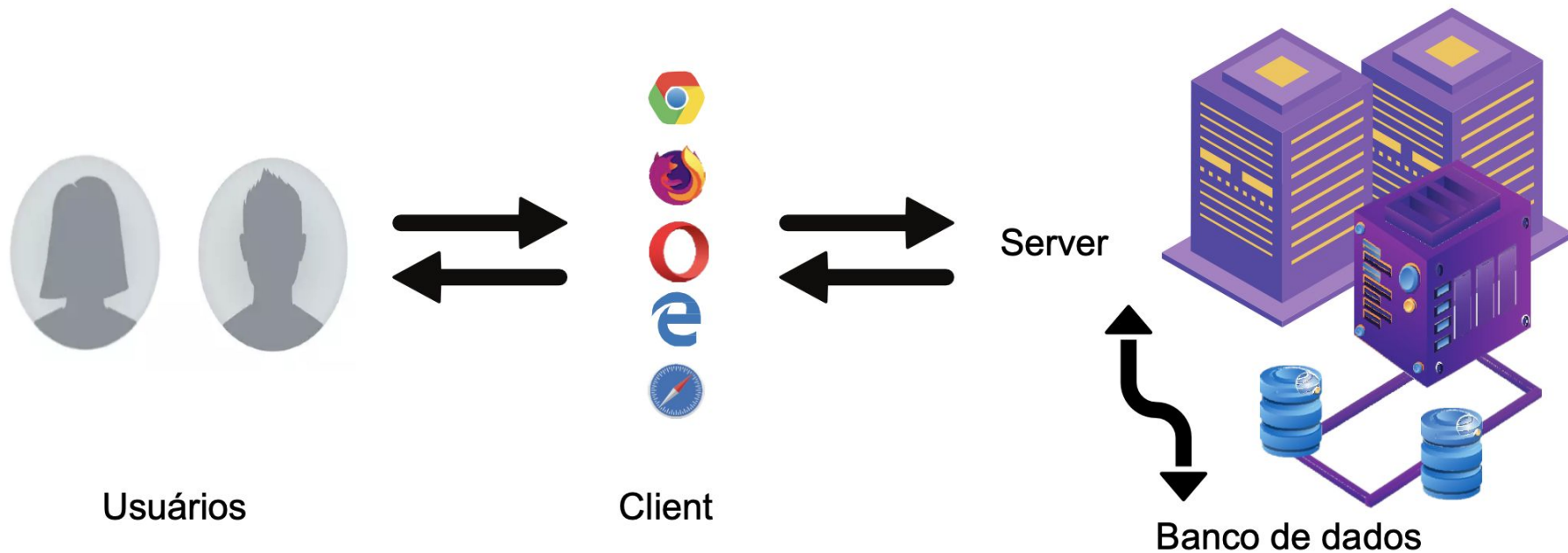
# Como funciona internet



# Comunicação



# Servidor/Cliente



# Client

Client é a interface que os usuários interagem, é essa camada que é responsável de **solicitar** serviços e informações de um ou mais servidores.

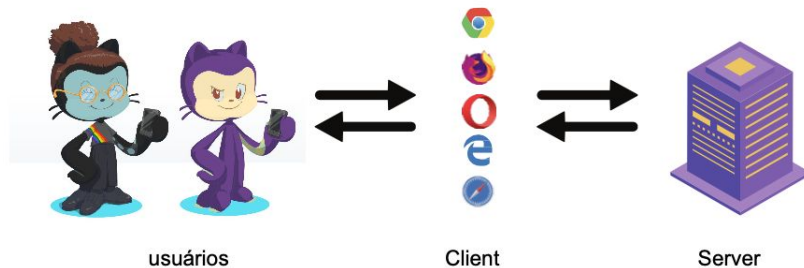
Algumas tarefas a serem realizadas pelo Cliente:

Manipulação de tela

Interpretação de menus ou comandos

Entrada e validação dos dados Recuperação de erro

Manipulação de janelas Gerenciamento de som e vídeo (em aplicações multimídia)



# Server

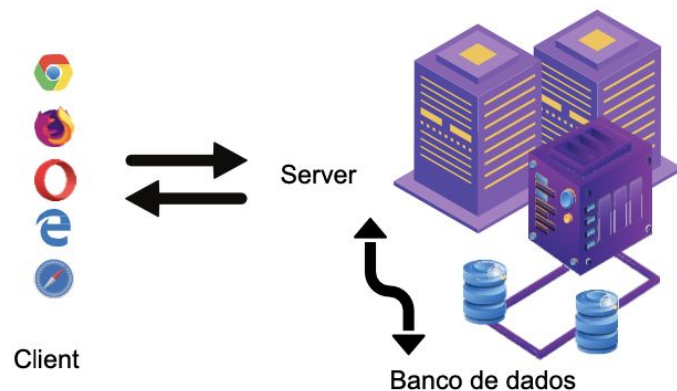
O Servidor é o responsável pelo processo, organização e gerenciamento das informações.

É ele que **responde às solicitações** feitas pelo Client.

Ele é um processo reativo, disparado pela chegada de pedidos de seus clientes

O processamento do servidor geralmente inclui:

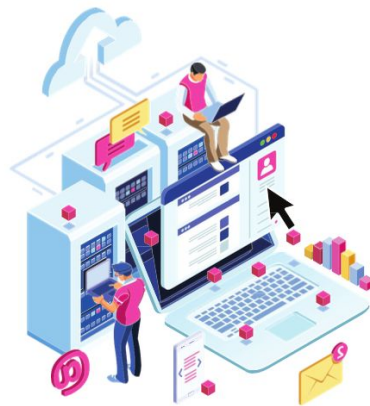
- Acessar
  - Organizar os dados compartilhados
  - Fazer a comunicação com o Banco de Dados
  - Atualizar dados previamente armazenados
- Gerenciamento dos recursos compartilhados.



[www.google.com.br](http://www.google.com.br)

O que acontece quando acessamos um site?

- 1 A URL é processada
- 2 É feita uma requisição
- 3 É dada uma Resposta
- 4 A página é renderizadas e aparece na tela





# A URL é processada

Todo site tem um **domínio**, normalmente é por ele que acessamos e conhecemos o Site.

Porém, no Server esse site não está registrado pelo nome de domínio, e sim pelo **endereço de IP**

Internet Protocol Address é o endereço exato de onde o site está dentro do servidor.

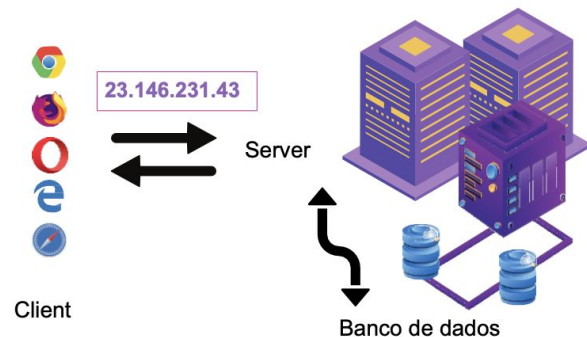
Então, antes de uma requisição ser feita o domínio deve virar o IP, e pra isso, usamos o **DNS**, o Domain Name System (Sistema de Nome de Domínio) que é como um grande dicionário de domínio para IP que já vem "de fábrica" no browser

## O Request é enviado

Agora com endereço certo, o Client faz uma requisição, ou Request, cheio de informações desejadas.

Pra que isso aconteça, tanto o Server quando o Client devem "falar a mesma língua".

Na maioria dos casos, essa comunicação entre Server e Client é feita a partir do **Protocolo HTTP**

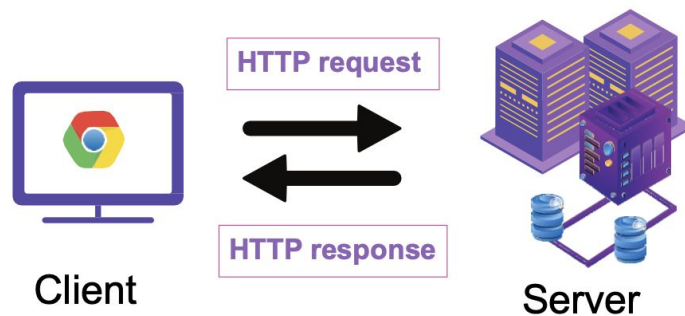


## O Request é enviado

Protocolo de Transferência de Hipertexto, o HTTP, é um protocolo usado dentro do modelo Client/Server é baseado em pedidos (requests) e respostas (responses).

O protocolo HTTP define um conjunto de métodos de requisição responsáveis por indicar a ação a ser executada.

Eles são chamados de **Verbos HTTP ou Métodos HTTP**.



# O Request é enviado

Os verbos HTTP mais utilizados são:

- GET
- POST
- PUT
- PATCH
- DELETE

Cada um deles corresponde a uma ação real no banco de dados.

**GET**

**ler**

**POST**

**criar**

**PUT**

**substituir**

**PATCH**

**modificar**

**DELETE**

**excluir**

## O Server responde

Quando o Client faz um Request o Server envia um **Response**.

E na resposta tem, além do resultado do que foi pedido, um código de status numerico padronizado

código	tipo de resposta
100-199	informação
200-299	sucesso
300-399	redirecionamento
400-499	erro do cliente
500-599	erro de servidor

4

# O site aparece na tela

[Gmail](#) [Imagens](#)



Google



Pesquisa Google

Estou com sorte

[Ajuda](#) [Negócios](#) [Como funciona a Pesquisa](#)

[Privacidade](#) [Termos](#) [Configurações](#)

---

# Dev Front

São as pessoas que são responsáveis muito mais do que construir as telas bonitas e funcionais.

Elas tem que criar aplicações preparadas para enviar Requests corretamente e receber as Responses, também disponibilizar elas para os usuários



# Dev Back

Dentro desse fluxo, são as pessoas que constroem toda a dinâmica do recebimento de Requests, o envio das Responses corretas, o tratamento das Responses, as execuções de ação no Banco de Dados e a disponibilização para a Dev Front



# Os Devs Backend

