WEB – HISTÓRIA

Em 1989, Tim Berners-Lee propôs um sistema de gestão de informação baseado em hipertexto

Robert Cailliau, reformulou e redistribuiu a proposta, referindo-se ao Sistema como a "World Wide Web"

No final de 1990, Berners-Lee tinha implementado um servidor e um navegador de linha de comandos, utilizando a versão inicial do protocolo HyperText Transfer Protocol (HTTP)

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

WEB – HISTÓRIA

Em meados de 1991, este servidor e navegador foram disponibilizados pelo CERN

No início de 1993, já havia cerca de 50 sítios diferentes a correr servidores HTTP, e pelo outono desse ano cresceu até 200

Adicionalmente, e uma vez que a especificação do protocolo HTTP estava aberta, outros foram escrevendo o seu próprio código para servidor e navegador, incluindo interfaces gráficas com suporte de imagens

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

WEB – COMPONENTES

Existem 3 componentes básicos que definem esta tecnologia Web:

- Uma linguagem de marcação para formatação de documentos de hipertexto - HTML
- Um esquema de notação uniforme para lidar com recursos acessíveis através da rede - URI
- Um protocolo para o transporte de mensagens através da rede - HTTP

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

WEB – COMPONENTES

A linguagem de marcação que permitiu referenciar documentos através de ligações foi o HyperText Markup Language (HTML)
O esquema de notação uniforme chama-se Uniform Resource Identifier (URI)
HTTP é o núcleo da fundação da World Wide Web, ele foi projetado para transportar mensagens específicas através da rede

SEBASTIAO@DI.UBI.PT
FACULDADE ENGENHARIA
Departamento de Informática

APLICAÇÕES WEB

O que é uma aplicação?

Uma Aplicação é um programa ou um conjunto de programas desenhados para servir um utilizador final

Se os utilizadores interagem com a aplicação através do browser, a aplicação designa-se por Aplicação Web

Uma Aplicação Web é desenhada para ajudar o utilizador a cumprir uma determinada tarefa

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

APLICAÇÕES WEB

Uma Aplicação Web consiste em duas peças importantes:

- Base de Dados: localizada num servidor da internet, a BD consiste na memória de longo prazo da nossa Aplicação Web;
- Aplicação: consiste no programa ou grupo de programas que realizam as seguintes tarefas:
  - Criam a interface com os utilizadores via browser;
  - Interagem com os utilizadores processando a informação digitada por estes nas janelas do browser;
  - Armazenam a informação na BD e/ou consultam e extraem informação da base de dados disponibilizando-a aos utilizadores via browser

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

CONCEITOS-CHAVE

O que é um protocolo de comunicação?

Um protocolo é um conjunto de regras que regula a interação entre dois ou mais interlocutores

No caso de um protocolo de rede temos a interação entre componentes de software e hardware dos dispositivos

Exemplos

HTTP

TCP/IP

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

CONCEITOS-CHAVE

Tipos de protocolos

- Protocolos end-to-end:
  - Garantem a integridade dos dados transmitidos através de ecanismos de reconhecimento e retransmissão
- Protocolos de roteamento:
  - Determinam o caminho de um pacote de dados da fonte até ao destino
- Protocolos de hardware em um adaptador de rede:
  - Controlam o fluxo de bits sobre os cabos que interligam dois dispositivos

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

CONCEITOS-CHAVE

Como funciona a Internet?

- Domínio registado: www.omeusitio.sufixo
- WebServer
- DNS – Server: (para resolver os nomes)
- Ligação a um ISP: (para rotear o domínio)

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

CONCEITOS-CHAVE

Que acontece quando requisitamos uma página da Internet?

Primeiro seu computador envia uma mensagem requisitando uma ligação com o servidor remoto (TCP connection request)

O servidor Web recebe sua requisição e responde afirmativamente (TCP connection reply)

Com a ligação já estabelecida, seu computador requisita então a página procurada (GET http://www.omeusitio.sufixo)

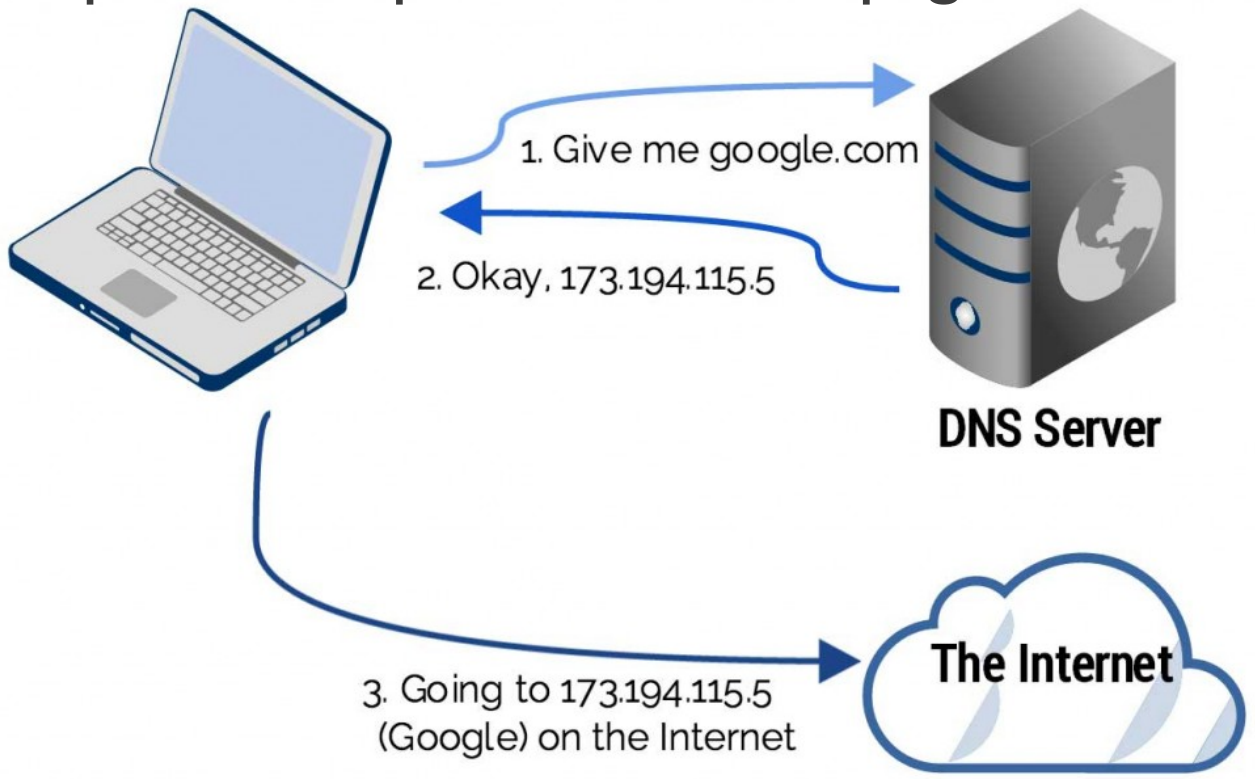
O servidor web remoto envia o ficheiro com o código HTML correspondente

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

CONCEITOS-CHAVE

Que acontece quando requisitamos uma página da Internet?



SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

O que oferecem as páginas dinâmicas que não oferecem as estáticas?

- E-commerce
- Chats
- Bases de Dados
- Pesquisas
- Dados em tempo-real
- …

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

Linguagens do lado servidor vs lado cliente

Existe diferença entre linguagens como Java ou JavaScript e outras como ASPX ou PHP

Prós e contras de cada tipo?

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

- Linguagens do lado cliente
  - Analisadas pelo browser do utilizador
  - Reagem aos dados introduzidos
  - Pode ser vistas e editadas pelo utilizador
  - Não podem armazenar dados além de uma atualização da página
  - Não é possível ler arquivos fora de um servidor diretamente

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

Linguagens do lado servidor

- Armazenam dados persistentes
- Não podem ser vistas pelo utilizador
- Só podem responder a solicitações HTTP para um URL particular
- Cria a página final que o utilizador vê

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

Linguagens do lado servidor

Existem diferentes linguagens do lado servidor utilizadas para o desenho de webs dinâmicas

ASP: Active Server Pages

PHP: Hypertext PreProcessor

PERL: Practical Extraction and Report Language

JSP: Java Server Pages

Ruby, Python, etc.

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

O que é um servidor?

O termo servidor (server) é geralmente aplicado a computadores, embora um servidor possa equivaler a um software, partes de um sistema computacional, ou até mesmo a uma máquina que não seja necessariamente um computador

Em informática, um servidor é um sistema de computação que fornece serviços a uma rede de computadores

Os computadores que acedem os serviços de um servidor são chamados clientes

As redes que utilizam servidores são do tipo cliente-servidor

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

Servidores dedicados

- Um servidor, ocasionalmente, pode fornecer mais de um serviço simultaneamente
- Um servidor pode actuar como servidor de DNS e proxy ao mesmo tempo
- Servidores que atuam como um único tipo de servidor são chamados de servidores dedicados
- Servidores dedicados têm a vantagem de serem rápidos a atender as requisições

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

- Hardware de servidores
  - Interfaces de entrada e saída e discos rígidos de alto desempenho e confiabilidade
  - Disco rígido de padrão SCSI - arranjos RAID
  - Processadores de alta velocidade – multiprocessadores
  - Quantidade de memória RAM – caching de dados
  - Sistema de dissipação de calor – coolers
  - Placas do tipo hot swapping

SEBASTIAO@DI.UBI.PT
FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

- Software
  - A rede cliente-servidor necessita de um sistema operativo que reconheça o tipo de rede
  - Os sistemas operativos para redes cliente-servidor são:
    - Windows (Windows Server 2008/2012/2016, Windows 7, Window 8, Windows 10, etc.)
    - Unix (Ubuntu, CentOS, SuseLinux, RedHat, etc.)
    - Solaris
    - FreeBSD
    - Mac OS X

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

Tipos de Servidores

Servidor de fax: Para transmissão e recepção automatizada de fax pela Internet.
Servidor de arquivos: Armazena arquivos de diversos utilizadores
Servidor web: Responsável pelo armazenamento de páginas web
Servidor de e-mail: Responsável pelo armazenamento, envio e receção de mensagens de correio electrónico
Servidor de impressão: Responsável por controlar pedidos de impressão de arquivos dos diversos clientes
Servidor de banco de dados: Servidor que gere informações contidas num base dados, como por exemplo, um perfil de utilizadores
Servidor DNS: Responsáveis pela conversão de endereços de sites em endereços IP e vice-versa
Servidor proxy: Servidor que actua como uma cache, armazenando páginas da internet recém-visitadas, aumentando a velocidade de carregamento destas
Servidor FTP: Permite acesso de outros utilizadores a um disco rígido ou servidor
Servidor webmail: servidor para criar emails na web

SEBASTIAO@DI.UBI.PT

FUNCIONAMENTO DE APLICAÇÕES NA WEB

O que é um servidor Web?

A expressão servidor web pode significar duas coisas:

Um programa de computador responsável por aceitar pedidos HTTP de clientes, geralmente browsers, e servi-los com respostas HTTP, incluindo opcionalmente dados, que geralmente são páginas web, tais como documentos HTML com objectos embutidos (imagens, etc.);

Um computador que executa um programa que fornece a funcionalidade descrita acima

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

- O que é um servidor Web?
  - Os servidores web são responsáveis por armazenar e trocar informações com outras máquinas
  - Há pelo menos dois participantes envolvidos em cada troca de informações:
    - Um cliente, que solicita informações
    - Um servidor que atende a esses pedidos
  - Cada lado exige também um programa especializado para gerir a troca de dados
    - No lado do cliente, um browser
    - No lado de servidor – há uma variada gama de opções

SEBASTIAO@DI.UBI.PT
FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

- O que é um servidor Web?
  - O software depende do sistema operativo escolhido para o servidor
  - O servidor Web também cuida de outras tarefas
    - Fazer a distinção entre vários tipos de erros e dados
    - Designar o código apropriado para qualquer erro interno e enviar de volta para o browser
    - Fazer a distinção entre vários elementos duma página Web (como .GIFs, JPEGs etc.) de forma a que o browser saiba que arquivos usar na hora de formatar a página
    - Registo de estatísticas, segurança de manipulação e criptografia
    - Funções de comércio electrónico

SEBASTIAO@DI.UBI.PT
FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

O que é um servidor Web?

Os servidores web são a espinha dorsal da Internet, são eles que alojam todas as páginas, incluindo os mecanismos de pesquisa e servem como base para todo o tipo de aplicações existentes na Web

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

- Servidores Web
  - Os servidores mais comuns são
    - O IIS (Internet Information Services) da Microsoft
    - O Apache da Apache Software Foundation
    - O TomCat da Apache Software Foundation

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

- Limitações dos servidores Web
  - Limite de ligações por endereço e por porto (2-60000)
  - Limite de respostas por segundo

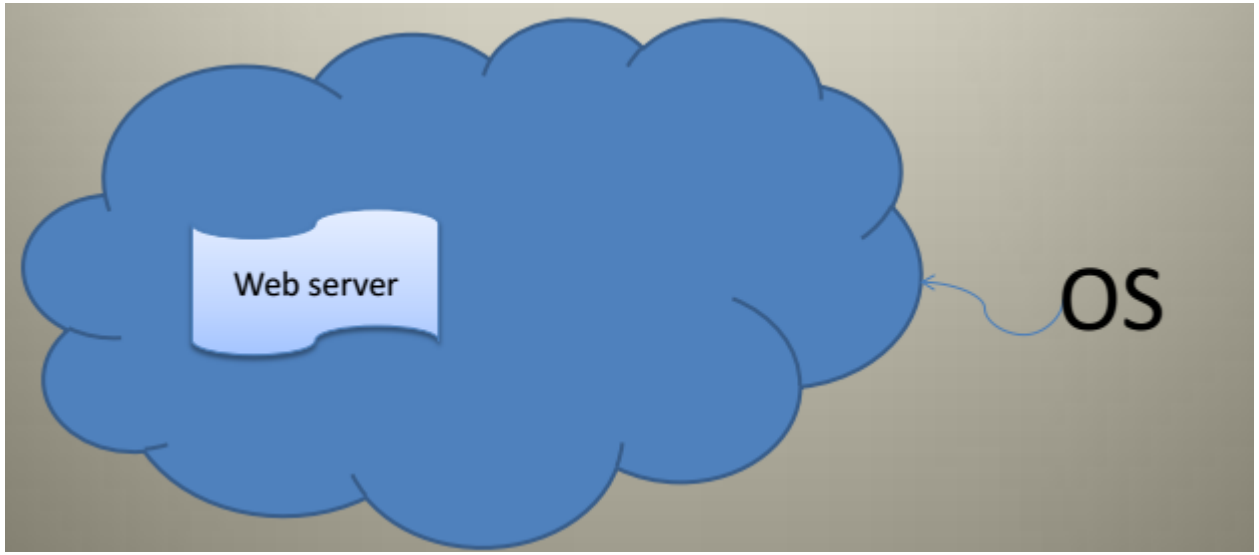
SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

Funcionamento dos servidores Web

Corre como um processo num sistema operativo



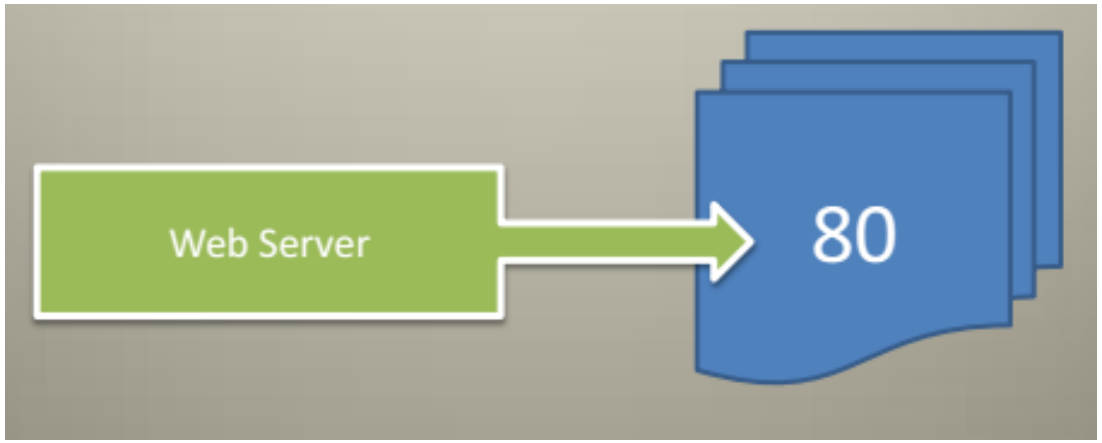
SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

Funcionamento dos servidores Web

O servidor web escuta os portos especiais, por exemplo o 80



SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

Funcionamento dos servidores Web

Quando o pedido chega, processa o pedido

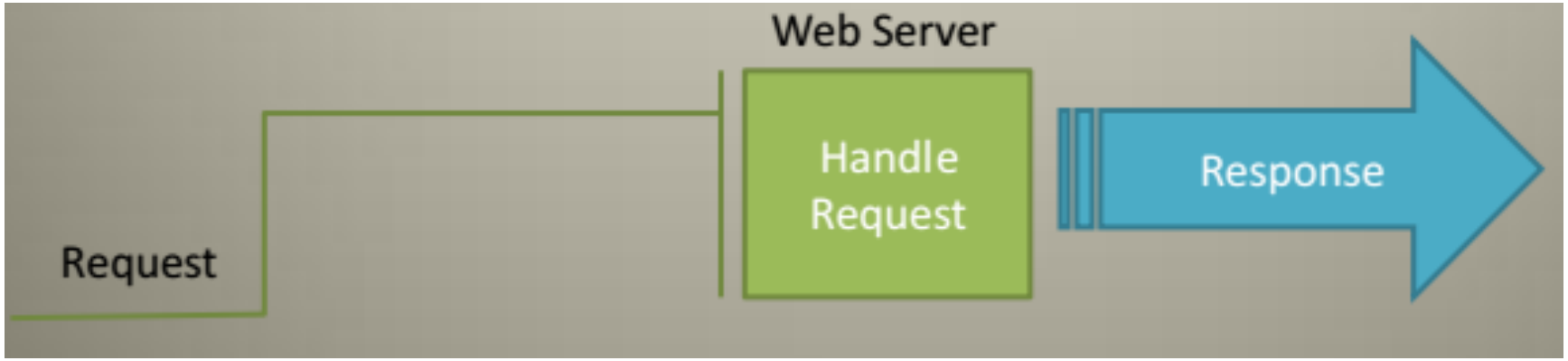
SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

Funcionamento dos servidores Web

Gera a resposta e envia de volta ao cliente



SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

- Funcionamento dos servidores Web
  - Conseguem mapear os conteúdos com URI/URL
    - Um recurso no sistema local significa um pedido estático
    - Uma aplicação interna ou externa implica um pedido dinâmico

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

FUNCIONAMENTO DE APLICAÇÕES NA WEB

- Tipos de servidores web
  - O primeiro servidor web surgiu em 1989
    - O CERN httpd

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

SERVIDORES WEB

Tipos de servidores web

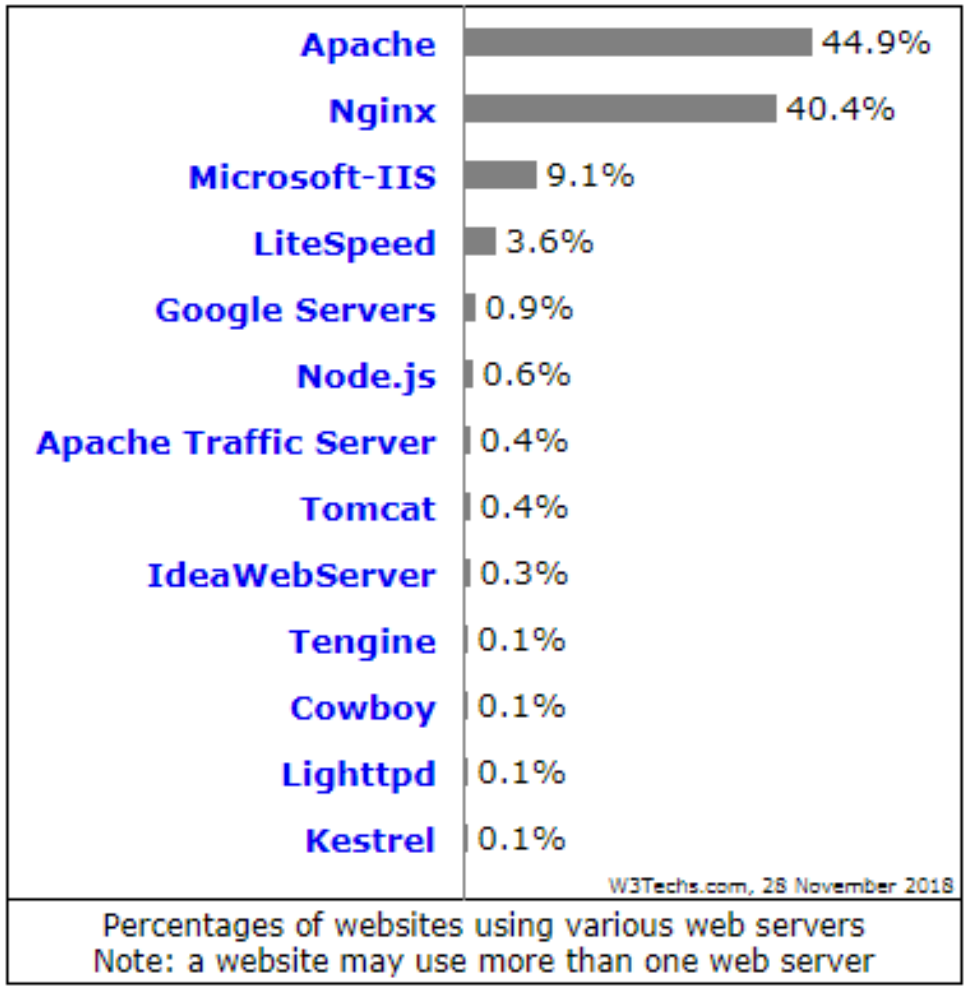
- Nginx web server
- Internet Information Services(IIS)
- Lighttpd
- Sun Java System Web Server
- Jigsaw Server
- Apache Tomcat
- Google server

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

SERVIDORES WEB

Tipos de servidores web
Utilização atual



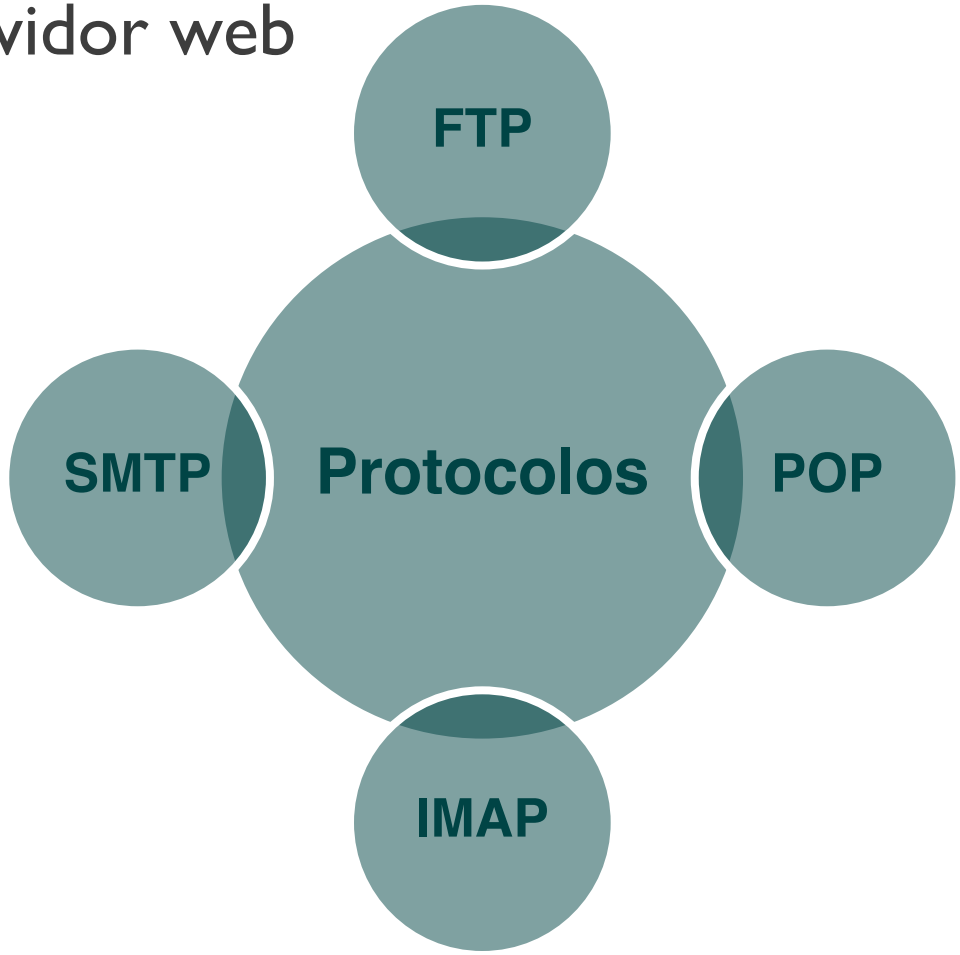
Percentages of websites using various web servers
Note: a website may use more than one web server

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

SERVIDORES WEB

Protocolos de um servidor web

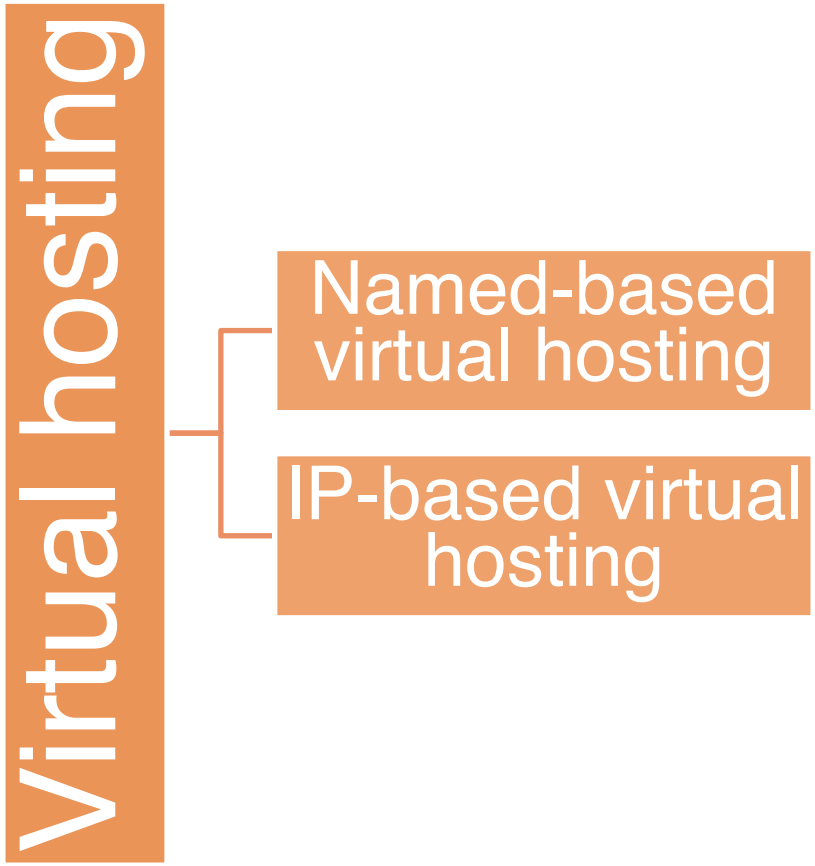


SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

SERVIDORES WEB

Alojamento virtual



SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

SERVIDORES WEB

Alojamento virtual – baseados em nome
- Múltiplos domínios para o mesmo endereço IP
- Determinação do anfitrião no cabeçalho HTTP
- Certificação SSL/TLS
- Registo no DNS

SEBASTIAO@DI.UBI.PT
FACULDADE ENGENHARIA
Departamento de Informática

SERVIDORES WEB

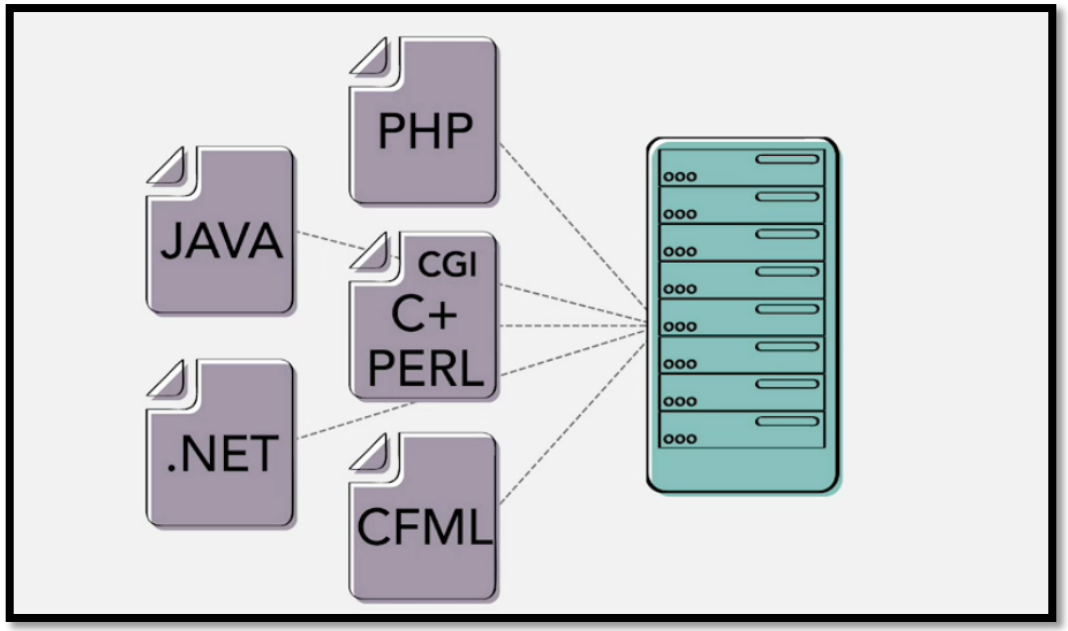
- Alojamento virtual – baseados em IP
  - Apenas um sítio para o mesmo endereço IP
  - Aumenta o esforço administrativo
  - Problema dos IPv4 finitos

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

SERVIDORES WEB

Código do lado do servidor



SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

URI

Uma peça do puzzle Web seria um esquema de notação para fazer referência a recursos acessíveis em qualquer lugar na Internet
Tim Berners-Lee criou este sistema de notação para que ele fosse flexível e extensível para suportar outros protocolos além de HTTP
Este esquema de notação é conhecido como o URL ou Uniform Resource Locator

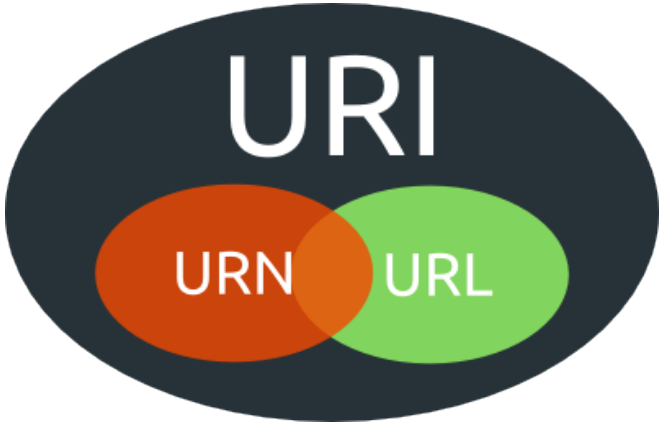
SEBASTIAO@DI.UBI.PT
FACULDADE ENGENHARIA
Departamento de Informática

URI

Um identificador é um objeto que pode atuar como uma referência a alguma coisa que tem identidade

No caso da URI, o objeto é uma sequência de caracteres com uma sintaxe RFC3986 restrito

Um URI pode ainda ser classificada como um Locator (URL), um nome (URN), ou ambos



SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

URI

Cada URL consiste no seguinte, na ordem dada:

- o nome do esquema (ou protocolo)
- dois pontos, duas barras ://
- um (nome de domínio ou endereço IP) do anfitrião
- dois pontos seguido por um número de porto, opcionalmente :80
- o caminho completo do recurso, opcionalmente
- uma query string, opcionalmente
- um identificador de fragmento

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

URI

A notação geral associada a URLs é:

```
scheme://host[:port#]/path/.. ./[;url-
params][?query-string][#anchor]
```


SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

URI

Nome do esquema

Para ligações HTTP, o nome do esquema pode ser **http** ou **https**

Hyper**t**ext **T**ransfer **P**rotocol **S**ecure (HTTPS) é apenas HTTP no topo do protocol **SSL/TLS**

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

URI

Número de porto

- O porto por omissão de um servidor HTTP server é o porto **80**
- Normalmente são usados outros: 8080, 8000
- O número de porto pode ser omitido no URL se for o 80

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

URI

Query string

- O texto da query contém dados a serem passados pelo software que está a corer no servidor
- Pode conter pares de nome/valor separados por &

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

URI

- Identificador de fragmentos
  - O identificador de fragmentos, se presente, especifica uma parte ou posição respetivo ao recurso ou documento
  - Se utilizado com HTML, representa um elemento na página identificado pelo seu **id**

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

URI

Exemplo

http://www.mywebsite.com/sj/test;id=8079?name=sviergn&x=true#stuff

```
ESQUEMA= http
ANFITRIÃO= www.mywebsite.com
CAMINHO= /sj/test
PARÂMETROS URL = id=8079
QUERY STRING = name=sviergn&x=true
ÂNCORA = stuff
```


SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP - FUNDAMENTOS

HTTP é o protocolo base da World Wide Web (WWW)
É simples, o que é tanto uma vantagem como uma desvantagem
Muitas pessoas na indústria criticaram o HTTP pela sua falta de suporte de estado e funcionalidade limitada, mas o HTTP expandiu-se, enquanto protocolos mais avançados e sofisticados nunca realizaram o seu potencial

SEBASTIAO@DI.UBI.PT
FACULDADE ENGENHARIA
Departamento de Informática

HTTP - FUNDAMENTOS

O HTTP é uma aplicação de nível protocolar no conjunto do protocolo TCP/IP, e utiliza o TCP como a canal subjacente para transmitir mensagens

As bases e estrutura das mensagens HTTP podem ser sintetizadas em 3 afirmações:

- O protocolo HTTP utiliza o paradigma pedido/resposta, o que significa que um cliente HTTP envia um pedido HTTP para um servidor HTTP, que responde com uma resposta HTTP

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP - FUNDAMENTOS

A estrutura das mensagens de pedido e resposta é similar às mensagens de email: consistem num grupo de linhas que contém um cabeçalho, uma linha vazia e um corpo

O HTTP é um protocolo sem estado, o que significa que não há suporte explícito da noção de estado; uma transação HTTP consiste num pedido único de um cliente a um servidor, seguido de uma resposta única de volta ao cliente

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP - SERVIDORES, NAVEGADORES E PROXIES

Os servidores web e os navegadores trocam informação utilizando o HTTP, sendo os servidores HTTP

Da mesma forma, os navegadores web são por vezes referidos como clientes HTTP, mas a sua funcionalidade não está limitada ao protocolo HTTP

Os primeiros navegadores foram desenhados para suportar outros protocolos, incluindo FTP e Gopher

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP - SERVIDORES, NAVEGADORES E PROXIES

Hoje em dia, os navegadores suportam não apenas HTTP, FTP e acesso a ficheiros locais, mas e-mail e plugins

As proxies HTTP são programas que atuam tanto como servidores como clientes, fazendo pedidos a servidores web em nome de outros clientes

As proxies possibilitam as transferências HTTP através das firewalls

Quanto nos referimos a clientes HTTP, isto é aplicável a navegadores, proxies e outros programas personalizados

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – PARADIGMA PEDIDO/RESPOSTA

Em primeiro lugar, o HTTP baseia-se no paradigma pedido/resposta: os navegadores enviam mensagens para os servidores HTTP

Estes servidores geram mensagens que são enviadas de volta aos navegadores

As mensagens que são enviadas pelos servidores HTTP são chamados pedidos, e as mensagens enviadas de volta são chamadas respostas

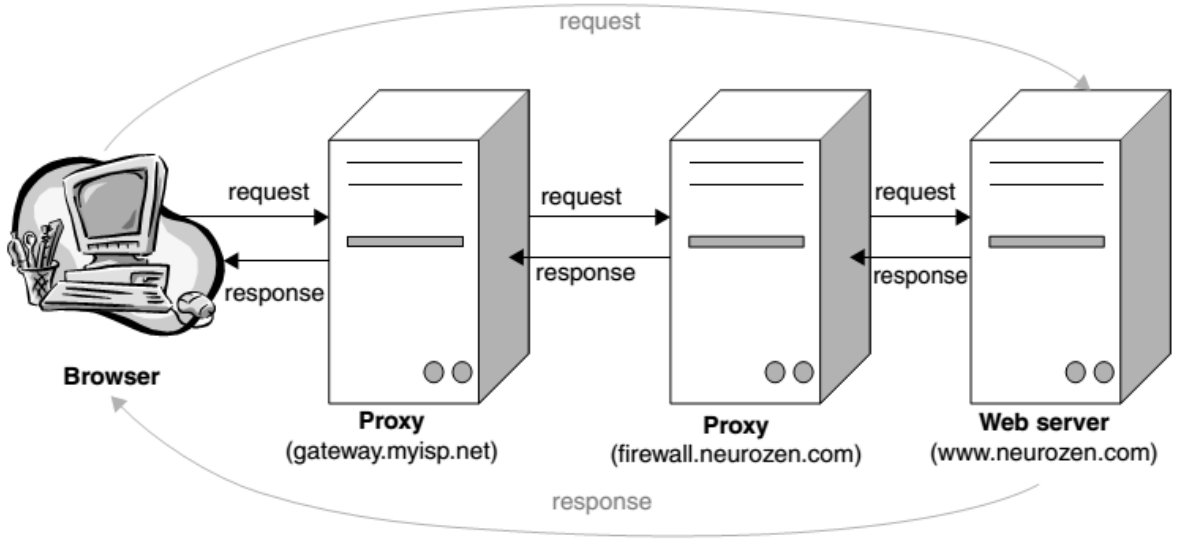
SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – PARADIGMA PEDIDO/RESPOSTA

Na prática, os servidores e navegadores raramente comunicam diretamente – há sempre pelo menos uma proxy entre eles

Uma ligação é definida como um circuito virtual composto por agentes HTTP, incluindo o navegador, o servidor e as proxies intermediárias que participa
processo



SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – PROTOCOLO SEM ESTADO

HTTP é um protocolo sem estado

Quando um protocolo suporta "estado", significa que fornece a interação entre o cliente e o servidor com uma sequência de comandos

O servidor tem que manter o "estado" da ligação através da transmissão de comandos sucessivos, até a ligação estar terminada

A sequência de comandos transmitidos e executados é normalmente chamada de sessão

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – PROTOCOLO SEM ESTADO

Muitos protocolos da Internet, incluindo o FTP, SMTP e POP não têm "estado"

A definição do HTTP como protocolo sem estado permitiu fazer as tarefas de maneira simples, mas também impôs limitações nas capacidades das aplicações web

Por definição, o tempo de uma ligação é apenas uma troca pedido/resposta

- Isto significa que que não existiria maneira de manter informação persistente entre a sessão entre interações sucessivas entre cliente e servidor

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – ESTRUTURA

As mensagens HTTP (ambos pedidos e respostas) (têm estrutura similar às mensagens de e-mail

- Consistem em blocos de linhas, compostos por um cabeçalho, uma linha vazia e um corpo de mensagem

A estrutura das mensagens HTTP é, porém, mais sofisticada

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – PEDIDO

A primeira linha contém um método de pedido seguido dos seus parâmetros:

- O caminho do documento (URL absoluto sem protocolo nem domínio)
- A versão do protocolo HTTP utilizado

```
GET /sj/index.html HTTP/1.1
```

As linhas seguintes representam um cabeçalho específico

O bloco final é de dados opcionais; está separada por uma linha vazia e contém mais informações, normalmente usadas no POST

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – MÉTODOS DOS PEDIDOS

- O método de pedido indica a ação a ser executada pelo servidor
- O padrão HTTP/1.1 define sete métodos
- Outros padrões podem adicionar métodos adicionais

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – MÉTODOS IDEMPOTENTES DOS PEDIDOS

Um método idempotente é um método onde os efeitos colaterais no servidor de pedidos idênticos são os mesmos de um pedido único

- O **HEAD** e o **GET** também são idempotentes
- O **PUT** é utilizado para transferir um novo recurso para o servidor. Se o recurso já existir, é substituído; se não existir é criado
- O **DELETE** é utilizado para remover um recurso do servidor

Estes métodos são opcionais

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – OUTROS MÉTODOS DE PEDIDOS

- **POST**: é utilizado para despoletar uma ação no servidor; tem efeitos secundários e pode ser utilizado para modificar uma base de dados p.e.
- **OPTIONS** e **TRACE**

Estes métodos são opcionais

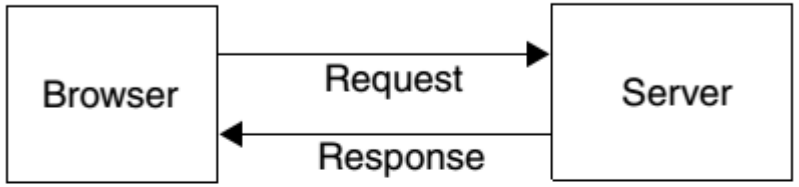
SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – OUTROS MÉTODOS DE PEDIDOS

Pedido inicial do utilizador de "http://www.cs.rutgers.edu/~shklar/"

```
GET /~shklar/ HTTP/1.1
Host: www.cs.rutgers.edu
```



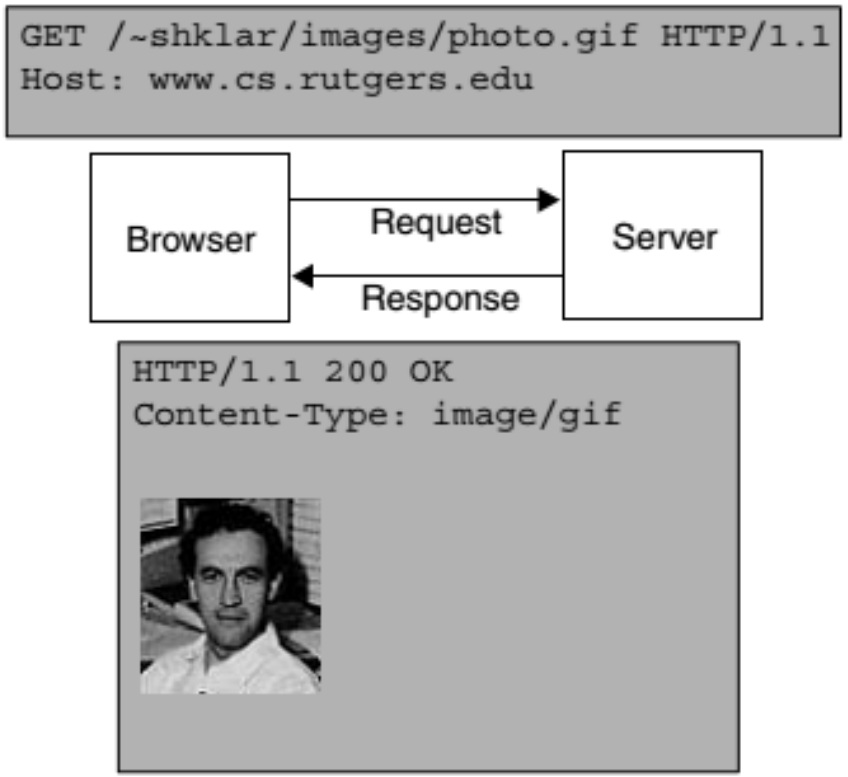

```
HTTP/1.1 200 OK
Content-Type: text/html

<html>
<head> ... </head>
<body>
...
<img src="./images/photo.gif">
...
</body>
</html>
```


SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – OUTROS MÉTODOS DE PEDIDOS

Pedido do navegador "http://www.cs.rutgers.edu/~shklar/images/photo.gif"



SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – RESPOSTA

Aquando da resposta a um pedido de um cliente, o servidor devolve um número de 3 dígitos para indicar se o pedido foi processado com sucesso

As respostas podem ser agrupadas em 5 categorias:

- **informacional** (1xx)
- **sucesso**(2xx)
- **redirecionamento** (3xx)
- **erro no pedido do cliente** (4xx)
- **erro do servidor** (5xx)

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – RESPOSTA

Exemplo:

```
HTTP/1.0 200 OK
Date: Sun, 11 Feb 2001 23:59:59 GMT
Content-Type: text/html
Content-Length: 1354
<html>
<body>
<h1>Hello World!</h1>
…
</body>
</html>
```


SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – CÓDIGOS DE RESPOSTA

Sucesso

200 OK - O pedido foi bem sucedido. A informação retornada com a resposta está dependente do método utilizado no pedido

GET uma entidade correspondente ao recurso pedido é enviada na resposta

HEAD o cabeçalho da entidade correspondente ao recurso pedido é enviado na resposta sem nenhum corpo

POST uma entidade que descreve ou contém o resultado da ação

201 Criado - O pedido foi satisfeito e resultou num novo recurso criado

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – CÓDIGOS DE RESPOSTA

Sucesso

- 202 Aceite - O pedido foi aceite para processamento, mas o processamento não foi concluído
- 204 Sem conteúdo - O servidor satisfez o pedido mas não necessita de devolver um corpo-entidade
- 206 Conteúdo parcial - O servidor satisfez o pedido GET para o recurso; o recurso deve ter incluído um campo de intervalo no cabeçalho

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – CÓDIGOS DE RESPOSTA

Redirecionamento

301 Movido permanentemente - O recurso pedido foi atribuído a um novo URI permanente e quaisquer referências futuras deverão utilizar os URI devolvidos. O novo URI permanente deve ser dado pelo campo localização na resposta

304 Não modificado - Se o cliente realizou um pedido GET condicional e o acesso é permitido mas o documento não foi alterado

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – CÓDIGOS DE RESPOSTA

Erro no pedido do cliente

- 400 Pedido errado - O pedido não foi compreendido pelo servidor devido a erro de sintaxe
- 401 Não autorizado - O pedido reque autenticação do utilizador. A resposta deve incluir um campo no cabeçalho WWW-Autenticado contendo uma prova aplicável ao recurso solicitado
- 403 Proibido - O servidor compreendeu o pedido, mas recusa-se a satisfazê-lo

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – CÓDIGOS DE RESPOSTA

Erro no pedido do cliente

- 404 Não encontrado - O servidor não encontrou nenhuma correspondência com o pedido de URI
- 405 Método não permitido - O método especificado no pedido não é permitido para o recurso identificado no URI. A resposta deve incluir um cabeçalho que permita conter uma lista de métodos válidos
- 408 Pedido expirado - O cliente não produziu um pedido dentro do tempo que o servidor estava preparado para esperar

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – CÓDIGOS DE RESPOSTA

Erro do servidor

- 500 Erro interno de servidor - O servidor encontrou uma condição inesperada que não permitiu satisfazer o pedido
- 502 Saída errada - O servidor, enquanto gateway ou proxy, recebeu uma resposta inválida ao tentar satisfazer o pedido
- 503 Serviço indisponível - O servidor não consegue lidar com o pedido de momento, devido a carga temporária ou manutenção

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – CABEÇALHOS

Os cabeçalhos HTTP são uma forma de mensagem de meta-dados

A utilização correta de cabeçalhos permite construir aplicações sofisticadas que permitem estabelecer sessões, definir políticas de cache, controlar autenticação e implementar lógicas de negócio

A especificação do protocolo HTTP faz uma distinção clara entre cabeçalhos globais, cabeçalhos de pedidos, cabeçalhos de resposta e cabeçalhos de entidade

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – CABEÇALHOS

Cabeçalhos globais aplicam-se a mensagens tanto de pedido como de resposta mas não se descrevem o corpo da mensagem

Cabeçalhos de pedido permitem aos clientes passar informação adicional acerca deles próprios e acerca do pedido

Cabeçalhos de resposta ajudam o servidor a passar informação adicional acerca da resposta que não pode ser inferida do código de estado por si só

Cabeçalhos de entidade descrevem tanto corpos de mensagem como recursos alvo

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – CABEÇALHOS DE CLIENTE

Accept: Tipos de conteúdo que são aceites para resposta

Accept-Charset: Grupos de caracteres que são aceites

Accept-Encoding: Lista de codificações aceites

Accept-Language: Lista de línguas humandas aceites para resposta

Connection: Que tipo de ligação o agente prefere

Cookie: Um cookie HTTP cookie previamente enviado pelo servidor

Content-Length: O tamanho do corpo do pedido em octetos (bytes de 8-bit)

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – CABEÇALHOS DE CLIENTE

Content-Type: O tipo MIME do corpo do pedido (usado com POST/PUT)

Date: Data e hora que a mensagem foi enviada

Host: Nome do domínio do servidor (para alojamento virtual) e o porto TCP no qual o servidor está a escutar

If-Modified-Since: Permite que um 304 Não Modificado seja retornado se o conteúdo se mantiver inalterado

Range: Pedido apenar parcial de uma entidade; os bytes são numerados a partir do 0

User-Agent: O texto agente-utilizador

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – CABEÇALHOS DE CLIENTE

Exemplo

```
Accept: text/plain
Accept-Charset: utf-8
Accept-Encoding: gzip, deflate
Accept-Language: en-US
Connection: keep-alive
Cookie: username=johndoe; session_id=7f3fe5016a9cda0c4adbd44aeea9d511;
Content-Length: 348
Content-Type: application/x-www-form-urlencoded
Date: Tue, 15 Nov 1994 08:12:31 GMT
Host: www.google.com:80
If-Modified-Since: Sat, 29 Oct 2015 19:43:31 GMT
Range: bytes=500-999
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:12.0) Gecko/20100101 Firefox/21.0
```


SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – CABEÇALHOS DE SERVIDOR

Accept-Ranges: Que tipos de conteúdo parciais suporta o servidor
Allow: Ações válidas para um recurso específico, usado para o método 405 Não Permitido
Cache-Control: Informa de todos os mecanismos de cache do servidor para o cliente, e se podem guardar em cache os objetos
Content-Encoding: O tipo de codificação utilizada nos dados
Content-Language: A língua do conteúdo
Content-Length: O tamanho do corpo da resposta em octetos (bytes de 8-bit)

SEBASTIAO@DI.UBI.PT
FACULDADE ENGENHARIA
Departamento de Informática

HTTP – CABEÇALHOS DE SERVIDOR

Content-Location: Uma localização alternative para os dados devolvidos Content-Range: Onde pertence a mensagem parcial dentro do corpo total

Content-Type: O tipo de MIME do conteúdo

Expires: Define a data/hora após a qual a resposta expira

Last-Modified: A última data de modificação do objeto solicitado

Location: Usado em redireção, ou quando um novo recurso foi criado

Set-Cookie: Um cookie HTTP

SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática

HTTP – CABEÇALHOS DE SERVIDOR

Exemplo

```
Accept-Ranges: bytes
Allow: GET, HEAD
Cache-Control: max-age=36001
Content-Encoding: gzip
Content-Language: da
Content-Length: 348
Content-Location: /index.htm
Content-Range: bytes 21010-47021/47022
Content-Type: text/html; charset=utf-8
Expires: Thu, 01 Dec 2015 16:00:00 GMT
Last-Modified: Tue, 15 Nov 2015 12:45:26 GMT
Location: http://www.w3.org/pub/WWW/People.html
Set-Cookie: session_id=7f...11; Domain=foo.com; Path=/; Expires=Wed, 13 ... GMT;
```


SEBASTIAO@DI.UBI.PT

FACULDADE ENGENHARIA
Departamento de Informática