

# Introdução às tecnologias Web - ITW

## Aula 1 – Introdução à WWW

# Sumário

Breve história da Internet

O protocolo http

CGI

A linguagem html – versões iniciais

Cabeçalhos, parágrafos, áreas, hiperligações, ...

Exemplos de aplicação



# Pré-História de Internet / WWW

**Vannevar Bush**, 1890-1974, foi um engenheiro, inventor e político americano, conhecido pelo seu papel político no desenvolvimento da bomba atómica e pela ideia do memex — visto como um conceito pioneiro, precursor da world wide web.

Memex (**memory + index**) – máquina visionária imaginada para auxiliar a memória e guardar conhecimentos. (As We May Think, 1945).

Constatação: o conhecimento aumenta a um ritmo prodigioso, e não encontra contrapartida na evolução dos meios de armazenamento e acesso aos dados

Observando o funcionamento da mente humana, operando sempre por meio de associações, Bush imaginou e descreveu de maneira detalhada uma máquina capaz de armazenar grandes quantidades de informação, para posterior e rápida recuperação.

Tal engenho, concebido para suprir as "falhas da memória humana" através de recursos mecânicos, é considerado o precursor do conceito de hipertexto.

# Pré-História de Internet / WWW

**Douglas C. Engelbart** (1925-2013) foi pioneiro na interação humano computador.

Baseou-se no conceito imaginado em "As we may think" de Vannevar Bush para desenvolver o hipertexto, computadores em rede e os precursores de interfaces gráficas

É conhecido por ter inventado o mouse / rato dos computadores.

# Pré-História de Internet / WWW

**Ted Nelson**, EUA, foi o “inventor” das palavras hipertexto e hipermédia  
Fundou o Projeto Xanadu em 1960 com o objetivo de criar uma rede de  
computadores de interface simples.

O trabalho está documentado no seu livro Computer Lib/Dream Machines, de 1974, e no  
livro Literary Machines, de 1981.

# História de Internet / WWW

6 de agosto de 1991 – **Tim Berners-Lee**, a trabalhar no CERN, Suíça, publica um resumo do projeto da World Wide Web no **alt.hypertext** newsgroup. Essa data marca o nascimento da Web como um serviço público da Internet;

1993 – **Marc Andreessen**, no NCSA, University of Illinois, EUA, apresenta o primeiro navegador gráfico, o **Mosaic**.

Depois da graduação Marc Andreessen associou-se a um anterior chefe da Silicon Graphics e iniciaram a comercialização do **Mosaic** que passou a chamar-se **Netscape** em Abril 1994.

# História de Internet / WWW

A **Netscape Communications Corporation** foi responsável pelo desenvolvimento dos:

- Secure Sockets Layer Protocol (SSL) – protocol para transmissão segura de dados, ainda hoje largamente utilizado;
- JavaScript – a linguagem de programação mais utilizada na criação de programas para páginas web, do lado do cliente.

A Netscape foi comprada em 1999 por 10MM\$ pela AOL;

# História de Internet / WWW

Antes da venda da Netscape Communications Corporation à AOL, foi disponibilizado o código fonte e criada a **Mozilla Organization** para manter o desenvolvimento futuro do produto.

A Mozilla Organization reescreveu todo o código do browser e denominou esse desenvolvimento **Gecko engine**;

Todas as versões do Netscape até ao seu encerramento foram baseadas no Gecko. O Gecko engine foi posteriormente utilizado para o browser **Firefox** – da Mozilla Foundation.

# História de Internet / WWW

Windows Internet Explorer é o browser da Microsoft desde 23 de agosto de 1995.

Em 2 de setembro de 2008 saiu a primeira versão beta do browser Chrome da Google. Em 11 de dezembro de 2008 foi lançada uma versão estável ao público em geral.

Está disponível em mais de 51 idiomas para as plataformas Windows, Mac OS X, Android, Ubuntu, Debian, Fedora e OpenSuSE.

Em 17 de Março de 2015, a Microsoft anunciou que o Microsoft Edge substituirá o Internet Explorer no Windows 10 devices.

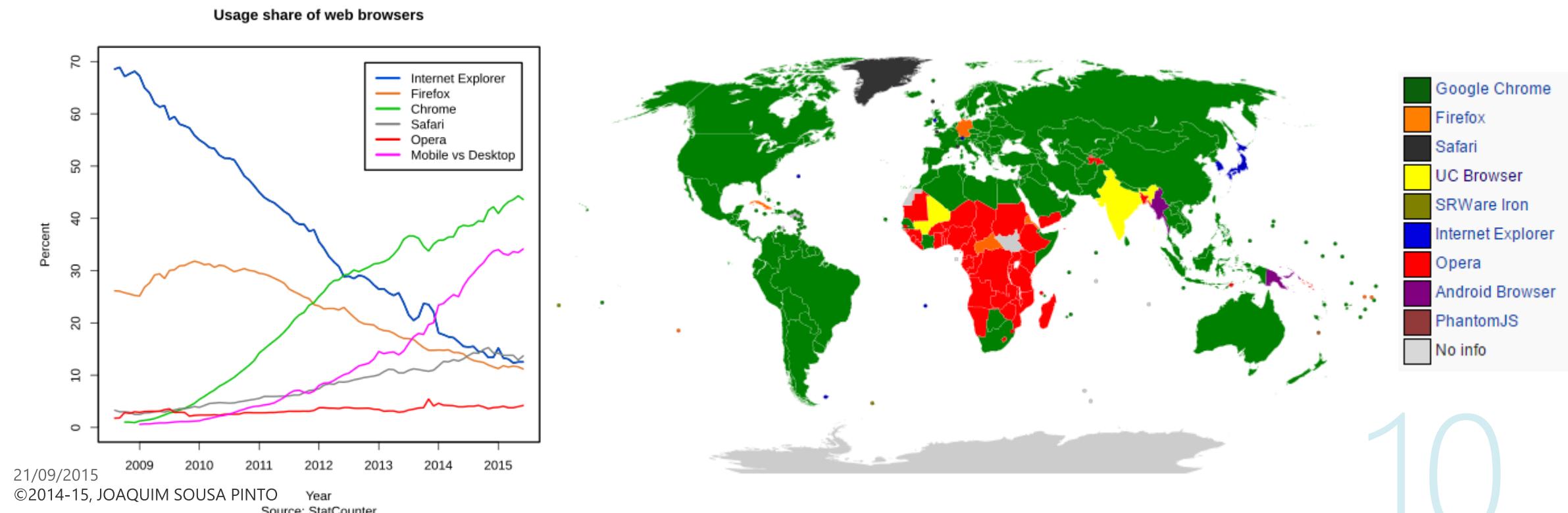
Assim, o Internet Explorer 11 foi a última versão deste browser.

# História de Internet / WWW

## Estatísticas da “guerra dos browsers”

O IE foi durante muitos anos hegemónico vindo, no entanto a perder cota de mercado para o Google Chrome da Google.

O Chrome assumiu a liderança em maio de 2012, de acordo com o site StatsCounter.



# História de Internet / WWW

## Os pioneiros - resumos

Tim Berners Lee

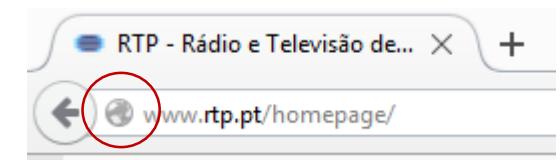
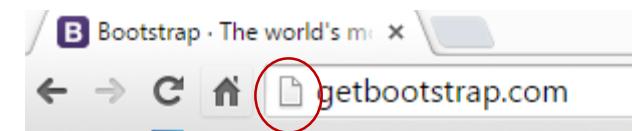
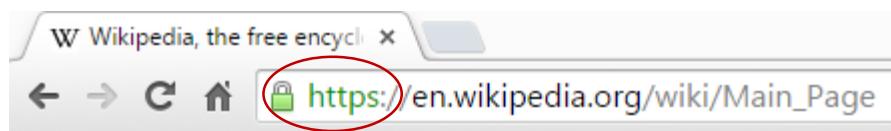
- Criador e ainda atual mentor da WWW
- Uniform Resource Locator (URL),
- Hypertext Transfer Protocol (HTTP),
- Hypertext Markup Language (HTML)

Marc Andreessen

- Criador do Mosaic
- Fundador da Netscape Communications Corporation

# Hypertext Transfer Protocol (HTTP)

Protocolo utilizado para transferir documentos de hipertexto e seus recursos de máquinas remotas.



# Hypertext Transfer Protocol (HTTP)

## Características gerais

Protocolo da camada de aplicação;

Modelo de funcionamento baseado em pedido-resposta;

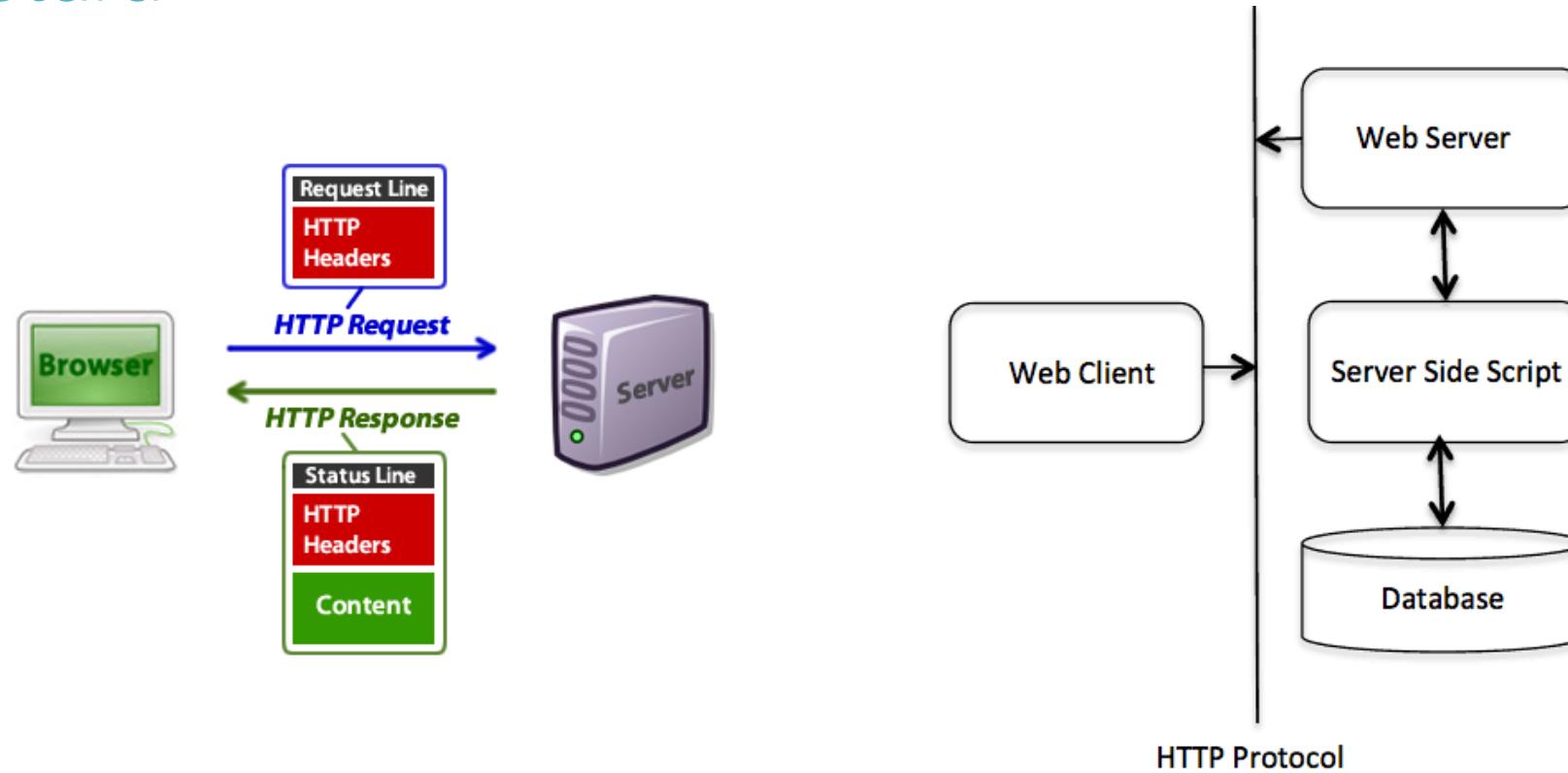
Cabeçalho das mensagens é texto puro (não binário);

Não orientado a conexões;

Não guarda estado entre conexões distintas, isto é, cada conexão é nova para o servidor.

# Hypertext Transfer Protocol (HTTP)

## Arquitetura



# Hypertext Transfer Protocol (HTTP)

## Sintaxe geral de uma URL

**<protocolo>://<servidor>:<porta>/<caminho>/<recurso>?dados**

- Protocolo: http / https / ...
- A porta é opcional para serviços em portas default
  - para o http a porta default é a 80;
  - Para o https a porta default é a 443.
- Caminho e recurso podem ser omitidos (URLs parciais)
- As URLs podem conter dados depois do nome do recurso

# Hypertext Transfer Protocol (HTTP)

## Exemplos de URL's

<http://www.ua.pt>

Neste exemplo de url parcial não se incluiu a porta default (80, para o http) nem o caminho/recurso.

O recurso por default é variável e dependente do tipo de servidor e do tipo de linguagem de programação utilizada no desenvolvimento do servidor.

Nomes típicos: index.htm / index.html / default.htm / default.html / default.asp / default.aspx / default.php / ...

<http://www.ua.pt/deti/PageCourse.aspx?id=383>

Neste exemplo a URL possui dados depois do recurso, neste caso, id=383.

Pergunta: Que representa esta URL?

# Hypertext Transfer Protocol (HTTP)

## Exemplos de URL's

<http://www.ua.pt/deti/PageCourse.aspx?id=383&p=4&a=9>

Neste exemplo há 3 parâmetros / dados depois no nome do recurso  
 $id=383, p=4, a=9$

*No primeiro separador é usado o ponto de interrogação e nos demais á utilizado o “i comercial / ampersand”*

Pergunta: como enviar o símbolo “&” nos dados, como no exemplo seguinte, sem baralhar o servidor?:

*Empresa= Dias & Dias A. Boavida*

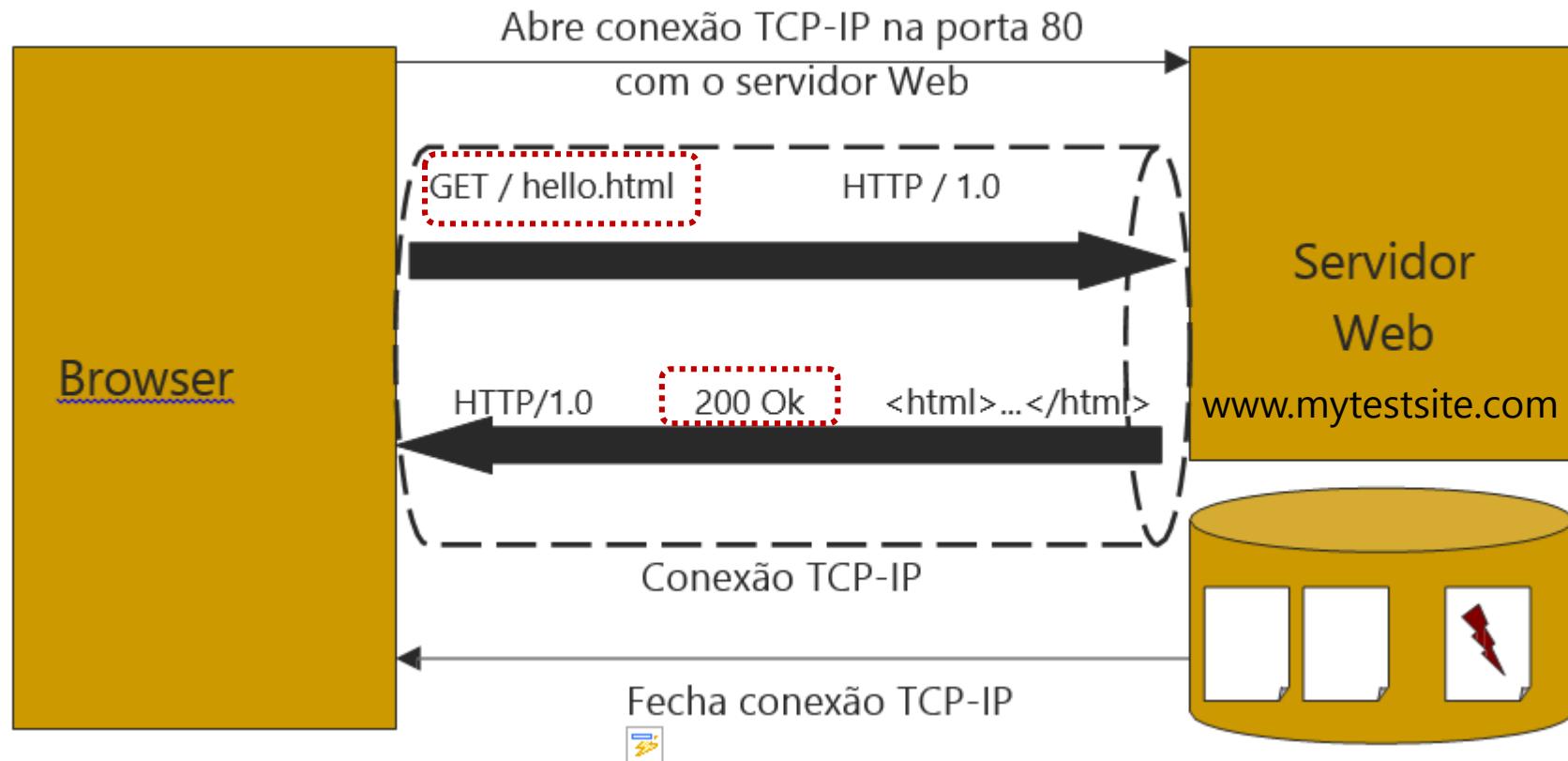
<http://localhost:6067/Specie?Shell ID=265>

Neste exemplo o nome do servidor é o “localhost”, o que significa que o servidor está no computador do próprio utilizador (*local host*).

Neste exemplo o servidor está a responder a pedidos na porta 6067.

# Hypertext Transfer Protocol (HTTP)

## Exemplo (<http://www.mytestsite.com/hello.htm> )



# Hypertext Transfer Protocol (HTTP)

## Pedido e resposta a um servidor web

```
GET /hello.htm HTTP/1.1
User-Agent: Mozilla/4.0 (compatible; MSIE 5.01; Windows NT)
Host: www.mytestsite.com
Accept-Language: en-us
Accept-Encoding: gzip, deflate
Connection: Keep-Alive
```

Pedido e resposta com sucesso

```
HTTP/1.1 200 OK
Date: Mon, 27 Jul 2009 12:28:53 GMT
Server: Apache/2.2.14 (Win32)
Last-Modified: Wed, 22 Jul 2009 19:15:56 GMT
ETag: "34aa387-d-1568eb00"
Vary: Authorization,Accept
Accept-Ranges: bytes
Content-Length: 88
Content-Type: text/html
Connection: Closed

<html>
<body>
<h1>Hello, World!</h1>
</body>
</html>
```

# Hypertext Transfer Protocol (HTTP)

## Pedido e resposta a um servidor web

```
GET /t.htm HTTP/1.1
User-Agent: Mozilla/4.0 (compatible; MSIE5.01; Windows NT)
Host: www.mytestsite.com
Accept-Language: en-us
Accept-Encoding: gzip, deflate
Connection: Keep-Alive
```

Pedido e resposta com erro

```
HTTP/1.1 404 Not Found
Date: Sun, 18 Oct 2012 10:36:20 GMT
Server: Apache/2.2.14 (Win32)
Content-Length: 230
Connection: Closed
Content-Type: text/html; charset=iso-8859-1

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html>
<head>
  <title>404 Not Found</title>
</head>
<body>
  <h1>Not Found</h1>
  <p>The requested URL /t.html was not found on this server.</p>
</body>
</html>
```

# Hypertext Transfer Protocol (HTTP)

## Pedido e resposta a um servidor web – Firebug analysys

The screenshot shows a web browser window with the following details:

- Tab bar: Conchas de Moçambique
- Address bar: localhost:6067/Specie?Shell\_ID=265
- Toolbar: Back, Forward, Stop, Refresh, Google search icon
- Header menu: CdM, Home, Species, Families, SubFamilies, Classes
- Main content: Specie detail

The screenshot shows the Firebug Net panel with the following details:

- Toolbar: All, HTML, CSS, JavaScript, XHR, Images, Plugins, Media, Fonts
- Table headers: URL, Status, Domain, Size, Remote IP, Timeline
- Table data:

URL	Status	Domain	Size	Remote IP	Timeline
GET Specie?Shell_ID=265	200 OK	localhost:6067	11.0 KB	[:]6067	426ms
POST abort?transport=webSocke..	200 OK	localhost:2776	0 B	[:]2776	14ms
GET jquery-ui.css	200 OK	localhost:6067	7.2 KB	[:]6067	17ms
GET magnific-popup.css	200 OK	localhost:6067	2.4 KB	[:]6067	28ms
GET bootstrap.min.css	200 OK	localhost:6067	24.8 KB	[:]6067	42ms
GET bootstrap-theme.css	200 OK	localhost:6067	3.8 KB	[:]6067	36ms
GET App.css	200 OK	localhost:6067	1.1 KB	[:]6067	39ms
GET ie-emulation-modes-warning.j	200 OK	localhost:6067	2.2 KB	[:]6067	49ms
GET MsAjaxJs?v=D6VN0fHlwFSIW..	200 OK	localhost:6067	44.9 KB	[:]6067	233ms
GET jquery-2.1.1.js	200 OK	localhost:6067	241.6 KB	[:]6067	108ms
GET jquery-ui-1.11.1.js	200 OK	localhost:6067	453.5 KB	[:]6067	130ms
GET WebFormsJs?v=N8tymL9KraM	200 OK	localhost:6067	19.6 KB	[:]6067	170ms
GET ...	200 OK	localhost:6067	71.1 KB	[:]6067	107ms

# Hypertext Transfer Protocol (HTTP)

## Pedidos ao servidor em modo GET

Nos exemplos até aqui apresentados os “dados” foram sempre enviados de modo embebido na URL.

Este método de trabalho é conhecido como envio de dados por GET.

Este método de trabalho é o mais simples porque, para a sua utilização, basta introduzir os dados na linha de URL do browser.

Tem no entanto graves limitações:

*O limite considerado “razoável” para um URL é de <2000 caracteres;*

*Os dados vão expostos e são capturados pelos sniffers de rede;*

*Ficam armazenados nas log files do servidor, porque armazenam a URL completa;*

*[Não recomendável, por exemplo para passagem de credenciais de acesso].*

# Hypertext Transfer Protocol (HTTP)

## Pedidos ao servidor em modo POST

Há outro método em que os dados são enviados ao servidor sem ser pela linha da URL. Esse método é conhecido como método POST.

### Exemplo comparativo entre headers GET e POST

```
GET /index.html?Nome=NomePessoa&Idade=99&Curso=Informática HTTP/1.1  
Host: www.exemplo.com
```

```
POST /index.html HTTP/1.0  
Accept: text/html  
If-modified-since: Sat, 29 Oct 1999 19:43:31 GMT  
Content-Type: application/x-www-form-urlencoded  
Content-Length: 41
```

```
Nome=NomePessoa&Idade=99&Curso=Informática
```

# Common Gateway Interface (CGI)

Para que serve?

- Produção de documentos com conteúdo dinâmico (normalmente extraído de SGBD's)
- Guardar dados em ficheiros ou SGBDs do lado do servidor

Características da execução de um comando CGI:

- O programa é executado do lado do servidor e gera um ficheiro (normalmente) html com o resultado da execução;
- Pode ser escrito em qualquer linguagem de programação;
- Cada pedido feito cria um novo processo idêntico na memória do servidor (ineficiente)

# Common Gateway Interface (CGI)

## Exemplos

<http://www.mytestsite.com/getcounter.cgi>

Possível script  
Desenvolvido em perl

```
#!/usr/bin/perl

open(IN,"counts");
flock(IN,2);                      # lock the file
seek(IN,0,0);                     # rewind it to the beginning

$count = <IN>;                   # read only the first line.
close(IN);

$count = $count + 1;

open(OUT,>"counts");
flock(OUT,2);                      # lock the file
seek(OUT,0,0);                     # rewind it to the beginning
print OUT "$count\n";
close(OUT);

print "Content-type:text/html\n\n";
print "You are visitor number $count.<p>\n";
```

# A linguagem html

HTML (abreviatura da expressão inglesa “HyperText Markup Language”, que significa “Linguagem de Marcação de Hipertexto”)

É uma linguagem de marcação utilizada para produzir páginas na Web.

É uma tecnologia baseada na linguagem SGML.

SGML é um padrão de formatação de textos.

Não foi desenvolvido para hipertexto, mas tornou-se conveniente para transformar documentos em hiper-objetos e para descrever as ligações.

Exemplo de marcador SGML:

```
<marcador>texto</marcador>
```

# Versões da linguagem html

November 24, 1995 - HTML 2.0 – IETF RFC 1866.

November 25, 1995: RFC 1867 (form-based file upload)

May 1996: RFC 1942 (tables)

August 1996: RFC 1980 (client-side image maps)

January 1997: RFC 2070 (internationalization)

January 1997 – HTML 3.2 published as a W3C Recommendation.

It was the first version developed and standardized exclusively by the W3C, as the IETF

December 1997 – HTML 4.0 published as a W3C Recommendation

# Versões da linguagem html

April 1998 – HTML 4.0, reissued with minor edits without incrementing the version number.

December 1999 – HTML 4.01, published as a W3C Recommendation.

Development of the parallel, XML-based language XHTML occupied the W3C's HTML Working Group through the early and mid-2000s.

As of mid-2008 – HTML 4.01, ISO/IEC 15445:2000.

December 2012 – HTML5, candidate recommendation of the World Wide Web Consortium.

October 2014 – HTML5 was published as a W3C Recommendation



# Características de um elemento html

Um marcador / etiqueta (tag em inglês) num documento html fica sempre colocada entre os símbolos "<" e ">". As etiquetas são responsáveis pela formatação da linguagem

Nas últimas versões da linguagem, um elemento é constituídos por um par de início e de fim de marcador.

```
<marcador>texto</marcador>
```

marcador de  
início do  
elemento

Marcador de fim do  
elemento

# Características de um elemento

Um elemento é formado por um nome de etiqueta (tag), atributos, valores e filhos (que podem ser outros elementos ou texto). Os atributos modificam os valores padrão dos elementos e os valores caracterizam essa mudança.

Exemplo de um elemento simples (não possui filhos):

```
<hr/>
```

Exemplo de um elemento com atributos:

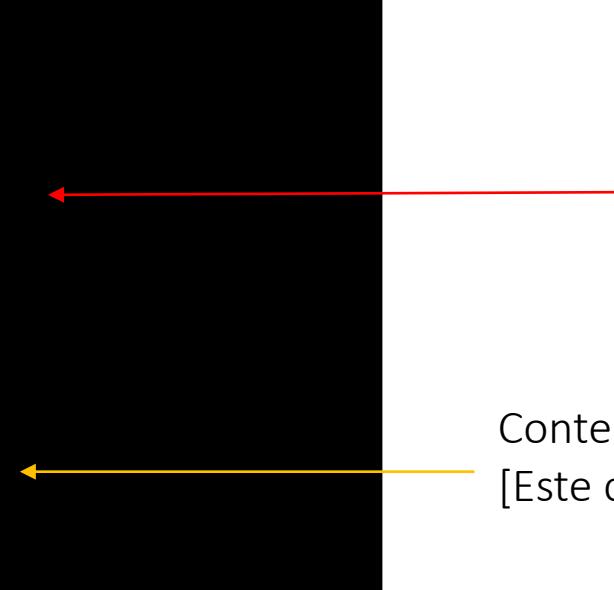
```
<a href="http://www.ua.pt/">Universidade de Aveiro</a>
```

Exemplo de um elemento com filhos e atributos:

```
<p>A <a href="http://www.ua.pt/">Universidade de Aveiro</a> é a  
minha Universidade.</p>
```

# Estrutura base de um documento html

```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta name="Author" content="">
    <meta name="Keywords" content="">
    <meta name="Description" content="">
    <title>Document</title>
  </head>
  <body>
    </body>
</html>
```



Cabeçalho do documento  
[Este conteúdo não é representado]

Conteúdo do documento  
[Este conteúdo é representado]

# Elementos base de um documento html

Os elementos básicos de um documento html são:

**<html>** – define o início de um documento html e indica ao navegador que todo conteúdo posterior a este marcador deve ser tratado como uma série de marcadores html.

*Neste marcador pode ter como atributo a língua em que o documento está escrito. Essa informação é importante para os motores de pesquisa / indexação.*

**<head>** – define o cabeçalho do documento e possui informações sobre o documento que vai ser representado;

**<body>** – define o conteúdo principal do documento e é a parte que é exibida no navegador. No marcador do corpo podem-se definir atributos comuns a toda a página, como cor de fundo, margens, e outras formatações.

# Elementos do cabeçalho de um documento html

Dentro do cabeçalho podemos ter os seguintes elementos:

**<title>** – define o título da página; é exibido na barra de título do browser;

**<style type="text/css">** – define a formatação das etiquetas em CSS;

**<script type="text/javascript">** – define a programação de certas funções da página escritas na linguagem JavaScript;

**<link>** – define ligações da página com outros arquivos, como por exemplo, feeds, CSS, scripts, etc.

**<meta>** – define propriedades da página, como, por exemplo, a codificação dos caracteres (UTF-8, ISO-8859-1, ...), a descrição da página (autor, keywords, descrição – ver "dublin core metadata")

# Etiquetas do corpo de um documento html

`<h1>, <h2>, ... <h6>` – define os cabeçalhos e títulos no documento em diversos tamanhos;

`<p>` – define um novo parágrafo;

`<br />` – impõe uma quebra de linha num texto;

`<table>` – cria uma tabela

As linhas são criadas com o marcador `<TR>`

As células são criadas com o marcador `<TD>`

Os cabeçalhos das colunas são criados com as etiquetas `<THead><TH>`

Os rodapés com `<TFooter><TR><TD>`.

# Etiquetas do corpo de um documento html

`<div>`,`<span>` – determinam uma divisão na página a qual pode possuir variadas formatações.

`<img />` – Insere uma imagem.

`<a>` – cria uma hiperligação para um outro local, seja uma página, um e-mail ou outro serviço.

`<textarea>` – delimita uma caixa de texto (com mais de uma linha); estas caixas de texto são muito usadas para a edição de textos grandes e com formatação (mudanças de linha).

`<b>`, `<i>`, `<u>` e `<s>` – determina diversas formatações de letra: negrito, itálico, sublinhado e riscado, respectivamente.

# Exemplos de hiperligações

Ligaçāo para outro site internet:

```
<a href="https://pt.wikipedia.org/">Ligaçāo à Wikipedia!</a>
```

Ligaçāo para outra página do mesmo site:

Estando dentro do site [www.ua.pt](http://www.ua.pt) e pretendendo aceder a página do curso da LEI ...

```
<a href="/deti/PageCourse.aspx?id=383">Licenciatura em Engenharia Informática</a>
```

# Exemplos da utilização de imagens

## Utilização de uma imagem:

```

```

Parâmetros:

- src= localização da imagem
- alt = texto apresentado nos browsers de texto ou quando as imagens não são apesentadas
- title = texto que é mostrado quando o rato passa sobre a imagem
- width = largura da imagem
- height = altura da imagem

# Exemplos de tabelas

Tabela simples com duas linhas (`<tr>`) e 3 colunas (`<td>`)

```
<table style="width:100%">
  <tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
</table>
```

Fonte: [http://www.w3schools.com/html/html\\_tables.asp](http://www.w3schools.com/html/html_tables.asp)

# Exemplos de tabelas

Exemplo de tabela com 2 linhas (`<tr>`) sendo a primeira com cabeçalhos (`<th>`) e a segunda com dados (`<td>`)

```
<table style="width:100%">
  <tr>
    <th>Firstname</th>
    <th>Lastname</th>
    <th>Points</th>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
</table>
```

Fonte: [http://www.w3schools.com/html/html\\_tables.asp](http://www.w3schools.com/html/html_tables.asp)

# Exemplos de tabelas

Exemplo de uma tabela com uma coluna colapsada.

Note que a 1<sup>a</sup> linha possui apenas 2 colunas enquanto a 2<sup>a</sup> linha possui 3 colunas.

Na primeira linha, as 2<sup>a</sup> e 3<sup>a</sup> colunas foram unidas (atributo colspan="2") numa única.

```
<table style="width:100%">
  <tr>
    <th>Name</th>
    <th colspan="2">Telephone</th>
  </tr>
  <tr>
    <td>Bill Gates</td>
    <td>555 77 854</td>
    <td>555 77 855</td>
  </tr>
</table>
```

Fonte: [http://www.w3schools.com/html/html\\_tables.asp](http://www.w3schools.com/html/html_tables.asp)

# Exemplos de tabelas

Exemplo de uma tabela com uma coluna colapsada.

Note que a 3<sup>a</sup> linha possui somente uma coluna enquanto as duas anteriores possuem duas.

Na 1<sup>a</sup> coluna da 2<sup>a</sup> linha a atributo rowspan="2" une esta célula com a seguinte.

```
<table style="width:100%">
  <tr>
    <th>First Name:</th>
    <td>Bill Gates</td>
  </tr>
  <tr>
    <th rowspan="2">Telephone:</th>
    <td>555 77 854</td>
  </tr>
  <tr>
    <td>555 77 855</td>
  </tr>
</table>
```

Fonte: [http://www.w3schools.com/html/html\\_tables.asp](http://www.w3schools.com/html/html_tables.asp)

# Exemplos de tabelas

O elemento <caption> introduz uma legenda na tabela

```
<table style="width:100%">
  <caption>Monthly savings</caption>
  <tr>
    <th>Month</th>
    <th>Savings</th>
  </tr>
  <tr>
    <td>January</td>
    <td>$100</td>
  </tr>
  <tr>
    <td>February</td>
    <td>$50</td>
  </tr>
</table>
```

Fonte: [http://www.w3schools.com/html/html\\_tables.asp](http://www.w3schools.com/html/html_tables.asp)

# Aulas práticas

Desenvolvimento de pequenos exercícios integradores que permitirão, ao longo do semestre conduzir os estudantes até ao objetivo final:

Desenvolver uma aplicação móvel para dispositivos Android baseada em páginas html.

## Pergunta?

Porque “apenas” para dispositivos Android e não para outro tipo de dispositivos?

*Porque para a criação de aplicações para dispositivos Android o desenvolvimento é grátis (free).*

*Para dispositivos Windows ou iPad/iPhone, há custos associados com o desenvolvimento.*

E essa aplicação pode ser colocada no meu telemóvel?

*Sim, sem dúvida!*

# Aulas práticas

## Que ferramentas são precisas?

Editor de texto

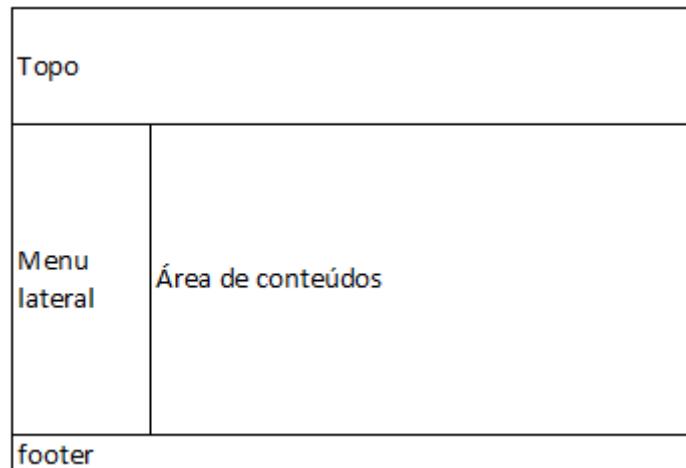
Mínimo: Notepad

Máximo: Microsoft Visual Studio (<http://dreamspark.ua.pt>)

# Exercício 1

Usando tabelas, crie uma página (index.html) que servirá de estrutura base para um site web.

O formato da página deverá ser o seguinte:



Não se esqueça de ir  
guardando o seu trabalho  
repetidas vezes!!!!

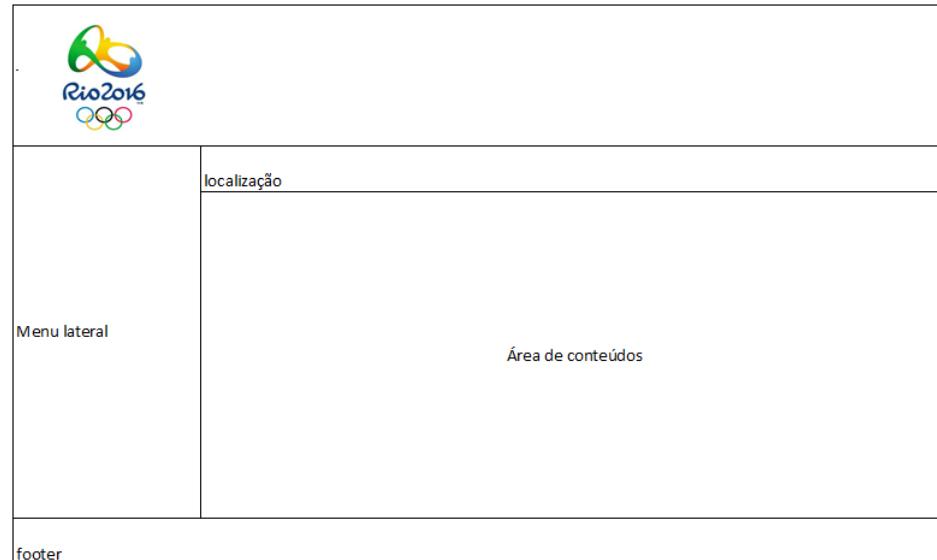
# Exercício 2

Escolha uma imagem ([images.google.com](http://images.google.com)) para colocar como logótipo do seu web site.

Coloque-a à esquerda na área de topo

Exemplo: <http://www.logotipo.pt/blog/wp-content/uploads/2011/01/logotipo-jogos-rio-2016.jpg>

Não se esqueça de ir  
guardando o seu trabalho  
repetidas vezes!!!!



# Exercício 3

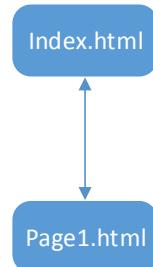
Escreva um texto com diversos parágrafos e dê corpo ao seu documento.

Nota: para que não perca o tempo todo a “imaginar” o texto a escrever, copie por exemplo um texto de um jornal ou recorra a um sítio na internet com textos aleatórios (loren ipsum sites)

<http://pt.lipsum.com/>

<http://hipsum.co/>

Crie outra página (page1.html) e interligue as duas páginas com hiperligações colocadas na área “Menu lateral”.



Não se esqueça de ir  
guardando o seu trabalho  
repetidas vezes!!!!