

# Aplicações para Internet



Cruzeiro do Sul Virtual  
Educação a Distância



# Material Teórico



## Conceitos Básicos do HTML

### **Responsável pelo Conteúdo:**

Prof. Ms. Alexander Gobbato Paulino Albuquerque

### **Revisão Textual:**

Profa. Ms. Fátima Furlan





- Conceitos Básicos de HTML
- Introdução as Tags de HTML
- Explicação do Conceito de Cliente/Servidor
- Ferramentas disponíveis



### OBJETIVO DE APRENDIZADO

Conhecer:

- Conceitos Básicos de Internet e sua História;
- Tecnologias atuais e emergentes para desenvolvimento de aplicações para Internet;
- Conceitos básicos da arquitetura cliente-servidor.
- O que é o HTML?
- O W3C e suas especificações
- Estruturação e conteúdo
- Corpo básico





# Orientações de estudo

Para que o conteúdo desta Disciplina seja bem aproveitado e haja uma maior aplicabilidade na sua formação acadêmica e atuação profissional, siga algumas recomendações básicas:



## Assim:

- ✓ Organize seus estudos de maneira que passem a fazer parte da sua rotina. Por exemplo, você poderá determinar um dia e horário fixos como o seu “momento do estudo”.
- ✓ Procure se alimentar e se hidratar quando for estudar, lembre-se de que uma alimentação saudável pode proporcionar melhor aproveitamento do estudo.
- ✓ No material de cada Unidade, há leituras indicadas. Entre elas: artigos científicos, livros, vídeos e sites para aprofundar os conhecimentos adquiridos ao longo da Unidade. Além disso, você também encontrará sugestões de conteúdo extra no item **Material Complementar**, que ampliarão sua interpretação e auxiliarão no pleno entendimento dos temas abordados.
- ✓ Após o contato com o conteúdo proposto, participe dos debates mediados em fóruns de discussão, pois irão auxiliar a verificar o quanto você absorveu de conhecimento, além de propiciar o contato com seus colegas e tutores, o que se apresenta como rico espaço de troca de ideias e aprendizagem.

# Introdução



## O que preciso saber para desenvolver aplicações para a WEB?

- Entender o funcionamento básico da execução dessa aplicação
- Conhecer as linguagens que são executadas na máquina cliente (HTML, CSS, JavaScript)
- Conhecer alguma linguagem que rode no servidor para aplicações mais robustas (PHP, ASPX, JSP, etc)
- Saber trabalhar com banco de dados (MySQL, Oracle, SQLServer, etc)
- Ter um pouco de criatividade para se trabalhar com os layouts das suas aplicações



A Evolução da WEB: <https://goo.gl/F090S>

# Ambiente Cliente/Servidor

Pensando nos elementos básicos para fornecer informação através da Web, temos:

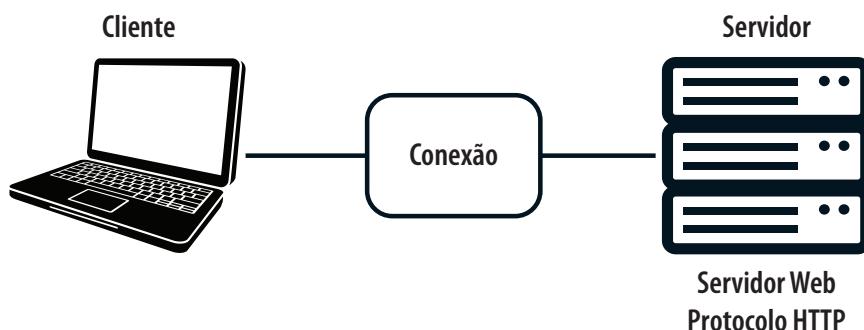


Figura 1

Quando falamos em desenvolvimento web podemos ter duas vertentes, podemos programar *client side* e/ou *server side*.

*Client Side* ou “Lado Cliente” são programas executados na máquina cliente, ou seja, os browsers são os responsáveis pela execução/interpretação das linguagens.



Figura 2

*Server Side* ou “Lado Servidor” são programas e instruções executadas em uma máquina servidor, é necessário um servidor, por exemplo, IIS da *Microsoft*, *GlassFish* e *Apache* que suportem os tipos de linguagens utilizadas, como *ASPX*, *JSP* e *PHP*. Essas linguagens são compiladas e gerados arquivos que somente os computadores podem interpretar. No servidor ainda podemos ter os bancos de dados que são softwares responsáveis por todo armazenamento de informações em estrutura de tabelas e relacionadas entre si.

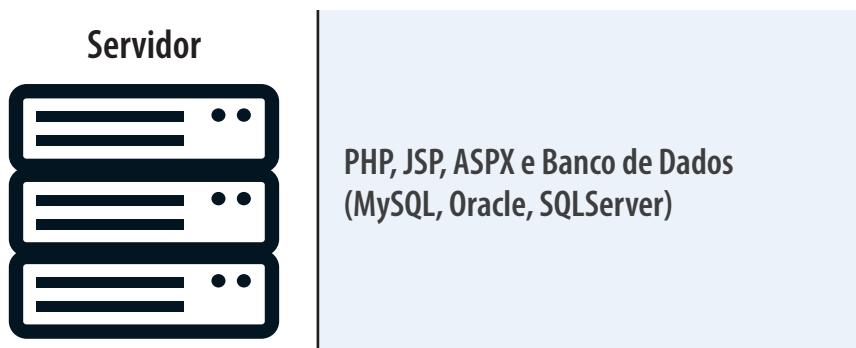


Figura 3

Então em resumo, podemos trabalhar tanto com programações cliente e/ou programações no servidor.

## O que é HTML?

O HTML (*Hypertext Markup Language*) é conhecido como uma Linguagem de Marcação de Hipertexto. O HTML é uma linguagem que se propõe em distribuir informação globalmente e pode ser entendido por diversos meios de acesso, ele foi desenvolvimento por Tim Bernes-Lee .

Um documento HTML é um documento texto que pode ser produzido utilizando qualquer editor de texto. O conteúdo HTML de uma página é processado por um Navegador (Internet Explorer, Chrome etc.)

O HTML define um conjunto de elementos para a marcação de uma página Web: cabeçalho, parágrafo, lista, tabelas, entre outros. Cada elemento possui sua função específica e são comumente chamados de tag (marca ou marcadores).

<b>Ferramentas:</b>	Para testar nossos arquivos: - Internet Explorer (última versão) - Mozilla Firefox (última versão) - Google Chrome (última versão)
---------------------	---



Figura 4

Para criar nossos códigos podemos utilizar qualquer editor de texto puro ou algumas ferramentas próprias para o desenvolvimento Web

- Microsoft Expression Web 4 ou
- **Brackets** ou
- Bloco de notas ou
- NetBeans ou
- **Notepad++**

## Brackets

Esta ferramenta é gratuita (desenvolvida pela Adobe), para baixar acesse o link <http://brackets.io/> (existe uma versão portable)

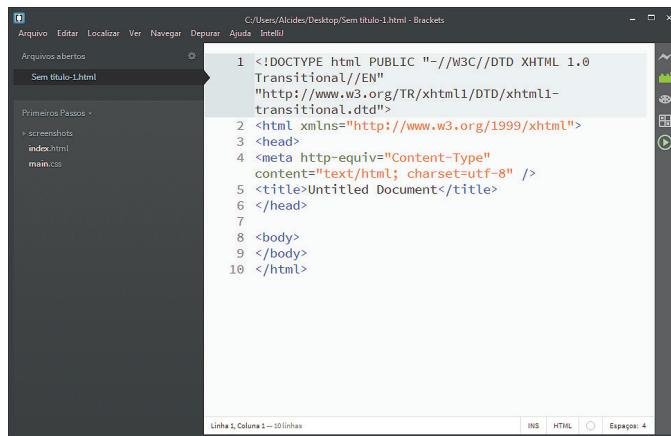


Figura 5

## Notepad ++

Esta ferramenta é gratuita, para baixar acesse o link (existe uma versão portable):



Downloads | Notepad++: <http://bit.ly/2v7BZQG>

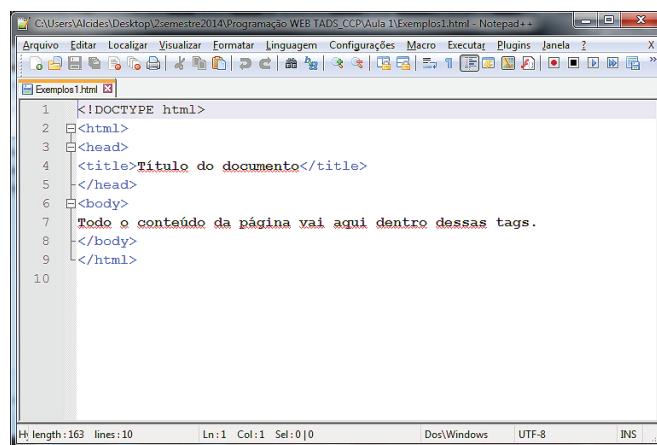


Figura 6

## Bloco de Notas

Ferramenta que já vem com o sistema Windows, muito simples de utilizar mas não ajuda em nada a criação do código.

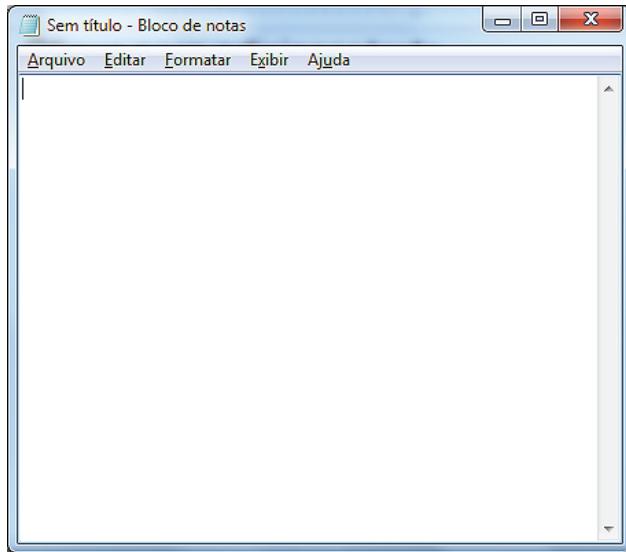


Figura 7

## Versões

A versão mais significativa para nossa realidade foi o HTML 4.01 que foi publicado como uma recomendação do W3C em 1999.

- » **HTML 5** - sua especificação começou em 2008 e está praticamente finalizada, porém a **W3C** ainda está analisando algumas tags.



W3C Brasil: <https://goo.gl/VHQ5n>

## Sobre a W3C

O Consórcio *World Wide Web* (W3C) é um consórcio internacional no qual organizações filiadas, uma equipe em tempo integral e o público trabalham juntos para desenvolver padrões para a Web. Liderado pelo inventor da web Tim Berners-Lee e o CEO Jeffrey Jaffe, o W3C tem como missão Conduzir a *World Wide Web* para que atinja todo seu potencial, desenvolvendo protocolos e diretrizes que garantam seu crescimento de longo prazo.



Para mais informações, contate: <https://goo.gl/5JL1jK>

# Marcas (tags)

Todo documento HTML possui marcadores, conhecidos e chamados como tags , que são palavras inseridas entre os parênteses angulares (< e >), esses marcadores são os comandos de formatação da linguagem. Um elemento é formado por um nome de marcador (tag), atributos, valores e filhos.

## Exemplo de elemento único:

```
<hr />
```

## Exemplo de elemento composto:

```
<a href="http://www.cruzeirodosul.edu.br/">Universidade Cruzeiro do Sul</a>
```

- » **<a>** é o marcador de abertura
- » **</a>** é o marcador de fechamento
- » **href** é o atributo onde está definido para qual página será direcionado ao receber o evento do clique.

# Estrutura Básica de um Documento

A estrutura básica de um documento HTML possui as seguintes informações:

```

1  <html>
2    <head>
3      <meta charset="utf-8" />
4      <meta name="description" content="a descrição do seu site">
5      <meta name="keywords" content="palavras-chaves curtas">
6      <title>Título do Documento</title>
7    </head>
8    <body>
9      <!--
10        Aqui fica a página que será visível para todos, onde pode-se inserir
11        textos, imagens, links para outras páginas, etc, geralmente usa-se:
12        -->
13
14      <div>Tag para criar um bloco, mais utilizada em "Cascading Style Sheets
15          (Folhas de Estilo em Cascata)</div>
16
17      <span>Tag para modificação de uma parte do texto da página</span>
18
19      
20
21      <a href="http://www.cruzeirodosul.edu.br/">Universidade Cruzeiro do Sul</a>
22    </body>
23 </html>

```

Figura 8

Os marcadores HTML não são case sensitive, o que isso quer dizer?

Quer dizer que tanto faz escrevermos em maiúscula ou minúscula que o navegador irá “entender”; <HTML>, <Html>, <html> ou <HtMl>.

Existem vários marcadores, mas para se construir uma página é essencial e recomendado que as seguintes tags estejam inseridas no documento.

### Os marcadores básicos são:

- » **<html>**: define o início de um documento HTML e indica ao navegador que todo conteúdo posterior deve ser tratado como uma série de códigos HTML.
- » **<head>**: define o cabeçalho de um documento HTML, que traz informações sobre o documento que está sendo aberto.
- » **<body>**: define o conteúdo principal, o corpo do documento. Esta é a parte do documento HTML que é exibida no navegador. No corpo podem-se definir atributos comuns a toda a página, como cor de fundo, margens, e outras formatações.

### Cabeçalho

- » **<title>**: define o título da página, que é exibido na barra de título de qualquer navegador.
- » **<style type="text/css">**: define formatação, para isso se utilizada marcadores de estilos chamado de css ou folha de estilo
- » **<script type="text/javascript">**: define programação de certas funções em página com scripts.
- » **<link>**: define ligações da página com outros arquivos.
- » **<meta>**: define propriedades da página, como codificação de caracteres, descrição da página, autor etc.

### Corpo

Dentro do corpo podemos encontrar outros vários marcadores que irão estruturar o documento, todo elemento que será exibido no browser deve ser inserido na tag <body> por exemplo, vejamos algumas tags.

- » **<br />**: quebra de linha.
- » **<table>**: cria uma tabela (linhas são criadas com <TR> e novas células com <TD>, já os cabeçalhos das colunas são criados com os marcadores <Thead><TH> e os rodapés com <TFooter><TR><TD>).
- » **<div>**: determina uma divisão na página a qual pode possuir variadas formatações.
- » **<img />**: imagem.
- » **<a>**: hiper-ligação para um outro local, seja uma página, um e-mail ou outro serviço.
- » **<textarea>**: caixa de texto (com mais de uma linha); estas caixas de texto são muito usadas em blogs, elas podem ser auto selecionáveis e conter outros códigos a serem distribuídos.

## Estrutura de um Documento utilizando Tags do HTML5

```

1  <!DOCTYPE html>
2
3  <html>
4      <head>
5          <title>Tutorial de HTML5</title>
6          <meta name="description" content="Como usar as tags header, footer e article em HTML5">
7          <meta charset="utf-8">
8          <link rel="stylesheet" type="text/css" href="estilo.css"/>
9          <script type="text/javascript" src="programacao.js"></script>
10     </head>
11
12     <body>
13         <header>
14             <h1>Título I</h1>
15             <h2>Título II</h2>
16         </header>
17
18         <article>
19             <header>
20                 <h3>Título III</h3>
21                 <p>Linha 1<br /> Linha 2</p>
22             </header>
23
24             <footer>
25                 <p>Aplicações para Internet</p>
26             </footer>
27         </article>
28
29         <footer>
30             <p>Todos os direitos reservados. Universidade Cruzeiro do Sul</p>
31         </footer>
32     </body>
33
34 </html>

```

Figura 9

- » **<!DOCTYPE html>**: Especifica quais são as regras para a linguagem de marcação, para que os navegadores processem o conteúdo corretamente. Nesse caso estamos informando que utilizaremos uma marcação para hmt5

Observem que dentro das tags <head> temos as tags <title>, <meta>, <link> e <script>

```

<head>
    <title>Tutorial de HTML5</title>
    <meta name="description" content="Como usar as tags header, footer e article em HTML5">
    <meta charset="utf-8">
    <link rel="stylesheet" type="text/css" href="estilo.css"/>
    <script type="text/javascript" src="programacao.js"></script>
</head>

```

Figura 10

- » **<title>**: define o título da página, que é exibido na barra de título de qualquer navegador.
- » **<meta>**: define propriedades da página, como codificação de caracteres, descrição da página, autor etc. O atributo name= "description" e content são utilizados para adicionarmos informações pertinentes a página, essas informações podem ser utilizados pelos mecanismos de busca para então

retornar o resultado de pesquisa. O atributo charset é utilizado para a configuração de caractere, muitos países não possuem caracteres especiais como acento agudo, circunflexo. O UTF-8 trata justamente esses itens, ou seja, se a sua página for acessada no Japão, os caracteres com acentos não serão trocadas por outra letra qualquer.

- » **<link>**: define ligações da página com outros arquivos. Nesse caso, a página html está fazendo um ligação com outra página de formatação de estrutura, conhecida com css.
- » **<script>**: define programação de certas funções em página com scripts. No caso desse exemplo, estamos fazendo uma chamada para um arquivo somente com programação javascript que será executada na página html do navegador.

```

<body>
  <header>
    <h1>Título I</h1>
    <h2>Título II</h2>
  </header>

  <article>
    <header>
      <h3>Título III</h3>
      <p>Linha 1<br /> Linha 2</p>
    </header>

    <footer>
      <p>Aplicações para Internet</p>
    </footer>
  </article>

  <footer>
    <p>Todos os direitos reservados. Universidade Cruzeiro do Sul</p>
  </footer>
</body>
  
```

Figura 11

- » **<body>**: todo elemento que será exibido no browser deve ser inserido nessa tag.
- » **<header>**: O elemento `<header>` representa um contêiner para a introdução de um conteúdo ou para um conjunto de itens de navegação, geralmente são acompanhadas das tags `<h1> ... <h6>`.
- » **<article>**: A tag `<article>` especifica conteúdo independente e autônomo. Um `<article>` deve fazer sentido por conta própria e deve ser possível distribuí-lo de forma independente do resto do site.
- » **<footer>**: A marca `<footer>` define um rodapé para um documento ou seção.



Quando abrimos uma página HTML em um navegador, o navegador irá interpretar as tags, ou seja, irá fazer uma análise sintática, definindo como cada parte do texto será apresentada.

Um problema comum no desenvolvimento de páginas HTML desde o seu surgimento e outras tecnologias como CSS, é a compatibilidade entre os navegadores, ou seja, uma página desenvolvida em HTML5 pode rodar somente no Chrome, por exemplo, ou aparecer de um jeito no Internet Explorer e de outro no Firefox, também temos que tomar cuidado quando falamos em dispositivos ou sistemas diferentes, como visualizar nossa página em um celular. Nem todos os aparelhos estão aptos a rodar os códigos atuais.

Motores de renderização é o mecanismo utilizado pelos navegadores para processar o código das páginas web, os principais motores dos principais navegadores são:

- » **Webkit** (é o mais compatível com HTML 5) – Safari, Chrome
- » **Gecko** – Firefox
- » **Trident** – Internet Explorer

Devemos sempre procurar deixar nossos códigos compatíveis com esses motores para que possamos atingir o maior número possível de usuários.

O WebKit é uma ferramenta utilizada nos navegadores Safari e Chrome. O WebKit é um motor de renderização.

Gecko é um motor de layout desenvolvido pelo Projeto Mozilla. Trata-se de um motor com código aberto utilizado em aplicações construídas pela Mozilla Foundation e pela Mozilla Corporation, como o navegador mundialmente conhecido Firefox e o SeaMonkey.

Vejam um exemplo de utilização de motor de renderização.

```
div{  
    -webkit-border-radius: 10px;  
    -moz-border-radius: 10px;  
    -ms-border-radius: 10px;  
    -o-border-radius: 10px;  
    border-radius: 10px;  
  
    height:50px;  
    width: 150px;  
    text-align:center;  
    background-color: green;  
}
```

Observe que para cada browser devemos utilizar o renderizador para que o efeito final seja o mesmo. Essa codificação deve ser feita utilizando o css, isso não é um tag html.

```

1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Tutorial de HTML5</title>
5          <meta name="description" content="trabalhando com renderização">
6          <meta charset="utf-8">
7          <style>
8              div{
9                  background-color: green;
10                 -webkit-border-radius: 10px;
11                 -moz-border-radius: 10px;
12                 -ms-border-radius: 10px;
13                 -o-border-radius: 10px;
14                 border-radius: 10px;
15                 height:50px;
16                 width: 150px;
17                 text-align:center;
18             }
19         </style>
20     </head>
21
22     <body>
23         <header>
24             <h1>Trabalhando com redenrizaçāo</h1>
25             <div>Texto Informativo</div>
26             <br/>
27             <button id="btnBotal">Clique aqui</button>
28         </header>
29
30         <footer>
31             <p>Todos os direitos reservados. Universidade Cruzeiro do Sul</p>
32         </footer>
33     </body>
34 </html>

```

Figura 12

Resultado no Internet Explorer

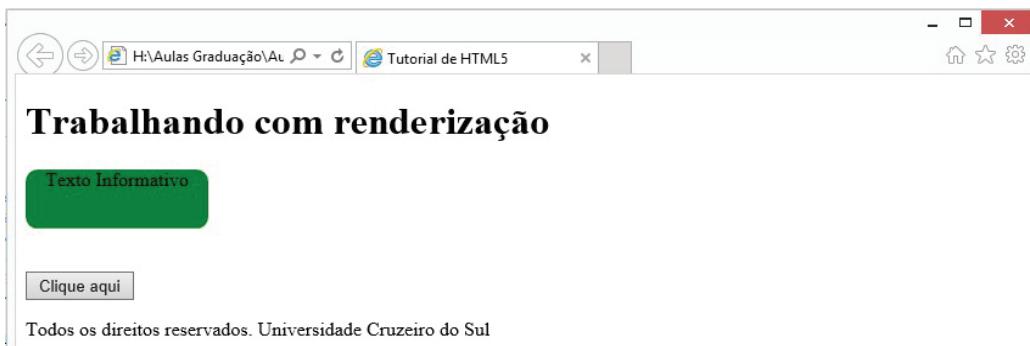


Figura 13

Resultado no Google Chrome

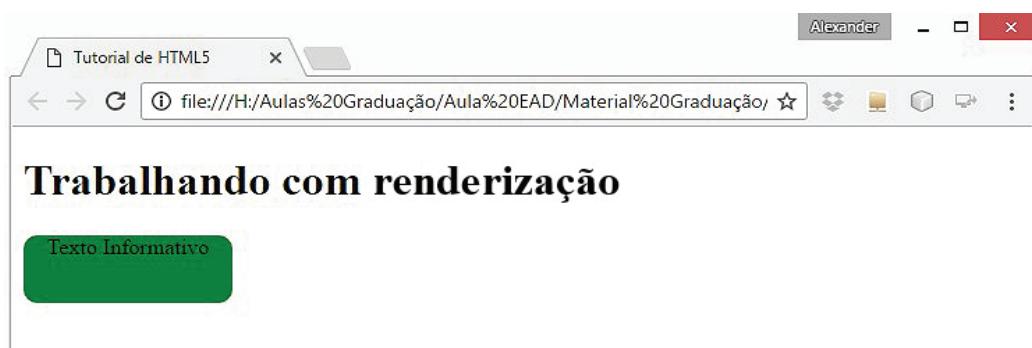


Figura 14

Resultado no Firefox



Figura 15



Todo desenvolvedor web deve se preocupar com essa formatação a fim de garantir que o resultado do site seja igual independente do browser

# Material Complementar

## Indicações para saber mais sobre os assuntos abordados nesta Unidade:

### Sites

#### **HTML5 - Tutorial**

<https://goo.gl/N0ra>

#### **HTML5 - Training**

<https://goo.gl/vHV69g>

### Vídeos

#### **HTML5**

Curso de HTML (aula 1) Iniciantes / básico - Introdução e estrutura do html

<https://youtu.be/5YdekrQOsM>

# Referências

SILVA, M. S. **Construindo sites com CSS e (X)HTML**: sites controlados por folhas de estilo em cascata. São Paulo: Novatec, 2008.

SILVA, M. S. **HTML 5**: a linguagem de marcação que revolucionou a web. São Paulo: Novatec, 2011.



**Cruzeiro do Sul Virtual**  
Educação a Distância

www.cruzeirodosulvirtual.com.br  
Campus Liberdade  
Rua Galvão Bueno, 868  
CEP 01506-000  
São Paulo - SP - Brasil  
Tel: (55 11) 3385-3000



**Cruzeiro do Sul**  
Educacional