

Minicurso introdutório de desenvolvimento para web em HTML5

Sobre mim

- Graduando em Engenharia Eletrônica - UNIFEI
- Trabalhou em desenvolvimento de sistemas empresariais (Base em servidores LAMP)
- Possui experiência em servidores Web, bancos de dados SQL e diversas linguagens server-side (PHP, JSP, Python e afins)

Revisão do HTML

Objetivo inicial da Web

- Criada para compartilhamento de artigos científicos entre pesquisadores de universidades
- Usava um formato simples de linguagem de marcação, somente para criar títulos, parágrafos, listas, tabelas e inserir imagens, pois não era necessário o uso de estilos, menus complexos e afins

Evolução da Web

- A web evoluiu para o desenvolvimento de páginas pessoais, corporativas, com menus, páginas de indexação e afins, necessitando a adição de novos componentes na sua estrutura:

CSS – Estilização

ECMAScript – Dinamização

- A necessidade por criar páginas dinâmicas fez com que fossem desenvolvidas tecnologias server-side, usando linguagens como **PHP, JSP, ASP, Python** para criar as páginas antes de enviá-las ao cliente

O HTML

- **H**ipertext **M**arkup **L**anguage
- Baseado no SGML
- Possibilita a descrição dos elementos da página, como textos, títulos, listas, tabelas, blocos e muitos outros
- Podem ser estilizados por um documento em outra linguagem de marcação (**CSS**)
- A maneira em que é organizado **DOM** é desenvolvida e mantida pela W3C (World Wide Web Consortium)

A estrutura do HTML

- Escrito através de tags no seguinte formato:
`<tag prop="valor">Valor</tag>`
- Elementos simples podem ser descritos por uma tag (Parágrafos, imagens...)
- Elementos compostos podem ser escritos por tags aninhadas (Lista e seus itens, tabela e suas colunas e linhas...)

Exemplo 1

Exemplo da estrutura HTML

CSS

- **Cascading Style Sheet**
- Linguagem de marcação
- Permite a descrição do estilo dos elementos da página (Tamanho da fonte, cor, posicionamento, altura, largura...)
- Baseada em blocos, ex:

```
#id {  
    prop1: valor;  
    prop2: valor;  
}
```

Exemplo 2

Exemplo da estrutura do CSS

Iniciando os estudos do HTML5

HTML5

- Começou a ser desenvolvido por um grupo paralelo chamado WHATWG formado por desenvolvedores dos principais navegadores e outros
- Sua proposta e suas novas características foram aceitas pela W3C sendo especificadas como uma nova versão da linguagem de marcação

HTML5

→ **Web Semântica**

- Novos elementos e recursos que permitem fazer coisas antes só eram possíveis com flash, applets java, e similares
- Novos métodos para manipulação dos elementos da página pelo javascript
- Novas ferramentas que permitem o melhor desenvolvimento de Web Apps, como armazenamento local, canvas (área para desenho) e outras novidades
- **Acessibilidade:** Novas tags para identificar sessões da página para que ela possa ser navegada facilmente por leitores de tela, navegadores minimalistas e algoritmos de busca

HTML5

- Não possui suporte total nos navegadores
- Desenvolvimento e implementação lentos e constantes
- **Necessidade da comunidade desenvolvedora de aumentar sua utilização para que o padrão seja adotado mais rapidamente**

Tags semânticas

section A tag `section` define uma nova seção genérica no documento. Por exemplo, a home de um website pode ser dividida em diversas seções: introdução ou destaque, novidades, informação de contato e chamadas para conteúdo interno.

nav O elemento `nav` representa uma seção da página que contém links para outras partes do website. Nem todos os grupos de links devem ser elementos `nav`, apenas aqueles grupos que contém links importantes. Isso pode ser aplicado naqueles blocos de links que geralmente são colocados no Rodapé e também para compor o menu principal do site.

article O elemento `article` representa uma parte da página que poderá ser distribuído e reutilizável em FEEDs por exemplo. Isto pode ser um post, artigo, um bloco de comentários de usuários ou apenas um bloco de texto comum.

aside O elemento `aside` representa um bloco de conteúdo que referencia o conteúdo que envolve do elemento `aside`. O `aside` pode ser representado por conteúdos em sidebars em textos impressos, publicidade ou até mesmo para criar um grupo de elementos `nav` e outras informações separados do conteúdo principal do website.

Tags semânticas

aside O elemento `aside` representa um bloco de conteúdo que referencia o conteúdo que envolve o elemento `aside`. O `aside` pode ser representado por conteúdos em sidebars em textos impressos, publicidade ou até mesmo para criar um grupo de elementos `nav` e outras informações separados do conteúdo principal do website.

hgroup Este elemento consiste em um grupo de títulos. Ele serve para agrupar elementos de título de H1 até H6 quando eles tem múltiplos níveis como título com subtítulos e etc.

header O elemento `header` representa um grupo de introdução ou elementos de navegação. O elemento `header` pode ser utilizado para agrupar índices de conteúdos, campos de busca ou até mesmo logos.

footer O elemento `footer` representa literalmente o rodapé da página. Seria o último elemento do último elemento antes de fechar a tag HTML. O elemento `footer` não precisa aparecer necessariamente no final de uma seção.

time Este elemento serve para marcar parte do texto que exibe um horário ou uma data precisa no calendário gregoriano.

Exemplo 3

Exemplo das tags de sessão:
header, nav, article, footer

Estrutura básica

```
<!DOCTYPE HTML>
<html lang="pt-br">
<head>
  <meta charset="UTF-8">
  <link rel="stylesheet" type="text/css" href="estilo.css">
  <title></title>
</head>
<body>

</body>
</html>
```

Modelos de conteúdo

Ver arquivo externo: [modelos_conteudo.html](#)

Novos elementos de formulário

Novos tipos na tag `<input>`:

- tel
- search
- email
- url
- datetime, date, month, week, time, datetime-local
- number
- range
- color

Novos tipos de dados e validadores

Novos elementos que podemos adicionar à tag input para adicionar filtros:

- autofocus
- placeholder
- required
- maxlength
- pattern
- novalidate (para form) e formnovalidate (para submit)

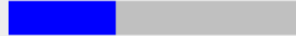
Caixa de detalhes

É possível fazer com que conteúdo considerado “detalhe” de outro seja escondido e mostrado, como na seguinte ilustração:

Veja como um agente de usuário poderia renderizar isso:

▶ Copiando  37,5%

E ao clicar:

▼ Copiando  37,5%

Tamanho total:
39.248KB
Transferido:
14.718
Taxa de transferência:
127KB/s
Nome do arquivo:
HTML5.mp4

Conteúdo editável

- Agora é possível fazer com que outros elementos que não são nativamente de entrada, tornarem editáveis
- Isso facilita a criação de editores dinâmicos, como os editores em DIV criados pelo Google no Google Docs
- Basta adicionar ao elemento, o atributo `contenteditable="true"`

Drag n' Drop

→ Adicionados os eventos de drag:

Para o objeto que está sendo arrastado:

ondragstart, ondrag, ondragend

Para o objeto sobre qual o outro é arrastado:

ondragenter, ondragleave, ondragover, ondrop

→ Possibilita criar facilmente novos elementos compostos de input

Tags audio e video

- Adicionadas as tags para embed de arquivos de áudio e vídeo, removendo a necessidade de usar flash para reproduzir esses conteúdos
- Os elementos de áudio e vídeo podem ser controlados por métodos no javascript, possibilitando montar players elaborados (como o novo player html5 do youtube)

Tag device, stream e P2P

- Permite acesso à periféricos do dispositivo do cliente, streaming de informações e acesso no formato peer-to-peer
- Os navegadores suportariam acesso à webcam, e futuramente, podem adicionar suporte à outros, realizar streaming de informação (acesso direto à um socket de comunicação) e comunicar com outro cliente
- **Não implementado por nenhum navegador no momento, estamos no aguardo :(**

MathML e SVG

- Permitem a adição de formulas matemáticas em MathML pela tag **math** e adição de desenhos vetoriais pela tag **svg**
- MathML não é suportado pelos navegadores modernos com perfeição, porem o svg é muito bem suportado
- Ambos podem ser manipulados por javascript

Canvas

- Permite a adição de desenhos dinâmicos gerados completamente por javascript
- Permite a criação de jogos completos que podem ter toda a sua lógica desenvolvida em javascript
- Longa lista de métodos e propriedades que podem ser desenvolvidas

Server-sent events

- Possível substituinte da API RESTless
- Permite a conexão com um servidor, mante-la ativa e permitir que o javascript dispare eventos quando o servidor enviar um dado
- <http://www.w3c.br/cursos/html5/conteudo/capitulo15.html>

Novos métodos para busca de elementos

- Adição do método `getElementsByClassName("classe")`: Retorna um array com todos os elementos pertencentes à classe especificada
- A propriedade `innerHTML` passa a ser considerada documentada e oficial, apesar de ser suportada pelos todos os navegadores populares à muito tempo
- Adição do método `getSelection()` que retorna a seleção atual do usuário e métodos auxiliares para manipular a seleção

Dataset

- Agora é possível adicionar dados adicionais à um elemento utilizando o prefixo data:

```

```

- Para acessar as propriedades **data** pelo javascript, utilizamos o seguinte atributo:

```
var img=document.getElementById('i1')  
proc=img.dataset.processor
```

Elemento menu

- Usado para criar menus, antes criados usando listas
- Permitiria que menus de dropdown fossem facilmente criados simplesmente aninhando tags
- Ainda não implementados nos navegadores, porém pode ser utilizado como **tag semântica**

Histórico de sessão

- Permite manipular o histórico da sessão do usuário (botão de voltar para a página anterior), para permitir que o botão de retorno reflita em uma ação na própria página
- Extremamente útil para o desenvolvimento de aplicações em que intuitivamente, o botão de retornar teria função de desfazer ou retornar para uma sessão anterior
- Impede que o usuário retorne à página anterior saindo da aplicação
- <http://www.w3c.br/cursos/html5/conteudo/capitulo21.html>

localStorage

- Permite um armazenamento de dados do lado do cliente muito mais eficiente que o uso de cookies
- Dados são armazenados no lado do cliente na forma de dicionário
- Pode ser utilizado para armazenar grande quantidade de dados do lado do usuário para criação de aplicações complexas
- <http://www.w3c.br/cursos/html5/conteudo/capitulo21.html>

Scrolling

- Faz com que a tela role até um elemento específico
- Já existiam implementações de funções similares em bibliotecas javascript (Jquery e afins), porém, agora é implementado nativamente e possui uma performasse maior
- Pode ser utilizado facilmente utilizando o seguinte método:
`document.getElementById('e1').scrollIntoView()`

Geolocalização

- Permite retornar a posição atual (latitude e longitude) do usuário
- Se o dispositivo do usuário tiver um periférico GPS, o mesmo pode ser utilizado para aumentar a precisão, caso o contrário, a geolocalização pode ser baseada por informações da rede e afins
- <http://www.w3c.br/cursos/html5/conteudo/capitulo24.html>

Dúvidas?!

Referências

W3C. Curso HTML5. Disponível em:
<<http://www.w3c.br/cursos/html5/conteudo/index.html>>. Acesso em: 25 maio 2015.

Mark Pilgrim. Como chegamos aqui. Disponível em:
<<http://diveintohtml5.com.br/past.html>>. Acesso em 25 maio 2015.