

INSTITUTO INFNET
ESCOLA SUPERIOR DA TECNOLOGIA DA INFORMAÇÃO
CURSO DE ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

Bruno Wendel Fernandes Camello

Assessment - Interfaces com HTML e CSS [21E4_1]

**Brasília
2021**

Perguntas a serem respondidas em ambas as propostas:

1) Quando é recomendando a criação de um site separado para mobile? Seria este o caso do site integrado de Interfaces - SPA?

Com base no que foi apresentado nas aulas e meu estudo sobre interfaces, sites mobile, por serem desenvolvidos especialmente para dispositivos móveis, permitindo uma melhor experiência e maiores taxas de conversão, tendem a ser mais rápidos e responsivos, por terem menos código e informações a serem trabalhadas.

No caso de uma SPA, temos a questão de ser adaptável para tipos diferentes de usuários, tendo um maior escopo e menor custo, por ser um único código para o site inteiro, economizando em certos fatores, porém perdendo em outros. No caso do site do AT, acredito que uma SPA seria mais interessante por ser uma loja, de escopo inicial, onde a proposta seria alcançar mais clientes e inicialmente ser menos custosa.

2) No caso de um CSS específico para print, como ele deve ser inserido?

Inserido utilizando o @media print {...} via tag html <link rel="stylesheet" type="text/css" href="/print.css" media="print" />

Ele deve ser inserido para dar uma melhor portabilidade, recurso offline, para disponibilidade em leitores, como eReaders, vão tirar proveito desse recurso.

3) O elemento audio permite como elemento-filho o elemento source. Quais são os atributos que o source permite e qual conteúdo eles suportam? Quais os outros atributos do elemento audio e quando devem ser utilizados?

Como todos os outros elementos de HTML, esse elemento suporta os global attributes.
src:

Requerido, endereço do arquivo de mídia.

type:

O tipo MIME do arquivo, opcionalmente com um parametro de codecs. Veja o RFC 4281 para informações sobre como especificar codec.

media:

Definição do tipo de mídia (Media query) pretendido.

Fonte: <https://developer.mozilla.org/pt-BR/docs/Web/HTML/Element/source>

Segundo o próprio Mozilla.org, O conteúdo permitido no elemento audio é:

Transparent content, tanto um atributo src quanto um ou mais elementos <source>, seguidos por um flow content ou phrasing content, sem <video> ou elementos <audio>.

E temos vários atributos que são permitidos, como: autoplay, buffered, controls, loop, mozCurrentSampleOffset, muted, played, preload e src. Cada um tem sua utilização para definir a apresentação de vídeo, o preload tem os seguintes valores: none, metadata, auto.

none: indica que o usuário não necessitará consultar o áudio ou que o servidor quer minimizar seu tráfego; em outros termos indica que o áudio não deve ser pré-carregado; metadata: indica que embora o usuário não necessitará consultar o áudio, pegar os meta-dados (ex: comprimento) é interessante;

auto: indica que o usuário necessita ter prioridade; em outros termos isso significa que, se necessário, o áudio inteiro pode ser baixado, mesmo que não seja esperado a execução;

muted recebe um boolean que define se o áudio será iniciado mutado ou não.

played tem um timeRaged que indica se o áudio foi reproduzido ou não.

controls deixa ou não presentes os controles de reprodução do áudio.

Dentre os outros atributos falados anteriormente, todos os detalhes de implementação estão definidos da documentação do mozilla.org(
<https://developer.mozilla.org/pt-BR/docs/Web/HTML/Element/audio>).

4) O elemento video permite como elemento-filho o elemento source. Quais são os atributos que o source permite e qual conteúdo eles suportam? Quais os outros atributos do elemento video e quando devem ser utilizados?

Como mencionado na questão anterior, o elemento source tem as definições descritas. Como descrito na questão anterior, temos os elementos de audio presentes na tag video e mais alguns, por ser uma mídia que contém áudio e imagem sendo apresentados, temos alguns elementos a mais como width e height, poster que apresenta uma imagem prévia sobre o vídeo, devemos adicionar uma URL indicando o caminho da imagem.

5) No caso do canvas, HTML 5 nunca, em momento algum, fez qualquer animação. Isso porque o HTML 5 é uma simples linguagem de marcação. Dessa forma, o que é necessário para criar animações com HTML5? O que são APIs do HTML5?

Para criar animações com HTML5 nos utilizamos do JS para dar vida ao site, manipulando a DOM. APIs do HTML5 são implementações da linguagem de marcação que ajudam a controlar a página melhor, são novas funcionalidades encapsuladas em APIs para controle do HTML5 e acesso via JS, temos muitas APIs, como:

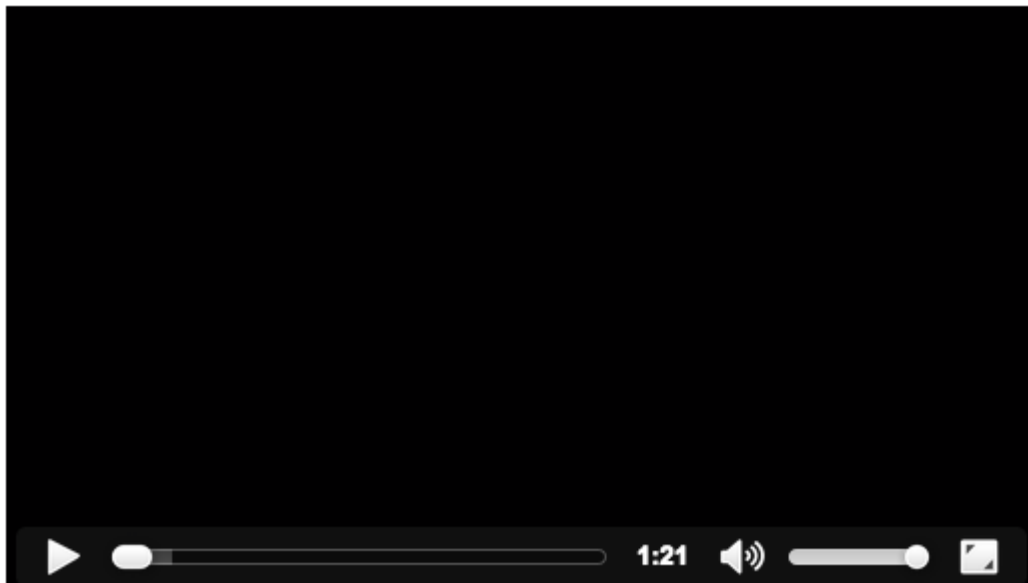
- Canvas
- Validação de formulários
- Controles de áudio e vídeo
- Application cache / offline applications
- Funcionalidade de markup editável (contenteditable)
- Drag and drop
- Novas funcionalidades para manipulação do histórico do navegador
- Web Storage (localStorage e sessionStorage)
- Web messaging

- Microdata
- Web Workers

- Web Sockets
- Geolocation
- File API
- Indexed DB
- File Writer
- Notifications

6) Como o CSS3 pode ser aplicado na aparência da multimídia HTML5.
Demonstre com um exemplo.

Podemos modificar como são mostrados os controles, tamanho da apresentação da multimídia para o usuário, podemos fazer inúmeras alterações na mídia que queremos apresentar, relacionadas a estilo.



Podemos alterar o rodapé do vídeo, podemos alterar como é a barra e os ícones dos controles, dentre várias alterações possíveis.