



HTML5 - POR QUE USÁ-LO?

Ícaro Carlos Andrade Costa (icarocarlosandrade@gmail.com)
Igor Antônio Santos Andrade (igorantonioandrade@gmail.com)

RESUMO

A linguagem HTML surgiu a mais de duas décadas como um padrão para a comunicação e troca de documentos pela internet, sendo considerado um dos pilares da própria web. Graças à sua última versão, o HTML5, houve um grande salto no desenvolvimento de sites e aplicações web, permitindo uma melhor experiência para o usuário e facilitando a manipulação dos elementos pelo desenvolvedor, permitindo a criação de aplicações mais ricas. Diversas novas tags foram criadas e várias modificadas, além da adição de APIs que permitem funcionalidades como a geolocalização, armazenamento de dados no cliente e o funcionamento da aplicação em modo offline. Com a utilização do HTML5 somado ao CSS3 e JavaScript, é possível desenvolver sites responsivos que atendam a múltiplos dispositivos, com grande interatividade e interfaces ricas. Além disso, através de frameworks como o PhoneGap proporciona-se o desenvolvimento de aplicações mobile híbridas multiplataforma. A proposta deste artigo é listar algumas das funcionalidades mais interessantes, destacando a importância e os principais motivos para trabalhar com HTML5. Foi desenvolvido através de pesquisas bibliográficas nas principais referências sobre o tema, identificando as características mais relevantes do padrão. Sendo um padrão definido pela W3C e que foi construído sobre anos de sucesso do seu antecessor, usar o HTML5 é garantia de estar usando aquilo que já é o presente e será o futuro da Web.

Palavras-chave: *HTML5, Multiplataforma, Mobile*

1 - INTRODUÇÃO

A HTML5 foi desenvolvida após anos de uso bem sucedido de seu antecessor, a versão 4. A nova versão surge então a partir de uma linguagem de marcação já consolidada e estabelecida, e procura trazer novas funcionalidade e aprimorar o formato, permitindo



construir aplicações web mais robustas, mais interativas e mais fáceis de serem lidas por humanos (SILVA, 2011).

Sendo assim, todo o conjunto de novas funcionalidades desenvolvidos na linguagem HTML5 foram planejadas após mais de uma década de uso da HTML 4. Toda essa experiência de alguns dos profissionais e de algumas das empresas mais importantes do mundo da web foi usada para criar um novo padrão que preenche as necessidades que foram encontradas durante esse tempo de trabalho (SILVA, 2011).

Parte essencial da decisão de desenvolvimento de uma nova versão para o padrão HTML foi a necessidade de compatibilidade com diferentes versões de navegadores em diferentes plataformas. A HTML5 tem como pré-requisito então a compatibilidade retroativa, permitindo que os clientes que suportam a nova versão também consigam entender completamente páginas desenvolvidas com a HTML 4 (W3C ESCRITÓRIO BRASIL, 2010).

Esse trabalho foi desenvolvido através de pesquisas bibliográficas nas principais referências sobre o tema, buscando identificar as características mais relevantes da nova versão do padrão HTML. Considerando que se trata de um padrão web, naturalmente a maior parte do material disponível e mais atualizado se encontra na própria web, inclusive na documentação da definição do padrão pela W3C.

2 - O QUE É A HTML

O *Hypertext Markup Language* - referenciado normalmente como HTML - é uma linguagem de marcação para criação de páginas Web. É uma linguagem definida como um padrão para publicação de conteúdo (texto, imagem, vídeo, áudio e etc) na internet, sendo um dos pilares da Web (W3C ESCRITÓRIO BRASIL, 2010).

Foi desenvolvido originalmente por Tim Berners-Lee quando este trabalhava na Seção de Computação da Organização Europeia de Pesquisa Nuclear, o CERN. Seu objetivo inicial era descobrir um método que permitisse aos cientistas de todo o mundo compartilhar eletronicamente suas descobertas e pesquisas, e acreditava que seria possível realizar a interligação entre esses diferentes documentos localizados em computadores diferentes através de links (SILVA, 2011; W3C, 2000).

Berners-Lee escreveu a HTML e desenvolveu o primeiro cliente (browser) e o software do servidor em 1989 e no mesmo ano implementou a primeira comunicação bem-sucedida usando o que seria a primeira versão do HTTP, protocolo utilizado para a comunicação de hipertextos na web. A HTML ganhou popularidade quando o Mosaic,



browser desenvolvido por Marc Andreessen na década de 1990, ganhou força. A partir daí, desenvolvedores e fabricantes de browsers utilizaram a HTML como base, mas nem sempre compartilhando as mesmas convenções (SILVA, 2011; W3C, 2015).

Com a fabricação de novos browsers por diversas empresas a HTML passou a possuir novas formas de marcação criada por cada um deles, compatíveis apenas com seus navegadores, gerando um caos absoluto. Com o objetivo de facilitar o desenvolvimento da linguagem foi criado em 1994 o World Wide Web Consortium (W3C), um consórcio internacional formado por empresas, instituições, pesquisadores, desenvolvedores e o público em geral interessado na evolução da internet. A principal finalidade da organização é o de desenvolver a web em seu potencial máximo, criando normas e especificações aplicáveis aos diversos segmentos e setores da web, desde tecnologias e softwares até fabricantes e fornecedores (SILVA, 2011; W3C, 2000).

Durante os anos algumas novas versões do HTML foram lançadas com o objetivo de dar mais poder para a linguagem e também de torná-la direcionada para o seu propósito exclusivo de marcação e estruturação de textos. Em 1997 foi lançada a versão 4 da HTML, sendo usada com sucesso por diversos anos e, no ano 2000, foi lançada a primeira versão da XHTML, uma versão da linguagem de marcação baseada em XML. Acreditava-se inicialmente que esse seria o futuro da linguagem (W3C ESCRITÓRIO BRASIL, 2010; XHTML, 2015).

Enquanto a W3C estava focada em desenvolver uma segunda versão para o padrão XHTML, foi criado o WHATWG - *Web Hypertext Application Technology Working Group* - um grupo fundado por desenvolvedores de várias empresas interessadas em ver uma nova versão da HTML e insatisfeitas com o rumo que seguia o XHTML. Por volta de 2006 o trabalho do grupo passou a ser reconhecido pela W3C e em 2009 o trabalho para desenvolvimento do XHTML 2 foi descontinuado. Em outubro de 2014 a W3C finalizou e lançou a sua versão completa e final da quinta versão do padrão HTML (W3C ESCRITÓRIO BRASIL, 2010; XHTML, 2015; WHATWG, 2015).

O conceito base da HTML é baseado na ideia de *hipertextos*, que são textos aos quais se agregam blocos de informação como outros textos, imagens, sons ou vídeos, e que são acessados através de referências. Na HTML essas referências são os *hyperlinks*, ou simplesmente *links*. Através das conexões criadas entre os links é possível acessar uma vasta quantidade de informação espalhada e conectada, como se fosse uma rede de informações



Estes elementos conectados formam uma grande rede de informação (W3C ESCRITÓRIO BRASIL, 2010).

O desenvolvimento de páginas em HTML se dá através da utilização das *tags*, elementos de marcação utilizados para inserir objetos e formatar a exibição de conteúdo na página. As *tags* são definidas como termos ou abreviações em inglês colocados entre colchetes angulares, como por exemplo `<html>`, que podem ou não possuir um correspondente para fechamento, `</html>`. Os clientes, conhecidos como browsers ou navegadores podem ler a HTML, interpretar e renderizar o conteúdo (W3C ESCRITÓRIO BRASIL, 2010).

3 - NOVIDADES DA HTML5 E SEUS BENEFÍCIOS

Não faz parte do objetivo desse artigo ser uma referência ou um catálogo das mudanças apresentadas na HTML5. Esses catálogos existem e são de fácil acesso para aqueles que estejam interessados em buscar mais informações técnicas e detalhadas de tudo aquilo que mudou, do que foi removido e adicionado com a nova versão do padrão. A proposta então é listar algumas das funcionalidades mais interessantes e que mudam a forma de desenvolver usando a HTML, sem se preocupar em detalhar ou exemplificar o uso dos elementos.

Um dos recursos mais importantes adicionados foi o suporte multimídia nativo na HTML. É possível agora adicionar vídeo e áudio para exibição no navegador simplesmente através do uso das *tags* próprias para isso, da mesma forma que sempre foi possível fazer com imagens. Com isso eliminamos a necessidade de utilizar plugins como o *Flash* ou outros, facilitando em muito a utilização desses recursos.

Para adicionar áudio através da HTML basta o uso de uma única *tag*. Como não há a utilização de plugins e sim a própria linguagem, diversas customizações podem ser feitas com o uso de CSS, como mudar a cor, o tamanho, ou até mesmo criar um player totalmente do zero, além de permitir controlar propriedades como *play*, *pause*, *volume*, através de JavaScript.

Não muito diferente é o uso de vídeos no HTML, bastando também apenas uma *tag*. Da mesma forma este permite um total controle através de CSS e JavaScript, sendo possível criar seu próprio player de uma maneira muito poderosa e flexível. Entretanto, um problema comum que ainda existe é quanto aos formatos disponíveis para uso e compatibilidade entre os diferentes browsers. Considerando que diferentes clientes podem suportar alguns formatos



de vídeo e áudio e não suportar outros, é possível apresentar o mesmo conteúdo em diversos formatos para serem renderizados da melhor forma possível de acordo com o navegador.

Dois novos elementos presentes nessa versão da linguagem são o canvas e o SVG. Ambos são elementos gráficos voltados para criação de imagens na HTML, cada um sendo mais específico para um determinado tipo de uso. O canvas é um elemento mais destinado à criação de imagens em 2D e 3D, quando se pretende lidar com imagens programaticamente e permite delimitar uma área para criação dinâmica de imagens, como gráficos estáticos, dinâmicos ou até mesmo jogos, permitindo o uso de uma linguagem de programação. Já o elemento SVG é um padrão utilizado para a criação de imagens e gráficos vetoriais, utilizado para interfaces animadas e exibição de dados, permitindo diversas resoluções. Saber quando utilizar cada um deles é uma tarefa importante (SILVA, 2011).

Diversos novos recursos foram adicionados à linguagem, como a existência de tags que permitem arrastar e soltar objetos pela página - *drag and drop*. Entretanto, alguns recursos realmente avançados permitem funcionalidades extremamente úteis e levaram a linguagem a um novo patamar. Entre esses podemos citar como os principais a API de armazenamento de dados, a API para funcionamento offline da aplicação e a API de geolocalização, sendo essa última uma funcionalidade sempre relacionada com a HTML5 mas que na verdade foi lançada alguns anos antes (SILVA, 2011).

O armazenamento de dados permite que dados sejam armazenados no navegador do usuário sem a utilização de cookies. O uso de cookies trazia alguns problemas como a complexidade ao lidar com eles e a baixa capacidade de armazenamento. O HTML5 trouxe uma nova maneira de fazer isso através da API Storage e seus dois objetos principais, o *localStorage* e o *sessionStorage*. Ambos funcionam no escopo global e permitem armazenamento no lado cliente, mas possuem suas diferenças.

O objeto *localStorage* permite o armazenamento dos dados no cliente sem expiração definida e que persistem em diferentes janelas ou abas do navegador. Com isso, se o usuário fechar o navegador e abrir novamente, mesmo que dias ou semanas depois, os dados estarão lá. Já o objeto *sessionStorage* armazena os dados durante a sessão atual do navegador e separada para cada documento HTML existente. Dessa forma é possível coexistirem múltiplas janelas ou abas do navegador ao mesmo tempo, mas cada uma com seus conjunto de dados independentes (SILVA, 2011).

Uma outra grande limitação dos cookies que foi resolvida pela API Storage é o fato destes só permitirem o armazenamento de strings. Na nova versão é possível o



armazenamento de qualquer valor javascript. Além disso, o espaço de armazenamento permitido é de 5MB para cada domínio, um tamanho bastante considerável, principalmente quando se compara com os 20KB dos cookies (SILVA, 2011; PILGRIM, 2015).

Através da HTML5 podemos informar ao navegador que determinados dados devem ser armazenados em cache para que a aplicação possa funcionar offline. Isso pode ser feito através do uso de um arquivo de manifesto, em que você indica quais objetos devem ser armazenados. Dessa forma o próprio navegador, ao ler o arquivo, armazena os dados internamente. Podemos fazer uso também da API ApplicationCache que define uma série de métodos, propriedades e eventos destinados a monitorar e controlar o status e os dados armazenados em uma aplicação offline, através do objeto *applicationCache* (SILVA, 2011; PILGRIM, 2015).

Como dito anteriormente o recurso de geolocalização não foi uma funcionalidade implementada para o HTML5. A API Geolocation já estava sendo desenvolvida a muito tempo e foi lançada em 2008, mas ficou bastante difundida na época do lançamento da nova versão do HTML e por isso a confusão. Devido à isso é sempre tratada juntamente com as novas funcionalidades do HTML5, o que na verdade é uma falsa informação (SILVA, 2011).

A API de geolocalização permite o acesso através de scripts à informações relativas à localização do dispositivo que hospeda a funcionalidade. Diversas podem ser as formas de captação dos parâmetros de localização geográficas, sejam elas feitas através de GPS, sinais de redes e endereço IP, redes sem fio, sinais de rádio, Bluetooth, sinais de rede GSM/CDMA ou ainda através do fornecimento dos dados diretamente pelo usuário (SILVA, 2011).

Muitas mudanças buscaram melhorar a legibilidade e simplicidade do código produzido. As indicações *doctype* e *charset* foram alteradas para uma forma bem mais simples e curta, sendo aconselhado a sempre dar preferência a usar o padrão Unicode para melhorar a acessibilidade em qualquer parte do mundo. Além disso, o HTML5 passa a adotar claramente a definição de divisão de responsabilidades e desenvolvimento em camadas, onde diversos elementos foram descontinuados por serem usados apenas para efeitos visuais e de estilo, papel esse que deve ser exercido pelo uso de CSS.

Alguns elementos antigos foram descontinuados simplesmente por ferirem princípios como acessibilidade e usabilidade, tais como o uso de frames. O atributo *lang* passa a ser utilizado em qualquer elemento, permitindo definir claramente qual língua deve ser lida em cada parte da página, facilitando a acessibilidade e mesmo o uso de ferramentas de leitura automática de conteúdo.



Com o objetivo de tornar as páginas desenvolvidas mais semânticas, vários novos componentes de seção foram criados para uma melhor divisão do conteúdo a partir do seu significado para o documento web. Agora as páginas são divididas em seções de acordo com o propósito, seja cabeçalho, navegação ou rodapé, tornando o conteúdo produzido mais legível para humanos e para computadores, permitindo que softwares possam saber o objetivo de cada parte do documento e como lidar com elas.

Muitos dos recursos adicionados permitem uma maior interação com o usuário e o desenvolvimento de ferramentas mais intuitivas. O HTML5 possui um conjunto de APIs para ajudá-lo a obter uma aplicação ou site dinâmico com facilidade. Isso garante um maior poder para a construção de ferramentas com uma rica experiência para o usuário.

4 - HTML5, CSS3 E JAVASCRIPT E O DESENVOLVIMENTO MOBILE

Uma das maiores novidades com relação ao desenvolvimento com HTML5 é o seu uso somado ao CSS3 e JavaScript para a construção de aplicação ricas para o usuário final. O HTML5 e todo o seu novo conjunto de APIs é somado aos recursos dessas outras duas linguagens - CSS, usada para descrever a aparência e formatação das páginas, e JavaScript, para garantir interatividade através da programação no lado cliente (browser) - para garantir uma evolução no desenvolvimento de aplicações e sites web.

Através desse conjunto de ferramentas é possível desenvolver sites responsivos que possam atender a múltiplos dispositivos, transformando os produtos desenvolvidos em multiplataforma. Sendo assim, uma aplicação ou site desenvolvido com HTML5, CSS3 e JavaScript que siga os padrões de qualidade definidos pela W3C e outros padrões comerciais, passa a funcionar em navegadores diferentes de sistemas operacionais diferentes e de dispositivos diferentes, seja desktop ou mobile.

Como mais uma novidade do que é possível fazer com esse conjunto de ferramentas surge o desenvolvimento de aplicativos para dispositivos móveis. Ou seja, não somente podem ser desenvolvidos sites responsivos que podem funcionar perfeitamente nos dispositivos móveis, mas também são construídos aplicativos que funcionam no aparelho e usam as APIs nativas para acesso a câmera, conexão com internet, geolocalização, contatos, armazenamento e outros recursos dos dispositivos.

Essa possibilidade de desenvolver aplicativos para dispositivos móveis utilizando HTML, CSS e JS leva a uma verdadeira revolução no desenvolvimento mobile. Enquanto para desenvolver um novo produto que atenda a vários usuários seria preciso de profissionais



experientes em cada uma das plataformas existentes no mercado, com o uso dessas ferramentas é possível desenvolver o aplicativo apenas uma vez e converter para as diferentes plataformas. Isso pode ser feito através de diversas ferramentas presentes hoje no mercado, alguns livres e outras proprietárias, que permitem ao desenvolvedor empacotar a aplicação num formato que seja compatível com os dispositivos desejados (CLARK, 2010).

Várias ferramentas surgiram e são criadas a cada dia para auxiliar na produção de aplicações mobile com HTML, CSS e JS, sendo que uma das ferramentas de maior destaque é o *PhoneGap*. O *PhoneGap* é um framework de desenvolvimento mobile que permite construir aplicações mobile com HTML, CSS e JS, ao invés de depender das APIs específicas de cada plataforma. Ele permite empacotar e estender as características da ferramenta desenvolvida para trabalhar com o dispositivo. O resultado são aplicativos híbridos, pois em parte usam recursos das APIs das plataformas e são empacotados como o tal para distribuição, e em parte usam recursos de aplicações web (PHONEGAP, 2015).

	iPhone / iPhone 3G	iPhone 3GS and newer	Android	Blackberry OS 6.0+	Blackberry 10	Windows Phone 8	Ubuntu	Firefox OS
Accelerometer	✓	✓	✓	✓	✓	✓	✓	✓
Camera	✓	✓	✓	✓	✓	✓	✓	✓
Compass	X	✓	✓	X	✓	✓	✓	✓
Contacts	✓	✓	✓	✓	✓	✓	✓	✓
File	✓	✓	✓	✓	✓	✓	✓	X
Geolocation	✓	✓	✓	✓	✓	✓	✓	✓
Media	✓	✓	✓	X	✓	✓	✓	X
Network	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Alert)	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Sound)	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Vibration)	✓	✓	✓	✓	✓	✓	✓	✓
Storage	✓	✓	✓	✓	✓	✓	✓	✓

✓ - supported feature
X - unsupported feature due to hardware or software restrictions

Figura 1 - O gráfico mostra quais APIs estão disponíveis para cada plataforma

Fonte: PHONEGAP, 2015

5 - SUPORTE PELOS NAVEGADORES

O *html5test.com* é um site que se propõe a fazer testes dos navegadores e determinar, através de métricas estabelecidas na dificuldade de desenvolvimento e importância das funcionalidades, uma pontuação para o nível de suporte que cada um dos navegadores mais

comuns oferece às novas funcionalidades da HTML5. No momento que foi feito esse artigo o número máximo que pode ser alcançado é 555 pontos. Segue abaixo um gráfico oferecido pelo próprio site mostrando a pontuação dos principais navegadores do mercado:



Figura 2 - Pontuação dos navegadores através dos anos

Fonte: HTML TEST, 2015

Considerando que o gráfico retrata até Janeiro de 2015, as informações de alguns navegadores mudaram para a data de confecção do artigo. Em testes feitos no momento em que o presente trabalho estava sendo produzido foi constatado que o navegador Chrome recebeu a pontuação 523, o Firefox 429 e o Internet Explorer recebeu 341.

6 - CONSIDERAÇÕES FINAIS

A nova versão da HTML trouxe consigo diversas mudanças na estrutura das páginas web desenvolvidas, adicionando novos recursos que trazem muitas vantagens para os profissionais e para os usuários. Mais que isso, a nova versão vem carregada de uma filosofia que envolve desde o desenvolvimento mais simples e intuitivo de páginas até uma maior acessibilidade para alcançar a todos, não importa sua localização geográfica ou mesmo o tipo de dispositivo que esteja usando. Com as novas técnicas de desenvolvimento responsivo e as novas ferramentas que permitem desenvolver aplicação mobile através HTML, CSS e JS, o HTML5 passa a ser multiplataforma.



Diversos browsers já adotaram a maior parte das definições do padrão, além do que várias ferramentas que antes tinham grande espaço foram trocadas pela simplicidade da forma que as coisas são feitas no HTML5. Considerando ainda que a HTML5 é um padrão adotado e definido pela W3C e que foi construído sobre anos de sucesso do seu antecessor, usar o padrão é garantia de estar usando aquilo que já é o presente e será o futuro da Web.

REFERÊNCIAS

CLARK, Scott. **Web-based Mobile Apps of the Future Using HTML 5, CSS and JavaScript**, 2010. Disponível em: <<http://www.htmlgoodies.com/beyond/article.php/3893911/Web-based-Mobile-Apps-of-the-Future-Using-HTML-5-CSS-and-JavaScript.htm>>. Acesso em: 02 de mai. 2015.

HTML5 TEST. Disponível em: <<https://html5test.com/>>. Acesso em: 02 de mai. 2015.

PHONEGAP. Disponível em: <<http://phonegap.com/>>. Acesso em: 02 de mai. 2015.

PILGRIM, Mark. **Dive Into HTML5**. Disponível em: <<http://diveintohtml5.info/>>. Acesso em: 02 de mai. 2015.

SILVA, Maurício S. **HTML5**. 1. ed, São Paulo: Novatec Editora, 2011.

XHTML. **X/HTML5 Versus XHTML 2**. Disponível em: <<http://xhtml.com/en/future/xhtml-5-versus-xhtml-2/>>. Acesso em: 02 de mai. 2015.

W3C. **A Little History of the World Wide Web**, 2000. Disponível em: <<http://www.w3.org/History.html>>. Acesso em: 02 de mai. 2015.

W3C ESCRITÓRIO BRASIL. **HTML5**, 2010. Disponível em: <<http://www.w3c.br/pub/Cursos/CursoHTML5/html5-web.pdf>>. Acesso em: 02 de mai. 2015.

W3C. **History of the Web**. Disponível em: <http://www.w3.org/wiki/The_history_of_the_Web>. Acesso em: 02 de mai. 2015.

WHATWG. **HTML Standard**. Disponível em: <<https://whatwg.org/>>. Acesso em: 02 de mai. 2015.

BIBLIOGRAFIA CONSULTADA

CAN I USE. Disponível em: <<http://caniuse.com/>>. Acesso em: 02 de mai. 2015.

DOUGLAS, Allan. **HTML5 - As tags Audio e Video**. Disponível em: <<http://www.devmedia.com.br/html5-as-tags-audio-e-video/26018>>. Acesso em: 02 de mai. 2015



MOZILLA DEVELOPER NETWORK. **General app development tutorial**. Disponível em: <<https://developer.mozilla.org/en/Apps/Tutorials/General>>. Acesso em: 02 de mai. 2015.

W3C. **A vocabulary and associated APIs for HTML and XHTML - W3C Recommendation 28 October 2014**, 2014. Disponível em: <<http://www.w3.org/TR/html5/>>. Acesso em: 02 de mai. 2015.

W3C. **W3C Editor's Draft 23 March 2015**, 2015. Disponível em: <<http://www.w3.org/html/wg/drafts/html/master/>>. Acesso em: 02 de mai. 2015.

WORLD WIDE WEB FOUNDATION. **History of the Web**. Disponível em: <<http://webfoundation.org/about/vision/history-of-the-web/>>. Acesso em: 02 de mai. 2015.