WEB APIs que você provavelmente não sabia que existiam.
HTML 5
A vocabulary and associated APIs for HTML and XHTML

W3C Working Draft 22 January 2008

This Version:
http://www.w3.org/TR/2008/WD-html5-20080122/
Latest Published Version:
http://www.w3.org/TR/html5/
Latest Editor's Draft:
http://www.w3.org/html/wg/html5/
Editors:
Ian Hickson, Google, Inc.
  David Hyatt, Apple, Inc.

Copyright © 2008 W3C® (MIT, ERCIM, Keio). All Rights Reserved. W3C liability, trademark and document use rules apply.

Abstract

This specification defines the 5th major revision of the core language of the World Wide Web, HTML. In this version, new features are introduced to help Web application authors, new elements are introduced based on research into prevailing authoring practices, and special attention has been given to defining clear conformance criteria for user agents in an effort to improve interoperability.

Status of this document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the W3C technical reports index at http://www.w3.org/TR/.

If you wish to make comments regarding this document, please send them to public-html-comments@w3.org (subscribe, archives). All feedback is welcome.

Implementors should be aware that this specification is not stable. Implementors who are not taking part in the discussions are likely to find the specification changing out from under them.
2009

If you’re a web developer looking forward to the new tools in HTML 5, the next generation of the language that powers the web, we have some bad news for you — you’re going to waiting a while.

Ian Hickson, the editor of the HTML 5 specification, recently outlined the time table for HTML 5 and, even assuming browser manufacturers embrace HTML 5 when it reaches the final draft stage, that puts HTML 5’s widespread adoption at 2013. Worse, the final proposal...
2010
Thoughts on Flash

Apple has a long relationship with Adobe. In fact, we met Adobe’s founders when they were in their proverbial garage. Apple was their first big customer, adopting their Postscript language for our new Laserwriter printer. Apple invested in Adobe and owned around 20% of the company for many years. The two companies worked closely together to pioneer desktop publishing and there were many good times. Since that golden era, the companies have grown apart. Apple went through its near death experience, and Adobe was drawn to the corporate market with their Acrobat products. Today the two companies still work together to serve their joint creative customers – Mac users buy around half of Adobe’s Creative Suite products — but beyond that there are few joint interests.

I wanted to jot down some of our thoughts on Adobe’s Flash products so that customers and critics may better understand why we do not allow Flash on iPhones, iPads and iPods. Adobe has characterized our decision as being primarily business driven — they say we want to protect our App Store — but in reality it is based on technology issues. Adobe claims that we a closed system, and that Flash is open, but in fact the opposite is true. Let me explain.

First, there’s “Open”.

Adobe’s Flash products are 100% proprietary. They are only available from Adobe, and Adobe has sole authority as to their future enhancement, pricing, etc. While Adobe’s Flash products are widely available, this does not mean they are open, since they are controlled entirely by Adobe and available only from Adobe. By almost any definition, Flash is a closed system.

Apple has many proprietary products too. Though the operating system for the iPhone, iPod and iPad is proprietary, we strongly believe that all standards pertaining to the web should be open. Rather than use Flash, Apple has adopted HTML5, CSS and JavaScript — all open standards. Apple’s mobile devices all ship with high performance, low power implementations of these open standards. HTML5, the new web standard that has been adopted by Apple, Google and many others, lets web developers create advanced graphics, typography, animations and transitions without relying on third party browser plug-ins (like Flash). HTML5 is completely open and controlled by a standards committee, of which Apple is a member.
Mark Zuckerberg: Our Biggest Mistake Was Betting Too Much On HTML5

Posted Sep 11, 2012 by Drew Olanoff (@drew)

Today, Mark Zuckerberg revealed that Facebook's mobile strategy relied too much on HTML5, rather than native applications.

Not only was this a big mistake with mobile, but Zuckerberg says that its biggest mistake period was the focus on HTML5. This is the first time that the Facebook CEO has openly admitted this, but things are looking good for the new iOS native app.

According to Zuckerberg, people are consuming twice as many feed stories since the update to the new iOS app, which is great.

*The first half year has been a little bit slow on product, but for the next six months I expect a lot of really cool stuff.*

This "really cool stuff" will probably have monetization in mind, as it's very clear that mobile is the path to ad revenue for the company.

It's extremely difficult for a company to nose-dive into an adoption of a particular set of tools and then quickly change course. I suspect that this is exactly what happened with Facebook and things are at least looking up.
2013
It's been a fantastic couple of years for HTML5 games, for both consumers and developers. In fact, HTML5 is turning into a great game development platform, rapidly catching up to the ubiquity of Flash-based browser gaming.

HTML5 comes with some huge advantages that allow users to build apps, accelerated graphics in games, stream HD video and so much more by just using native web code.

So, here's a collection of the 30 most creative and addictive HTML5 games, all of which can be played in your web browser. But, beware, they are extremely addictive!

1) SAND TRAP

Fill the bucket with sand. Sounds simple enough but it's actually trickier than you think as there's a rotating cube maze that hovers above the bucket.
Open Web Platform Milestone Achieved with HTML5 Recommendation

Next Generation Web Technologies Build on Stable Foundation

28 October 2014 — The World Wide Web Consortium (W3C) published a Recommendation of HTML5, the fifth major revision of the format used to build Web pages and applications, and the cornerstone of the Open Web Platform. For application developers and industry, HTML5 represents a set of features that people will be able to rely on for years to come. HTML5 is now supported on a wide variety of devices, lowering the cost of creating rich applications to reach users everywhere.

"Today we think nothing of watching video and audio natively in the browser, and nothing of running a browser on a phone," said Tim Berners-Lee, W3C Director. "We expect to be able to share photos, shop, read the news, and look up information anywhere, on any device. Though they remain invisible to most users, HTML5 and the Open Web Platform are driving these growing user expectations."

HTML5 brings to the Web video and audio tracks without needing plugins; programmatic access to a resolution-dependent bitmap canvas, which is useful for rendering graphs, game graphics, or other visual images on the fly; native support for scalable vector graphics (SVG) and math (MathML); annotations important for East Asian typography (Ruby); features to enable accessibility of rich applications; and much more.

HTML5 is Widely Deployed

HTML5 has been in use for years. According to a 2014 Vision Mobile Survey, 42% of 10,000 developers surveyed are using the combination of HTML, CSS, and JavaScript for all or part of their mobile applications. Gartner identified HTML5 as one of their top 10 mobile technologies and capabilities for 2015 and 2016, saying HTML5 "will be an essential technology for organizations delivering applications across multiple platforms."

To help achieve the "write once, deploy anywhere" promise of HTML5 and the Open Web platform, during the 22 months since W3C announced the completed definition of HTML5, the W3C community has been adding to the HTML5 test suite, which includes over 100,000 tests and continues to grow. The Test the Web Forward community effort now plays an important and ongoing part in driving Open Web Platform interoperability.

With today's publication of the Recommendation, software implementers benefit from Royalty-Free licensing commitments from over sixty companies under W3C's Patent Policy. Enabling implementers to use Web technology without payment of royalties is critical to making the Web a platform for innovation.

What's Next: Application Foundations for Developers, New Use Cases for the Web
PAGE
visibility
Provê uma API que checa se a aba atual está visível ou não no navegador.
window.addEventListener('visibilitychange', () => {
  if (document.hidden) {
    console.log('Tab is hidden');
  }
  else {
    console.log('Tab is focused');
  }
});
Page Visibility Level 2
W3C Candidate Recommendation 28 March 2017

This version: https://www.w3.org/TR/2017/CR-page-visibility-2-20170328/

Latest published version: https://www.w3.org/TR/page-visibility-2/

Latest editor’s draft: https://w3c.github.io/page-visibility/

Implementation report: https://w3c.github.io/test-results/page-visibility/all.html

Previous version: https://www.w3.org/TR/2016 CR-page-visibility-2-20161122/

Editors:
Ilya Grigorik, Google, iligrigorik@gmail.com
Arvind Jain, Google Inc., (Until December 2014)
Jatinder Mann, Microsoft Corp. (Until February 2014)

Repository:
We are on GitHub.
File a bug.
Commit history.

Mailing list:
public-web-perf@w3.org

Implementation:
Can I use Page Visibility?
Test Suite
Test Suite repository

Copyright © 2017 W3C® (MIT, ERCIM, Keio, Beihang). W3C liability, trademark and permissive document license rules apply.
window.addEventListener('visibilitychange', () => {
    switch(document.visibilityState) {
        case 'prerender':
            console.log('Tab is pre-rendering');
            break;
        case 'hidden':
            console.log('Tab is hidden');
            break;
        case 'visible':
            console.log('Tab is focused');
            break;
    }
});
BROWSER support

caniuse.com/#feat=pagevisibility
ONDE usar?
AJUSTES NECESSÁRIOS

Pachequinho vê "coisas não acontecerem" para o Coritiba e critica decisões do árbitro

ATUAÇÕES

Dodô falha em gol, e Wilson evita goleada contra o Coritiba; confira as notas

Wilson lamenta derrota e destaca mérito do Grêmio
WEB

share
WEB SHARE

Permite que um site compartilhe algo para um determinado destino. Pode ser um serviço de sistema, um aplicativo nativo ou outro site.
shareButton.addEventListener('click', () => {
    navigator.share(
        {
            title: 'Some Example',
            url: window.location.href
        }
    ).then(console.log('Successful share'))
    .catch(console.log(error));
});
onde usar?
Web Share API
Draft Community Group Report 23 June 2017

Latest editor’s draft:
https://w3c.github.io/web-share/

Editor:
Matt Giuca, Google Inc.

Participate:
GitHub WICG/web-share
File a bug
Commit history

Implementation status:
Chromium

Copyright © 2017 the Contributors to the Web Share API Specification, published by the Web Incubator Community Group under the W3C Community Contributor License Agreement (CLA). A human-readable summary is available.

Abstract

This specification defines an API for sharing text, links and other content to an arbitrary destination of the user’s choice.

The available share targets are not specified here; they are provided by the user agent. They could, for example, be apps, websites or contacts.

Status of This Document

This specification was published by the Web Incubator Community Group. It is not a W3C Standard nor is it on the W3C Standards Track. Please note that under the W3C Community Contributor License Agreement (CLA) there is a limited end-esh and other conditions apply. Learn more about W3C Community and Business Groups.
online
STATE
ONLINE STATE

Expõe o estado atual de disponibilidade da rede.
ONLINE STATE

console.log(navigator.onLine ? 'online' : 'offline')
window.addEventListener('offline', networkStatus);
window.addEventListener('online', networkStatus);

function networkStatus(e) {
    console.log(e.type);
}
BROWSER support

caniuse.com/#feat=online-status
Network Information API
Living Document

Draft Community Group Report 13 June 2017

Latest editor's draft:
https://wicg.github.io/netinfo/

Editors:
Ilya Grigorik, Google
Marcos Cáceres, Mozilla Corporation
Fernando Jiménez Moreno, Telefonica

Participate:
GitHub WICG/netinfo
File a bug
Commit history

Implementations:
Chromium

Copyright © 2017 the Contributors to the Network Information API Specification, published by the Web Incubator Community Group under the W3C Community Contributor License Agreement (CLA). A human-readable summary is available.

Abstract

The Network Information API enables web applications to access information about the network connection in use by the device.

Status of This Document
let type = navigator.connection.type;
// ex: bluetooth, cellular, ethernet, wifi

let max = navigator.connection.downlinkMax;
// in megabits
ONDE usar?
DEVICE orientation
DEVICE ORIENTATION

Expõe as coordenadas de orientação física de um dispositivo.
window.addEventListener('deviceorientation', (e) => {
    console.log('Gamma:', e.gamma);
    console.log('Beta:', e.beta);
    console.log('Alpha:', e.alpha);
});
let logo = document.querySelector('img');

window.addEventListener('deviceorientation', (e) => {
  let tiltLR = e.gamma;
  let tiltFB = e.beta;

  logo.style.transform = `rotate(${tiltLR}deg)
  rotate3d(1,0,0, ${tiltFB * -1}deg)`;
});
BROWSER support

caniuse.com/#feat=deviceorientation
ONDE
usar?
Não tem quem duvide que Minas Gerais é uma terra de grandes belezas. Terra de montanhas que acolhem, de história que ensina e de um povo
VIBRATION
VIBRATION

Provê acesso ao hardware de vibração de dispositivos móveis.
// Vibrate for 1 second
navigator.vibrate(1000);

// Vibrate with a pattern
navigator.vibrate([400, 300, 300, 200, 500]);

// Turn off vibration
navigator.vibrate(0);
// Super Mario
navigator.vibrate([[125, 75, 125, 275, 200, 275, 125, 75, 125, 75, 200, 600, 200, 600]]);

// Star Wars
navigator.vibrate([[500, 110, 500, 110, 450, 110, 200, 110, 170, 40, 450, 110, 200, 110, 170, 40, 500]]);

// Go Go Power Rangers
navigator.vibrate([[150, 150, 150, 150, 75, 75, 150, 150, 150, 150, 75, 150, 450]]);

https://goo.gl/bX4ZQv
browser support

caniuse.com/#feat=vibration
ONDE usar?
SkiiFree.js

Fork me on GitHub

BLOG POST

Blogs:

- piste to control the player
- Skiiers left: 5
- High Score: 0

Created by Dan Hough (@basicallydan)
Current Speed: 5
CLIPBOARD

copy & paste
Clipboard

Possibilita interação com o clipboard do usuário através de operações de copiar, cortar e colar.
ZeroClipboard v2.x

The ZeroClipboard library provides an easy way to copy text to the clipboard using an invisible Adobe Flash movie and a JavaScript interface.

The "Zero" signifies that the library is invisible and the user interface is left entirely up to you.

Looking for v1.x?  Testing this page locally
let button = document.querySelector('button');

button.addEventListener('click', () => {
  select();
  copy();
});
// 2. Programatically select an element

function select() {
    let input = document.querySelector('input');

    input.focus();
    input.setSelectionRange(0, input.value.length);
}
// 3. Copy selected element text

function copy() {
    try {
        document.execCommand('copy');
    }
    catch (err) {
        console.error(err);
    }
}
document.addEventListener('copy', (e) => {
    console.log(e.target.value);
});

document.addEventListener('cut', (e) => {
    console.log(e.target.value);
});

document.addEventListener('paste', (e) => {
    console.log(e.clipboardData.getData('text/plain'));
});
clipboard.js

A modern approach to copy text to clipboard
No Flash. No frameworks. Just 3kb gzipped

Why

Copying text to the clipboard shouldn't be hard. It shouldn't require dozens of steps to configure or hundreds of KBs to load. But most of all, it shouldn't depend on Flash or any bloated
BROWSER support

caniuse.com/#feat=clipboard
ONDE usar?
Modern copy to clipboard. No Flash. Just 3kb gzipped - https://clipboardjs.com

216 commits
2 branches
22 releases

Latest commit 68b66887 on Jun 17

- .github
- demo
- dist
- src
- test
- .babelrc
- .banner
- .editorconfig
- .gitignore
- .npmignore
- .browserconfig
Components

Over a dozen reusable components built to provide buttons, dropdowns, input groups, navigation, alerts, and much more.

Pagination

Provide pagination links for your site or app with the multi-page pagination component.

Contents

- Overview
- Disabled and active states
- Sizing

Overview
ambient LIGHT
Expõe dados do sensor que capta a intensidade de luz de um ambiente.
window.addEventListener('devicelight', (e) => {
    console.log(`$${e.value} lux`);
});
Abstract

This specification defines a framework for exposing sensor data to the Open Web Platform in a consistent way. It does so by defining a blueprint for writing specifications of concrete sensors along with an abstract Sensor interface that can be extended to accommodate different sensor types.
let sensor = new AmbientLightSensor();
sensor.start();
sensor.onload = (e) => {
    console.log(e.reading.illuminance);
};
sensor.stop();
Browser support

caniuse.com/#feat=ambient-light
BATTERY STATUS

Permite que uma página web acesse informações sobre a bateria de um dispositivo.
navigator.getBattery().then((battery) => {
  console.log(`${battery.level * 100}%`);

  battery.addEventListener('levelchange', () => {
    console.log(`${this.level * 100}%`);
  });
});
BROWSER support

caniuse.com/#feat=battery-status
ONDE
usar?
Catch Omanyte!
You got 23min remaining until the battery dies.
WEB VR

Uma API experimental que trás o mundo de Realidade Virtual para web usando Oculus Rift ou Google Cardboard por exemplo.
BROWSER support

[Icons of different browsers: Safari, Firefox, Chrome, Opera, Edge]
SHAPE
detection
SHAPE DETECTION

É capaz de detectar diferentes formas em imagens como rostos, códigos de barras e até mesmo texto em uma imagem.
let faceDetector = new FaceDetector();

faceDetector.detect(image).then(faces => {
    console.log('Faces found:', faces.length);
}).catch((err) => {
    console.error(err);
});
Accelerated Shape Detection in Images
Editor's Draft, 10 May 2017

This version:
https://w3c.github.io/shape-detection-api

Issue Tracking:
GitHub

Editor:
Miguel Casas-Sanchez (Google Inc.)

Translations (non-normative and likely out-of-date):
简体中文

Participate:
Join the W3C Community Group
Fix the text through GitHub

Copyright © 2017 the Contributors to the Accelerated Shape Detection in Images Specification, published by the Web Platform Incubator Community Group under the W3C Community Contributor License Agreement (CLA): A human-readable summary is available.

Abstract

This document describes an API providing access to accelerated shape detectors (e.g., human faces) for still images and/or live image feeds.

Status of this document

This specification was published by the Web Platform Incubator Community Group. It is not a W3C Standard nor is it on the W3C Standards Track. Please note that under the W3C Community Contributor License Agreement (CLA) there is a limited opt-out and other conditions apply. Learn more about W3C Community and Business.
BROWSER support under development
ONDE usar?
tracking.js
A modern approach for Computer Vision on the web

The tracking.js library brings different computer vision algorithms and techniques into the browser environment. By using modern HTML5 specifications, we enable you to do real-time color tracking, face detection and much more — all that with a lightweight core (~7 KB) and intuitive interface. Learn more »

```
bower install tracking.js
```

Star 4,977 Fork 735

DOWNLOAD TRACKING.JS

Tag Friends
Iron Spheres
Racing Car
Draw Something
Random Particles
Face (Image)
Face (Camera)
Color (Image)
Color (Camera)
Color (Video)
Feature Detection
WebAssembly, ou wasm, é um novo formato binário para desenvolvimento de aplicações de baixo nível.
BROWSER support

under development
GAMEPAD

Permite que páginas web se conectem com controles de video game via USB.
window.addEventListener('gamepadconnected', () => {
  let gp = navigator.getGamepads()[0];

  console.log('ID:', gp.id);
  console.log('Axes:', gp.axes.length);
  console.log('Buttons:', gp.buttons.length);
});
window.addEventListener('gamepadconnected', gameLoop);

function gameLoop() {
    let gp = navigator.getGamepads()[0];

    if (gp.buttons[0].pressed) {
        console.log('X');
    }

    requestAnimationFrame(gameLoop);
}
BROWSER support

caniuse.com/#feat=gamepad
Where the magic happens

Your comfort zone
muito OBRIGADO @zenorocha