

WEBAPP介绍及其iOS Web开发技巧总结

标签：[WebApp](#) [iOS开发](#)

2013-08-26 19:17

46556人阅读

[评论\(2\)](#) [收藏](#) [举报](#)

分类：[前端开发 \(43\)](#)

版权声明：本文为博主原创文章，未经博主允许不得转载。

结合网上相关资料，以及自己项目中的经验，收集汇总了iOS Webapp相关的开发知识，如下。

WebApp是一种新出现的基于WEB形式的类应用程序，运行在高端的移动终端设备上，其应用范围会越来越广。

开发者们都知道在高端智能手机系统中有两种应用程序：一种是基于本地（**操作系统**）运行的APP；一种是基于高端机的浏览器运行的WebApp，本文将主要讲解后者。

WebApp与Native App有何区别呢？

Native App：

1、开发成本非常大。

一般使用的开发语言为**Java**、C++、**Objective-C**。

2、更新体验较差、同时也比较麻烦

每一次发布新的版本，都需要做版本打包，且需要用户手动更新（有些应用程序即使不需要用户手动更新，但是也需要有一个恶心的提示）。

3、非常酷

因为native app可以调用**ios**中的UI控件以UI方法，它可以实现WebApp无法实现的一些非常酷的交互效果

4、Native app是被Apple认可的

Native app可以被Apple认可为一款可信任的独立软件，可以放在Apple Store出售，但是Web app却不行。

Web App：

1、开发成本较低

使用**HTML5** + **CSS3** + **js** 等web开发技术就可以轻松的完成web app的开发。效果上面能够完全模拟传统应用程序效果。

2、升级较简单

由于不需要通过苹果商店发布，所以升级不需要通知用户，在服务端更新文件即可，用户完全没有感觉

3、维护比较轻松

和一般的web一样，维护比较简单，它其实就是一个站点

Webapp说白了就是一个针对Iphone、**Android**优化后的web站点，它使用的技术无非就是HTML或**html5**、**css3**、**JavaScript**，服务端技术**JAVA**、**PHP**、ASP。

当然，因为这些高端智能手机（Iphone、**android**）的内置浏览器都是基于webkit内核的，所以在开发WEBAPP时，多数都是使用**Html5**和**CSS3**技术做UI布局。当使用HTML5和CSS3做UI时，若还是遵循着一般web开发中使用HTML4和CSS2那样的开发方式的话，这也就失去了WEBAPP的本质意义了，且有些效果也无法实现的，所以在此又回到了我们的主题—webapp的布局方式和技术。

在此所说的移动平台**前端开发**是指针对高端智能手机（如Iphone、Android）做站点适配也就是WebApp，并非是针对普通手机开发 Wap 2.0，所以在阅读本篇文章以前，你需要对webkit内核的浏览器有一定的了解，你需要对HTML5和CSS3有一定的了解。如果你已经对此有所了解，那现在就开始往下阅读吧……

1、首先我们来看看webkit内核中的一些私有的meta标签，这些meta标签在开发webapp时起到非常重要的作用

```
1 <meta content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=0;" name="viewport" />
2 <meta content="yes" name="apple-mobile-web-app-capable" />
3 <meta content="black" name="apple-mobile-web-app-status-bar-style" />
4 <meta content="telephone=no,email=no" name="format-detection" />
```

第一个meta标签表示：强制让文档的宽度与设备的宽度保持1:1，并且文档最大的宽度比例是1.0，且不允许用户点击屏幕放大浏览；user-scalable定义是否可缩放（0为不缩放），使页面固定设备上面的大小。

（注意：据说HTC G7自身系统浏览器不支持这一条规则，能对页面进行放大，一旦放大导致页面布局错乱，解决方法：定义页面的最小宽度 min-width，body{min-width: 300px;}）

iOS 7.1的Safari为meta标签新增minimal-ui属性，在网页加载时隐藏地址栏与导航栏。<meta name="viewport" content="width=device-width, initial-scale=1, user-scalable=no, minimal-ui" />

第二个meta标签是ios设备(不只iphone)中的safari私有meta标签，它表示：允许全屏模式浏览，开启对Web Aapp程序的支持。；

第三个meta标签也是ios系统的私有标签，它指定在web app状态下，ios设备中顶端的状态条的颜色；默认值为default（白色），可以定为black（黑色）和black-translucent（灰色半透明）。若值为"black-translucent"将会占据页面px位置，浮在页面上方（会覆盖页面20px高度-iphone4和itouch4的Retina屏幕为40px）。

第四个meta标签表示：使设备浏览网页时对数字不启用电话功能（不同设备解释不同，itouch点击数字为存入联系人，iphone为拨打电话），忽略将页面中的数字识别为电话号码。

若需要启用电话功能将telephone=yes即可，具体调用格式可以这样书写代码Call Me，若在页面上面有google maps，iTunes和youtube的链接会在ios设备上打开相应的程序组件。

2、HTML5标签的使用

在开始编写webapp时，哥建议前端工程师使用HTML5，而放弃HTML4，因为HTML5可以实现一些HTML4中无法实现的丰富的WEB应用程序的体验，可以减少开发者很多的工作量，当然了你决定使用HTML5前，一定要对此非常熟悉，要知道HTML5的新标签的作用。比如定义一块内容或文章区域 可使用section标签，定义导航条或选项卡可以直接使用nav标签等等。

3、放弃CSS float属性

在项目开发过程中可以会遇到内容排列排列显示的布局(见下图)，假如你遇见这样的视觉稿，哥建议你放弃float，可以直接使用display:block;

4、利用CSS3边框背景属性

这个按钮有圆角效果，有内发光效果还有高光效果，这样的按钮使用CSS3写是无法写出来的，当然圆角可以使用CSS3来写，但高光和内发光却无法使用CSS3编写，这个时候你不妨使用-webkit-border-image来定义这个按钮的样式。-webkit-border-image就个很复杂的样式属性。

5、块级化a标签

请保证将每条数据都放在一个a标签中，为何这样做？因为在触控手机上，为提升用户体验，尽可能的保证用户的可点击区域较大。

6、自适应布局模式

在编写CSS时，我不建议前端工程师把容器（不管是外层容器还是内层）的宽度定死。为达到适配各种手持设备，我建议前端工程师使用自适应布局模式（支付宝 采用了自适应布局模式），因为这样做可以让你的页面在ipad、itouch、ipod、iphone、android、web safarik、chrome都能够正常的显示，你无需再次考虑设备的分辨率。

7、学会使用webkit-box

上一节，我们说过自适应布局模式，有些同学可能会问：如何在移动设备上做到完全自适应呢？很感谢webkit为display属性提供了一个webkit-box的值，它可以帮助前端工程师做到盒子模型灵活控制。

8、如何去除Android平台中对邮箱地址的识别

看过iOS webapp API的同学都知道iOS提供了一个meta标签:用于禁用iOS对页面中电话号码的自动识别。在iOS中是不自动识别邮件地址的，但在Android平台，它会自动检测邮件地址，当用户touch到这个邮件地址时，Android会弹出一个框提示用户发送邮件，如果你不想Android自动识别页面中的邮件地址，你不妨加上这样一句meta标签在head中 1 <meta content="email=no" name="format-detection" />

9、如何去除iOS和Android中的输入URL的控件条

你的老板或者PD或者交互设计师可能会要求你：能否让我们的webapp更加像nativeapp，我不想让用户看见那个输入url的控件条？

答案是可以做到的。我们可以利用一句简单的javascript代码来实现这个效果

```
1 setTimeout(scrollTo,0,0,0);
```

请注意，这句代码必须放在window.onload里才能够正常的工作，而且你的当前文档的内容高度必须是高于窗口的高度时，这句代码才能有效的执行。

10、如何禁止用户旋转设备

我曾经也想禁止用户旋转设备，也想实现像某些客户端那样：只能在肖像模式或景观模式下才能正常运行。但现在我可以很负责的告诉你：别想了！在移动版的webkit中做不到！

至少Apple webapp API已经说到了：我们为了让用户在safari中正常的浏览网页，我们必须保证用户的设备处于任何一个方位时，safari都能够正常的显示网页内容（也就是自适应），所以我们禁止开发者阻止浏览器的orientationchange事件，看来苹果公司的出发点是正确的，苹果确实不是一般的苹果。

iOS已经禁止开发者阻止orientationchange事件，那Android呢？对不起，我没有找到任何资料说Android禁止开发者阻止浏览器orientationchange事件，但是在Android平台，确实也是阻止不了的。

11、如何检测用户是通过主屏启动你的webapp

看过Apple webapp API的同学都知道iOS为safari提供了一个将当前页面添加主屏的功能，按下iphoneipodipod touch底部工具中的小加号，或者ipad顶部左侧的小加号，就可以将当前的页面添加到设备的主屏，在设备的主屏会自动增加一个当前页面的启动图标，点击该启动图标就可以快速、便捷的启动你的webapp。从主屏启动的webapp和浏览器访问你的webapp最大的区别是它清除了浏览器上方和下方的工具条，这样你的webapp就更加像是nativeapp了，还有一个区别是window对象中的navigator对象的一个standalone属性。iOS中浏览器直接访问站点时，navigator.standalone为false,从主屏启动webapp时，navigator.standalone为true，我们可以通过navigator.standalone这个属性获知用户当前是否是从主屏访问我们的webapp的。在Android中从来没有添加到主屏这回事！

12、如何关闭iOS中键盘自动大写

我们知道在iOS中，当虚拟键盘弹出时，默认情况下键盘是开启首字母大写的功能的，根据某些业务场景，可能我们需要关闭这个功能，移动版本webkit为input元素提供了autocapitalize属性，通过指定autocapitalize="off"来关闭键盘默认首字母大写。

13、iOS中如何彻底禁止用户在新窗口打开页面

有时我们可能需要禁止用户在新窗口打开页面，我们可以使用a标签的target="_self"来指定用户在新窗口打开，或者target属性保持空，但是你会发现iOS的用户在这个链接的上方长按3秒钟后，iOS会弹出一个列表按钮，用户通过这些按钮仍然可以在新窗口打开页面，这样的话，开发者指定的target属性就失效了，但是可以通过指定当前元素的-webkit-touch-callout样式属性为none来禁止iOS弹出这些按钮。这个技巧仅适用iOS对于Android平台则无效。

14、iOS中如何禁止用户保存图片\复制图片

我们在第13条技巧中提到元素的-webkit-touch-callout属性，同样为一个img标签指定-webkit-touch-callout为none也会禁止设备弹出列表按钮，这样用户就无法保存\复制你的图片了。

15、iOS中如何禁止用户选中文字

我们通过指定文字标签的-webkit-user-select属性为none便可以禁止iOS用户选中文字。

16、iOS中如何获取滚动条的值

桌面浏览器中想要获取滚动条的值是通过document.scrollTop和document.scrollLeft得到的，但在iOS中你会发现这两个属性是未定义的，为什么呢？因为在iOS中没有滚动条的概念，在Android中通过这两个属性可以正常获取到滚动条的值，那么在iOS中我们该如何获取滚动条的值呢？通过window.scrollToY和window.scrollToX我们可以得到当前窗口的y轴和x轴滚动条的值。

17、如何解决盒子边框溢出

当你指定了一个块级元素时，并且为其定义了边框，设置了其宽度为100%。在移动设备开发过程中我们通常会对文本框定义为宽度100%，将其定义为块级元素以实现全屏自适应的样式，但此时你会发现，该元素的边框(左右)各1个像素会溢了文档，导致出现横向滚动条，为解决这一问题，我们可以为其添加一个特殊的样式-webkit-box-sizing:border-box;用来指定该盒子的大小包括边框的宽度。

18、如何解决Android 2.0以下平台中圆角的问题

如果大家够细心的话，在做wap站点开发时，大家应该会发现android 2.0以下的平台中问题特别的多，比如说边框圆角这个问题吧。

在对一个元素定义圆角时，为完全兼容android 2.0以下的平台，我们必须按照以下技巧来定义边框圆角：

1 \ -webkit这个前缀必须要加上（在iOS中，你可以不加，但android中一定要加）；

2 \ 如果对针对边框做样式定义，比如border:1px solid #000;那么-webkit-border-radius这属性必须要出现在border属性后。

3 \ 假如我们有这样的视觉元素，左上角和右上角是圆角时，我们必须要先定义全局的(4个角的圆角值)-webkit-border-radius:5px;然后再依次覆盖左下角和右下角，-webkit-border-bottom-left-radius:0;-webkit-border-bottom-right-radius:0;否则在android 2.0以下的平台中将全部显示直角，还有记住！-webkit这个前缀一定要加上！

19、如何解决android平台中页面无法自适应

虽然你的html和css都是完全自适应的，但有一天如果你发现你的页面在android中显示的并不是自适应的时候，首先请你确认你的head标签中是否包含以下meta标签：

```
1 <meta name="viewport" content="width=device-width,initial-scale=1.0,maximum-scale=1.0,user-scalable=0;" />
```

如果有的话，那请你再仔细的看清楚有没有这个属性的值width=device-width，如果没有请立即加上吧！

20、如何解决iOS 4.3版本中safari对页面中5位数字的自动识别和自动添加样式

新的iOS系统也就是4.3版本，升级后对safari造成了一个bug：即使你添加了如下的meta标签，safari仍然会对页面中的5位连续的数字进行自动识别，并且将其重新渲染样式，也就是说你的css对该标签是无效的。

```
1 <meta name="format-detection" content="telephone=no" />
```

我们可以用一个比较龌龊的办法来解决。比如说支付宝wap站点中显示金额的标签，我们都做了如下改写：

```
1 <button class="t-balance" style="background:none;padding:0;border:0;">95009.00</button>元
```

21、如何检测iOS4、iOS5或是iOS6？

iPhone 4带来的革新，retina display绝对是最吸引眼球的一项。正是依赖这视网膜显示屏，iPhone 4的分辨率达到了640×960 pixels，不过为了保持向下兼容性，它采用的仍然是320×480 points。也就是说，在不进行缩放的情况下，显示普通图片时，它会用4个像素来显示图片中的1个像素；而在显示retina图片时，每个像素都对应图片中的1个像素。

如此一来，老的应用无需修改就可以在iPhone 4上运行了——虽然显示效果差了点，但是不会出现只有左上角那1/4的区域有内容的情况。

iOS 6发布有许多新的变化，比如smart app banner功能，新版iPhone屏幕也由3:2变长为16:9，分辨率从iPhone 4s的640px*960变成了640*1136，DPI依然是326。平时我们习惯用useragent检测，返回：

```
Mozilla/5.0 (iPhone; CPU iPhone OS 6_0 like Mac OS X) AppleWebKit/536.26 (KHTML, like Gecko) Version/6.0
```

但是，这个方法无法分辨是刷机升级到iOS 6的低版本iPhone，还是最新的原装iPhone 5，这里用javascript或者通过CSS3的Media Queries特性进行区分。

在网页中，pixel与point比值称为device-pixel-ratio，普通设备都是1，iPhone 4是2，有些Android机型是1.5。

那么-webkit-min-device-pixel-ratio:2可以用来区分iphone(4/4s/5)和其它的手机

iPhone4/4s的分辨率为640*960 pixels，DPI为是326，设备高度为480px

iPhone5的分辨率为640*1136 pixels，DPI依然是326，设备高度为568px

那么我们只需要判断iphone手机的device-height(设备高)值即可区别iPhone4和iPhone5

方式一，直接写到样式里面

```
/*由于iPhone4inches = (window.screen.height==568);*/
@media (device-height:480px) and (-webkit-min-device-pixel-ratio:2){/* 兼容iphone4/4s */
.class{
}
}
@media (device-height:568px) and (-webkit-min-device-pixel-ratio:2){/* 兼容iphone5 or iPod Touch 5th generation*/
.class{
}
```

```
}
```

方式二，链接到一个单独的样式表，把下面的代码放在<head>标签里

```
<link rel="stylesheet" media="(device-height: 480px) and (-webkit-min-device-pixel-ratio:2)" href="iphone4.css" />
<link rel="stylesheet" media="(device-height: 568px) and (-webkit-min-device-pixel-ratio:2)" href="iphone5.css" />
```

使用JS

```
//通过高度来判断是否是iPhone 4还是iPhone 5
isPhone4inches = (window.screen.height==480);
isPhone5inches = (window.screen.height==568);
```

22、如何实现Web App添加到主屏幕

如果你之前通过apple-mobile-web-app-capable这个meta标签来将网页添加到主屏幕的话，这种方法只支持iPhone 的3.5 " 屏幕，而iPhone 5会比较悲催，页面顶部和顶部会出现一条黑色区域。即便你提供了一个大尺寸的启动界面(640*1096)，iPhone 5依然会将其压缩至640*920。

对iOS6的解决方案：

你需要放弃之前使用的viewport属性width=device-width或者width=320。如果你不指定viewport，它也可以很正常的显示：

```
<meta name="viewport" content="initial-scale=1.0">
```

或者你也可以指定一个非320的宽度：

```
<meta name="viewport" content="width=320.1">
```

如果你不想影响iPhone 4s以前的safari，也可以用js动态设置viewport：

```
if (window.screen.height==568) { // iPhone 4"
document.querySelector("meta[name=viewport]").content="width=320.1";
}
```

对于启动画面，可以用media query来调整：

```
<link href="startup-568h.png" rel="apple-touch-startup-image" media="(device-height: 568px)">
<link href="startup.png" rel="apple-touch-startup-image" sizes="640x920" media="(device-height: 480px)">
```

23、如何设置webapp的主屏默认标题？

iOS6可以通过meta标签来给主屏webapp指定标题：

```
<meta name="apple-mobile-web-app-title" content="XXX网站欢迎你">
```

24、iOS6文件上传限制

iOS6引入了HTML5 文件上传API，支持单个文件/多文件上传。

```
<input type="file"><!--单文件-->
<input type="file" multiple><!--多文件-->
```

但是，由于iOS的资源管理机制的限制，你只能上传照片和视频，不能上传其它格式文件，也不支持getUserMedia api(camera api)。

25、iOS广告条Smart App Banner

如果你的网站同时提供的有app在itunes app store，可以通过一个简单的meta标签来提示用户，让用户下载安装你的native app(或者是hybrid app)：

```
<meta name="apple-itunes-app" content="app-id=xxx">
```

也支持itunes affiliate program (推广联盟) :

```
<meta name="apple-itunes-app" content="app-id=9999999, app-argument=xxxxxx">
```

```
<meta name="apple-itunes-app" content="app-id=9999999, app-argument=xxxxxx, affiliate-data=partnerId=99&siteID=XXXX">
```

需要注意的是, app banner占位为156px的高度——高分屏为312px。

26、CSS3 Filter

详细请阅读《-webkit-filter是神马?》<http://www.qianduan.NET/what-is-webkit-filter.html>

27、Safari 全屏

这个很赞, 在横屏模式下, 点击Safari右下角的icon即可全屏。有些像 Mac OS X 的全屏方法, 缺点是只支持横屏的场景。

28、Animation Timing API

这个对游戏开发者非常有用, 详情可访问Animation Timing API (<http://www.w3.org/TR/animation-timing/>), 也可以看下Paul Irish的这个教程<http://www.paulirish.com/2011/requestanimationframe-for-smart-animating/>。

29、CSS image set

考虑到位用户节省流量, iOS6这个属性非常有用。你可以为CSS选择器指定多个特定的图片, 用于区分高分屏和非高分屏:

```
-webkit-image-set(url(low.png) 1x, url(hi.jpg) 2x)
```

可以支持background-image之类的属性。

这个是webkit私有属性, 其它浏览器不支持。希望os x下也开始支持吧, 这样就不用 background-size来区分视网膜屏了。

30、CSS 3 cross-fade

iOS 6支持一些最新的CSS3 image values标准, 包括cross-fade。这样我们可以在同一个选择器上使用多张图片, 以实现半透明或者动画的效果:

```
background-image: -webkit-cross-fade(url("logo1.png"), url("logo2.png"), 50%);
```

31、Web View更新:

需要注意的是, webview中的javascript速度, 比nitro引擎的Safari慢3.3倍。

嗯, 上面说的hybrid模式就是用webview组件封装的webapp, phonegap之类的第三方开发工具即是这种模式。

32、同步调试

iOS 6中Safari和webview, 支持用桌面Safari同步调试了。像在pc端上一样调试webapp或者hybrid app对前端开发者无疑是极大的方便。方法很简单:

第一步, 手机上设置Safari开启 web inspector (设置->safari->高级)

第二步, 手机连上电脑

第三部, 打开电脑上的Safari, 然后菜单-》开发, 即可看到设备。点击即可调试。

33、link标签apple-touch-icon

通过设置相应的apple-touch-icon标签, 添加到主屏幕的图标就会使用我们指定大小的图片。以下是针对ios不同设备的设置方法。

```
<link rel="apple-touch-icon" href="touch-icon-iphone.png">
```

```
<link rel="apple-touch-icon" sizes="76x76" href="touch-icon-ipad.png">
```

```
<link rel="apple-touch-icon" sizes="120x120" href="touch-icon-iphone-retina.png">
```

```
<link rel="apple-touch-icon" sizes="152x152" href="touch-icon-ipad-retina.png">
```

(默认iphone的大小为60px, ipad1为76px, retina屏则是2倍大小。)

图标搜索的优先级如下：

如果没有跟相应设备推荐尺寸一致的图标，那个会优先使用比推荐尺寸大，但最接近推荐尺寸的图标。

如果没有比推荐尺寸大的图标，会优先选择最接近推荐尺寸的图标。

如些有多个图标符合推荐尺寸，会优先选择包含关键字precomposed的图标。

如果未在区域指定用link标签指定图标，会自动搜索网站根目录下有apple-touch-icon...或者 apple-touch-icon-precomposed...前缀的图标。

注：ios7不再为icon添加特效（圆角和高亮），ios7以前则默认为icon添加特效，，如果不希望系统添加特效，则可以将apple-touch-icon.png替换为apple-touch-icon-precomposed.png

34、apple-touch-startup-image

apple-touch-startup-image是用来为WebApp设置一个类似NativeApp的启动画面。使用方法为：

```
<link rel="apple-touch-startup-image" href="/startup.png">
```

不过和apple-touch-icon不同，apple-mobile-web-app-capable不支持sizes属性，所以使用media来控制普屏、retina屏和横竖屏分别加载不同的启动画面。

```
[html]
01. // iPhone
02. <link href="apple-touch-startup-image-320x460.png" media="(device-width: 320px)" rel="apple-touch-startup-image" />
03.
04. // iPhone Retina
05. <link href="apple-touch-startup-image-640x920.png" media="(device-width: 320px) and (-webkit-device-pixel-ratio: 2)" rel="apple-
    touch-startup-image" />
06.
07. // iPhone 5
08. <link rel="apple-touch-startup-image" media="(device-width: 320px) and (device-height: 568px) and (-webkit-device-pixel-
    ratio: 2)" href="apple-touch-startup-image-640x1096.png">
09.
10. // iPad portrait
11. <link href="apple-touch-startup-image-768x1004.png" media="(device-width: 768px) and (orientation: portrait)" rel="apple-touch-
    startup-image" />
12.
13. // iPad landscape
14. <link href="apple-touch-startup-image-748x1024.png" media="(device-width: 768px) and (orientation: landscape)" rel="apple-touch-
    startup-image" />
15.
16. // iPad Retina portrait
17. <link href="apple-touch-startup-image-1536x2008.png" media="(device-width: 1536px) and (orientation: portrait) and (-webkit-
    device-pixel-ratio: 2)" rel="apple-touch-startup-image" />
18.
19. // iPad Retina landscape
20. <link href="apple-touch-startup-image-1496x2048.png" media="(device-width: 1536px) and (orientation: landscape) and (-webkit-
    device-pixel-ratio: 2)" rel="apple-touch-startup-image" />
```

参考文献：meta标签的使用：

<http://developer.apple.com/library/safari/#documentation/appleapplications/reference/SafariHTMLRef/Articles/MetaTags.html>

WebApp配置：

<http://developer.apple.com/library/safari/#documentation/AppleApplications/Reference/SafariWebContent/ConfiguringWebApplications/ConfiguringWeb/>