**平台前端开发规范**

**一、规范目的**

为提高团队协作效率，便于开发人员添加功能及前端后期优化维护，输出高质量的文档，特制订此文档。

**二、基本准则**

　　符合web标准，语义化html，结构表现行为分离，兼容性优良。页面性能方面，代码要求简洁明了有序，尽可能的减小服务器负载，保证最快的解析速度。

**三、文件规范**

1. html、css、js、images文件均归档至<平台js框架使用教程.docx>约定的目录中；

2. html文件命名：英文命名，后缀.html，将同一子模块的模板放于同目录中，模块入口模板以main.html或index.html命名；

3. css文件命名: 英文命名，后缀.css；

4. js文件命名: 英文命名，后缀.js，共用common.js（全局变量comm），入口js以main.js命名；

**四、html书写规范**

1. 文档类型声明及编码: 统一为html5声明类型<!DOCTYPE html>；编码统一为<meta charset="utf-8"/>， 书写时利用编辑器实现层次分明的缩进；

2. 非特殊情况下样式文件必须外链至<head>...</head>之间；非特殊情况下JavaScript文件必须外链至页面底部；

3. 引入样式css文件或JavaScript文件时，须略去默认类型声明，写法如下:

<link rel="stylesheet" href="..."/>

<style>...</style>

<script src="..."></script>

4. 引入JS库文件，文件名须包含库名称及版本号及是否为压缩版，比如jquery-1.4.1.min.js；引入插件，文件名格式为库名称+插件名称，比如jQuery.cookie.js；

5. 所有编码均遵循xhtml标准，标签&属性&属性命名必须由小写字母及下划线数字组成，且所有标签必须闭合，包括 br (<br/>)，hr(<hr />)等；属性值必须用双引号包括；

6. 充分利用无兼容性问题的html自身标签，比如span，em，strong，optgroup，label等等；需要为html元素添加自定义属性的时候，首先要考虑下有没有默认的已有的合适标签去设置，如果没有，可以使用须以"data-"为前缀来添加自定义属性，避免使用"data:"等其他命名方式；

7. 语义化html，如标题根据重要性用h\*(同一页面只能有一个h1)，段落标记用p，列表用ul，内联元素中不可嵌套块级元素；

8. 尽可能减少div嵌套， 如<div class="box"><div class="welcome">欢迎访问XXX， 您的用户名是<div class="name">用户名</div></div></div>完全可以用以下代码替代: <div class="box"><p>欢迎访问XXX，您的用户名是<span>用户名</span></p></div>；

9. 书写链接地址时，必须避免重定向，例如：href="http://itaolun.com/"，即须在URL地址后面加上“/”；

10. 在页面中尽量避免使用style属性，即style="…"；

11. 必须为含有描述性表单元素(input，textarea)添加label，如<p>姓 名: <input type="text" id="name" name="name"/></p>须写成:<p><label for="name">姓 名: </label><input type="text" id="name" /></p>

12. 能以背景形式呈现的图片，尽量写入css样式中；

13. 重要图片必须加上alt属性；给重要的元素和截断的元素加上title；

14. 给区块代码及重要功能(比如循环)加上注释，方便添加功能；

15. 特殊符号使用: 尽可能使用代码替代: 比如 <(<) & >(&gt；) & 空格( ) & ?(?) 等等；

16. 书写页面过程中，请考虑向后扩展性；

17. class & id 参见 css书写规范.

**五、css书写规范**

1. 编码统一为utf-8；

2. 协作开发及分工: 根据各个模块，同时根据页面相似程序， 事先写好大体框架文件， 分配给前端人员实现内部结构&表现&行为；框架中的css文件不可随意修改，变动样式可以自定义覆盖样式；

3. class与id的使用: id是唯一的并是父级的，class是可以重复的并是子级的，所以id仅使用在大的模块上，class可用在重复使用率高及子级中；id原则上都是分发框架文件时命名的，为JavaScript预留钩子的除外；

4. 为JavaScript预留钩子的命名，请以js\_起始，比如: js\_hide，js\_show；

5. class与id命名:样式名称由 小写英文、数字、\_、-来组合命名，如ace-nav，i\_comment， fontred，w200；避免使用中文拼音，尽量使用简易的单词组合；总之，命名要语义化，简明化。

6. css属性书写顺序，建议遵循: 布局定位属性-->自身属性-->文本属性-->其他属性. 此条可根据自身习惯书写，但尽量保证同类属性写在一起。

7. 书写代码前，考虑并提高样式重复使用率；

8. 充分利用html自身属性及样式继承原理减少代码量，比如:

<ul class="list"><li>这儿是标题列表<span>2010-09- 15</span></ul>

定义：ul.list li{position:relative} ul.list li span{position:absolute;right:0}

9. 样式表中中文字体名，请务必转码成unicode码，以避免编码错误时乱码；

10. 背景图片请尽可能使用sprite技术，减小http请求，考虑到多人协作开发，sprite按模块制作；

11.使用table标签时(尽量避免使用table标签)，请不要用width/ height/cellspacing/cellpadding等table属性直接定义表现，应尽可能的利用table自身私有属性分离结构与表现；

12. 使用<meta http-equiv="X-UA-Compatible" content="IE=edge" />；

13. 避免兼容性属性的使用，比如text-shadow || css3的相关属性；

14. 减少使用影响性能的属性，比如position:fixed、background-position: fixed、border-radius、background-size、box-shadow、gradients；

15. 必须为大区块样式添加注释，小区块适量注释；

16. 代码缩进与格式: 建议单行书写，可根据自身习惯，生产环境的css文件可做压缩；

**六、JavaScript书写规范**

1. 文件编码统一为utf-8，书写过程过，每行代码结束必须有分号；

2. 库引入: 原则上仅引入框架中已经引入的库，若需引入第三方库，须与团队其他人员讨论决定；

3. 变量命名: 驼峰式命名。原生JavaScript变量纯英文字母， 首字母须小写，如iTaoLun；jQuery变量首字符为'$'，其他与原生JavaScript 规则相同，如: $iTaoLun；

除了框架中声明的全局变量，避免在模块中定义全局变量。

4. 类命名: 首字母大写，驼峰式命名，如ITaoLun；

5. 函数命名: 首字母小写驼峰式命名，如iTaoLun()；

6. 命名语义化，尽可能利用英文单词或其缩写；

7. 尽量避免使用存在兼容性及消耗资源的方法或属性，比如eval() & innerText；

8. 后期优化中，JavaScript非注释类中文字符须转换成unicode编码使用， 以避免编码错误时乱码显示；

9. 代码结构明了，加适量注释，提高函数重用率；

10. 注重与html分离，减小reflow，注重性能。

**七、图片规范**

1. 所有页面元素类图片均放入images文件夹，测试用图片放于images/demo文件夹；

2. 图片格式仅限于gif、png、jpg；

3. 命名全部用小写英文字母、数字、下划线的组合，其中不得包含汉字、空格、特殊字符；尽量用易懂的词汇，便于团队其他成员理解；命名分头尾两部分，用下划线隔开，比如ad\_left01.gif或btn\_submit.gif；

4. 在保证视觉效果的情况下选择最小的图片格式与图片质量，以减少加载时间；

5. 尽量避免使用半透明的png图片；

6. 运用css sprite技术集中小的背景图或图标，减小页面http请求，请务必在对应的sprite psd源图中划参考线，并保存至images目录下的psd文件夹。

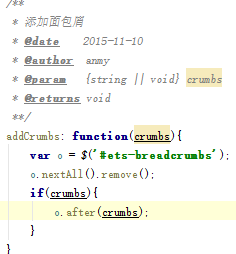
**八、注释规范**

1. html注释: 注释格式 <!--这儿是注释-->，'--'只能在注释的始末位置，不可置入注释文字区域；

2. css注释: 注释格式 /\*这儿是注释\*/；

3. JavaScript注释，单行注释使用//这儿是单行注释，多行注释使用/\* 这儿有多行注释 \*/；

4．Js函数注释，采用多行注释，如下例：



**九、开发及测试工具约定**

开发工具使用WebStorm，亦可根据自己喜好选择，但编码必须格式化，比如缩进；

测试: 前期开发测试FireFox、Chrome、IE8+，后期优化时加入Opera、Safari及国产的其他浏览器版本。

**十、其他规范**

1. 开发过程中按分工完成页面，尽量使用框架中已经定义的css，避免重复开发；

2. 减小沉冗代码，书写所有人都可以看的懂的代码，简洁易懂是一种美德，为用户着想，为服务器着想。