页面构建规范

1、布局示例图

2、样式统一

不同的浏览器在边距、字体、颜色方面会有些细微差异，现在主流以reset代码将所有的浏览器呈现基础样式加以统一。

在resource目录下，可以找到reset4pcweb.css以及reset4mobile.css，这两个文件分别对应了桌面端阅读以及移动端阅读的reset代码。

reset代码并不是死板不允许修改的，应该根据实际项目需要对其做修改，特别是字体大小、默认字体颜色、连接颜色之类的样式，以减少后期修改样式的情况。移动端项目也需要，但基本上不需要怎么改，需要考虑到的是使用rem来做单位，它是指相对HTML元素的字体大小，建议为HTML标签设置字体大小为10px或者20px，即1rem=10px或者1rem=20px。至于设置为20px的原因是chrome模拟器不支持渲染10px大小的文字，设置为20px会方便调试一点。

3、文件规范

从功能上划分，并且从减少http请求方面考虑，我们应该把全站的样式代码区分为三种：

* 通用模块：common.css
* 页面模块：page\_xxx.css
* 皮肤风格：themes.css或theme\_xxx.css

通用模块主要包含了reset代码、全局模块、全局布局等到处在用的代码。

页面模块是在通用模块的基础上，针对页面特有的私有模块做补充，并且按照页面本身实际情况针对使用到的通用模块加以样式调整，它的起名应该是跟当前页面一致，但前边加了个page前缀以表示该文件是用于某个页面的。

皮肤风格文件并不是所有项目都会有，当项目有皮肤风格变换的需求时，可以考虑两种实现：

1. 皮肤风格少并且固定，可以考虑把全部皮肤代码写到themes.css中。
2. 皮肤风格格多，并且经常增删减的情况下，应该根据皮肤的名称建立theme\_xxx.css来分开管理

一般来说我们只需要建立一个通用模块文件，每个页面再加一个对应的page\_xxx.css即可实现需求。

在实际应用中，请特别关注一下尽量不要加载太多css文件。

4、页面布局

注意：页面布局元素由于有时候确实无法为其定下合适的名称，允许在规范内为它们启用数字索引来区分。

同时也提倡为内容区（pbd）、布局容器（layout）起一个更语义化的名称，而放弃使用数字索引。

一般来说，页面将会分成三个部份：页首（.phd）、内容区（.pbd）、页尾（.pft）。请于common.css中定义这三者的通用样式。

其中，页首和页尾一般来说一个页面只会出现一次，而内容区可能会有多个，可以给它再添加一个pbd-index来标识，通过.pbd-1可以限定某个内容区中的某个模块的样式。

例子：

<div class=“pbd pbd-1”></div>

在某个区域下，往往会有上下左右的布局区域，每个布局容器存放了多个页面模块，但区域本身并没有比如合适的名称，可以使用.layout-index来标识，每个内容区中的layout都从1开始递增，不同内容区的layout可以使用同样的索引。

例子：

<div class=“pbd pbd-1”>

<div class=“layout-1”></div>

<div class=“layout-2”></div>

</div>

<div class=“pbd pbd-2”>

<div class=“layout-1”></div>

<div class=“layout-2”></div>

</div>

在对layout做样式设置时，请限定它属于哪个内容区：

.pbd-1 .layout-1 { float: left; }

5、模块定义

除去布局元素之外，在页面构建过程中应该为页面按功能划分好UI模块，并以m为前缀命名，比如m-feed、m-topNav。

模块的定义应该考虑到它是否有可能被用于其它位置，通用模块应该多考虑可移植性，允许它基本上不需要有样式调整即可轻松移植到其它位置（比如宽度自适应）。

模块内部结构如果属于中规中矩的上中下结构，建议使用.mhd、.mbd、.mft来起class名。

另外，模块名称建议使用以下词汇并做适当调整进行命名：

容器：box

侧栏：sidebar

栏目：column

主体：main

导航：nav

子导航： nav-sub

列表：list

图标：icon

列表项：item

标题：title

菜单：menu

子菜单：menu-sub

摘要：summary

操作：opra

按钮：btn/button

提示：msg/message

浮层：tips

标签页：tab

当前：current

6、测试环境

前端页面一般需要在以下两个环境阅读：

桌面端浏览器：一般需要兼容IE6+、safari、chrome以及firefox，并且针对类似于360安全浏览器等国产双内核浏览器进行测试，建议使用业界所谓的优雅降级方式实现兼容，即旧版本浏览器上与产品确认放弃一些新特性，以一些降级方式去实现，但不得影响美观以及功能的正常使用。代表性例子是圆角边框、阴影以及CSS3动画等功能。建议安装WINXP+IE6、WIN7+IE11两种虚拟机用于测试，如果涉及到一些安全权限方面的功能（比如说上传图片前预览），用IE11模拟其它版本并不一定可以做到安全权限一致，这时候还是需要使用真实版本的浏览器进行测试。

移动端浏览器：开发阶段必须保证于Chrome的移动端模拟器上正常使用，确认无误后，应该在常用的苹果手机、安卓手机上进行测试。

7、移动端高清屏适配

移动端（手机、pad）一般会有高清屏，比如iPhone 5s使用的是2倍高清屏，iPhone 6s使用的是3倍高清屏，即字体大小、图片尺寸都必须乘以2或者3。

请在设计初期就应该跟设计师沟通这方面的问题，并要求给出2x以及3x的PSD图。

另外，在css书写上，也必须考虑到这方面的问题，需要用媒体查询语句做调整。

8、设计原则

代码结构尽量清晰简洁，标签语义化，关注SEO知识，不遗漏页面标题、keyword之类的页面描述信息，以做到尽可能被搜索引擎识别并索引。

另外，可以用相对定位的时候尽量用相对定位，而不用浮动。

使用浮动时一定要记得使用统一的clearfix清浮动，尽量不要用overflow:hidden来处理，以免对容器内的浮层造成不良影响。

当设计到处被复用的小型通用模块（比如美观的文本框）时，一定要注意它经常被嵌入在其它模块中，注意不要起经常会冲突的内部class。