Web 前端开发规范
应用指南

版本 2.0

驴妈妈旅游网
www.lvmama.com

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。本规范文档一经确认，前端开发人员必须按本文档规范进行前台页面开发。本文档如有不对或者不合适的地方请及时提出，经讨论决定后方可更改。

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。

文件规范

文件命名一般采用英文，长度一般不超过 20 个字符，命名才用小写字母。除特殊情况才使用中文拼音。

一些常见的文件夹命名如：images（存放图形文件），flash（存放 flash 文件），style（存放 CSS 文件），scripts（存放 javascript 脚本），inc（存放 include 文件），media（存放多媒体文件）等。

文件名称统一用小写的英文字母、数字和下划线的组合。命名原则的指导思想一是使用你自己和工作组的每一个成员能够方便的理解每一个文件的意义，二是当我们在文件夹中使用“按名称排序”的命令时，同一种大类的文件能够排列在一起，以便我们查找、修改、替换、计算负载量等操作。

html 书写规范

1. 文档类型声明及编码: 统一用<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">; 编码统一为<meta charset="charset=utf-8" />, 书写时实现层次分明的缩进;目的：统一性和网站提高开发合作效率。
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</body>之前；目的：网站的优化。
3. 所有编码均遵循 xhtml 标准，标签&属性&属性命名必须由字母及下划线数字组成，且所有标签必须闭合，包括 br（
）,hr(<hr />)等；属性值必须用双引号包括；目的：更加符合 web 标准，也有利于 seo
4. 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。目的：减少代码量
5. 语义话 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，内联元素不可嵌套块级元素。目的：减少代码量，有利于 seo
6. 尽可能减少 div 嵌套， 充分利用 html 自身属性及样式继承原理减少代码量。如
   欢迎来到驴妈妈 您的用户名是 用户名
   完全可以用一下代码替代：
   欢迎来到驴妈妈，您的用户名是用户名
   目的：减少代码量，有利于 seo
7. 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;
8. 书写链接地址时，必须避免重定向，例如：href=“http://www.lvmama.com/“，即须在 URL 地址后面加上“/”；
9. 在页面中尽量避免使用 style 属性，除非考虑网站 http 请求，网站响应速度等因素币种情况具体分析
10. 严格区分作为内容的图片和作为背景的图片。作为背景的图片才用 css sprite 技术。Css sprite 技术的优点是减少了 http 请求数，但使图片面向 css 的 background-position 增加了耦合度，也增加了维护成本。如果图片有修改，不要删除已添加的图片，在空白处新增修改后的图片，减少修改风险。

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 , 如 thead,tr,th,td,tbody,tfoot,colgroup,scope; (cellspaing 及 cellpadding 的 css 控制方法:table{border:0;margin:0;border-collapse:collapse;} table th, table td{padding:0;} , base.css 文件中我会初始化表格样式)
12. 用 png 图片做图片时, 要求图片格式为 png-8 格式,若 png-8 实在影响图片质量或其中有半透明效果, 请为 ie6 单独定义背景: _background:none;_filter:progid:DXImageTransform.Microsoft.AlphaImageLoader (sizingMethod=crop, src='img/bg.png');
13. 减少使用影响性能的属性, 比如 position:absolute || float ;
14. 须为大区块样式添加注释, 小区块适量注释;
15. 代码缩进与格式: 建议单行书写,统一使用 tab 进行缩进。
16. 图片必须加上 alt 属性；给重要的元素和截断的元素加上 titile。目的：有利于 seo，搜索引擎的爬虫。
17. 给区块代码及重要功能（比如循环）加上注释，方便后台程序员嵌套模板
18. 特殊符号的使用：尽可能使用代码替代：比如<>空格>><<等等;
19. 书写页面过程中，请考虑向后扩展性

css 书写规范

1. 代码统一为 utf-8
2. 都使用 class，特殊情况除外
3. 样式命名推荐使用英文避免使用汉语拼音，尽量使用简易的单词组合，命名方式参照《WEB 前端开发 CSS 命名参考》。命名方式采用驼峰命名法和下划线命名法两种，提高可读性。例如：topNav、topNavMenu、topNavMenu_noBorder。驼峰命名法用来区别不同的单词，下划线命名法表明从属关系。如：“.timeList”和“.time_list”分别表示时间列表和时间部分下的列表。
4. CSS 命名：

➢ 常用的 css 命名规则：

页面各元素 class 命名		
容器：container	.bold{font-weight:bold;}	当前：current
页头：header	.cl{clear:left;}	内容器： inbox 或 innerbox
登录条：loginBar	.cr{clear:right;}	标题：title
标志：logo	.cb{clear:both;}	内容：content

侧栏：sideBar	.fl{float:left;}	输入框：textbox
广告：banner	.fr{float:right;}	按钮：btn
导航：nav	.fn{float:none;}	当鼠标 over：xxx_over
子导航：subnav	.fontn{font-weight:normal;}	当鼠标 out：xxx_out
菜单：menu	.txtl{text-align:left;}	
子菜单：submenu	.txtr{text-align:right;}	
搜索：search	.txtc{text-align:center;}	
滚动：scroll	.inbox{padding:10px;}	
页面主体：main	.pointer{cursor:pointer;}	
内容：content		
列：column		
左边列：col_left		
右边列：col_right		
标签页：tab		
文章列表：list		
提示信息：msg		
小技巧：tips		
栏目标题：title		
加入：joinus		
指南：guild		
服务：service		
热点：hot		
新闻：news		
下载：download		
注册：register		
状态：status		
按钮：btn		

投票：vote 合作伙伴：partner 友情链接：friendlink 页脚：footer 版权：copyright		

- 直观命名

  当在设计 web 页面以及需要对一个 div 进行标识的时候，最自然的想法就是使用可以描述元素所在页面位置的词汇来对其命名。这种方法使得类以及 id 的名称如下面所示：

  自上而下小组：top-panel

  横向：horizontal-nav

  左面：left-side

  中心-栏目：center-column

  右面：right-col

  这些是 css 以及 xhtml 类和 id 的有效命名方式。这些词汇简单并且能够使人顾名思义，因此满足了标识页面元素以及相应的 css 样式的需要。

  但问题是这样的名称同页面内容的特定表达方式相关联。这些命名参考了某种特定页面布局中的页面元素位置，因此在这样的布局之外使用就会显得不合适甚至造成理解混乱。同时，这些命名没有涉及文档内容的结构。因此，下面给出了对 css 类以及 id 命名更好的方法。

- 结构化命名

  结构化的标记意味着表达方式/位置信息同内容的完全分离——这其中包括出现在标记（markup）中的类和 id 名称。

  有标记的相关信息都是用来描述文档的结构而不是外观。这样的特点使得我们可以通过简单的改变 css 的方式来对不同外观格式下的内容（content）以及标记（markup）进行重用。当你理解这种方式时，很容易就可以发现采用页面位置来为类以及 id 命名的方式在处理如音频（audio）等外观格式上显得非常不合适。因此，应当根据在文档中的使用目的而非出现位置来对类以及 id 进行结构化命名。

  可以按照如下所示的结构化方式来对类以及 id 名称命名：

  顶部抢眼部分：branding

  导航部分：main-nav

  主要内容部分：main-content

这些名字同直观命名方式一样非常易懂，但他们描述了页面元素的作用而非位置。这使得代码更加符合使用纯粹的结构化标记（structural markup）的初衷，即开发人员可以在不改变标记的情况下对各种各样媒体下的显示格式进行处理。

即使你不打算在其他的媒体上对 web 页面进行格式修改，使用结构化命名方式还可以帮助你在日后的站点升级或重新设计中更为轻松。例如，结构化命名避免了当一个 div 同 id right-column 移动到页面左边后所带来的混乱。对 div sidebar 的采用这样的命名方式就显得更加适当，因为无论它出现在页面的哪一边，这个名字仍然对本项目人员来说直观易懂。

- 自定义命名：

  根据 w3c 网站上给出的,最好是用意义命名

  比如：是重要的新闻高亮显示（像红色）

  有两种

  .red{color:red}

  .important-news{color:red}

  很显然第二种传达的意义更加明确,所以尽量不要用意义不明确的作为自己自定义的名字

5. css 属性书写顺序，建议遵循定位布局属性–>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。

   定位属性（如：display，position，float，clear，visibility，table-layout 等）

   自身属性（如：width，height，margin，padding，border 等）

   文本属性（如：font，line-height，text-align，text-indent，vertical-align 等）

   其他属性（如：color，background，opacity，cursor，content，list-style，quotes 等）
6. 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。

   如 font-family：“宋体”，应替换成 font-family：“\5b8b\4f53“

以上 CSS 命名规范其尽可能的按 3c（css 2.1　XHTML1.0 Transitional）标准来执行操作，在一定的程度上加强了团队的协作水平，提升了项目的开发效率和质量，同时也助于大家良好书写规范代码的习惯。

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; 原则上所有功能均根据 XXX 项目需求原生开发, 以避免网上 down 下来的代码造成的代码污染(沉冗代码 || 与现有代码冲突 || ...)等。
2. 库引入: 原则上仅引入 jQuery 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 变量要求是纯英文字母, 首字母须小写, 如 iZhangWei;
   jQuery 变量要求首字符为'_', 其他与原生 JavaScript 规则相同, 如: _iZhangWei;
   另, 要求变量集中声明, 避免全局变量.
4. 类命名: 首字母大写, 驼峰式命名. 如 IZhangWei;
5. 函数命名: 首字母小写驼峰式命名. 如 iZhangWei();
7. 命名语义化, 尽可能利用英文单词或其缩写;
8. 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;
9. 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;
10. 代码结构明了, 加适量注释. 提高函数重用率;
11. 注重与 html 分离, 减小 reflow, 注重性能.

图片规范

1. 所有页面元素类图片均放入 pic 应用下，根据项目名称进行文件夹命名;
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 || _ 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; 另, 命名分头尾两部分, 用下划线隔开, 比如 ad_left01.gif || 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(若使用, 请参考 css 规范相关说明);
6. 运用 css sprite 技术集中小的背景图或图标, 减小页面 http 请求, 但注意, 请务必在对应的 sprite psd 源图中划参考线.

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 */;

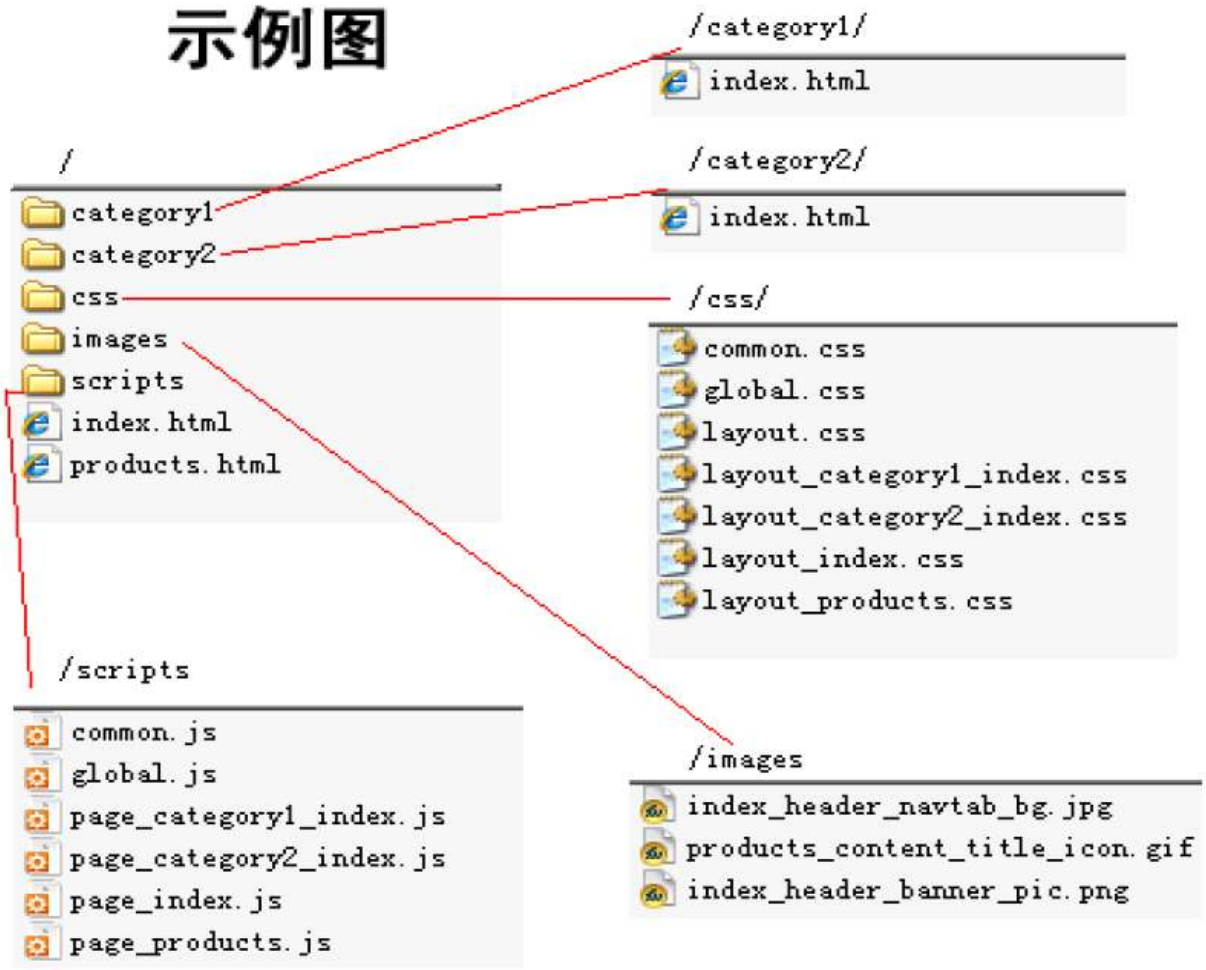
开发及测试工具约定

测试工具: 前期开发仅测试 FireFox & IE6 & IE7 & IE8 , 后期优化时加入 Opera & Chrome & Safari;
建议测试顺序: FireFox-->IE7-->IE8-->IE6-->Opera-->Chrome-->Safari, 建议安装 firebug 及 IE Tab Plus 插件.

目录结构

我们单独一节来说明网站目录结构的目的是让我们重视。因为无论你的 XHTML、CSS、Javascript 写得多熟练多好，而网站的目录结构和其命名是让人和搜索引擎读不懂的，那么网站就没有真正做到标准化而且整个网站的后期扩展和维护的成本和代价会很大。
还有一点需重视就是 xhmtl css js 的代码注意缩进，并保持格式整齐的并且提供注释，保证可阅性。同时为后期编写程序提供良好的开发条件。
首先网站的目录结构和文件命名清晰、XHTML 里面的元素命名清晰会给 SEO 带来好处。例如文件的命名，如果使用全拼，那么 Google 是自动识别拼音进行排名的。
关于 SEO 相关的知识，就不在这里一一阐述了。接下来我们详细介绍目录结构和命名规范。

以下是一个大概的目录结构示例图



目录结构主要分为四类，需要注意的是，所有命名必须为小写英文，不能大写或中文。

- categorys(目录)

  目录的命名尽可能使用英文或者全拼表达目录内的页面作用（语义化），需要注意的是不要使用中文词组简拼（eg: 目录--> ml）。简拼容易出现重复、或者目录结构复杂的时候容易出现混乱，给后期维护带来很大的麻烦。

- css

  css 的目录命名可以为 style、css、skin 等，如果网站的目录结构不是很复杂的，尽

量把 css 统一放在跟目录。这样可以方便后期的维护操作。如果网站的目录结构很复杂，
层次超过 3 层以上的，可以在对应的层设置目录页面结构（layout）的 css。

- js
  js 的目录命名一般用 scripts 或者其个容易让人知道里面是放 js 脚本的名称。同样 js
  的目录结构也是和 css 一样。
- images
  images 根据网站规模来调整放图片的目录。，一般根目录设置的 images 是存放整站
  共用的图片（包括图片图标背景等），而各二级三级目录里面也可以设置相应的 images 目录存放当前级的图片。

其他规范

1. 开发过程中严格按分工完成页面, 以提高 css 复用率, 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.

CSS Hack

即便是完美的 CSS 也未必能在目前众多的终端浏览器中呈现一致的效果，所以，CSS Hack 在很多情况下都是必要的，建议先以对 CSS 标准支持得比较好的浏览器（如 FF 或 Chrome）为主编辑 CSS，最后再处理 IE 的兼容性——单独为 IE 建立一个 CSS 文件（比如 for-ie.css，fuck-ie.css,ie-hack.css 等），最后在 HTML 文件中，通过 IE 的条件注释按需引用。