**研发部前端代码规范**

**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**

1. html, css, js, img文件均归档至“系统开发规范”约定的目录中

2. html文件命名: 英文命名, 后缀.html

3. css文件命名: 英文命名, 后缀.css共用common.css, 首页index.css, 其他页面依实际模块需求命名

4. js文件命名: 英文命名, 后缀.js共用common.js, 其他依实际模块需求命名

**HTML书写规范**

1. 统一为html5声明类型<!DOCTYPE html>
2. 编码统一为<meta charset="UTF-8">
3. 向浏览器表示该页面是简体中文语言<html lang="zh-cmn-Hans">
4. 书写时利用IDE实现层次分明的缩进（注：我们一直默认用4个空格替换一个Tab，请调整自己的IDE）
5. 标签语义化（所谓标签语义化，就是指使用正确的标签表达合适的网页内容）
6. 非特殊情况下样式文件外链至<head>...</head>之间; JavaScript文件外链至页面底部
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-1.11.1.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. 内联（行内）元素中不可嵌套块级元素
9. 尽可能减少div嵌套，节省资源，尽可能用最少的代价做最多的事情，避免无用标签嵌套
10. 书写链接地址时, 必须避免重定向，例如：href=http://www.webjx.com/css/"http://itaolun.com/", 即须在URL地址后面加上“/”
11. 非特殊情况不要写行内样式即：style=””
12. 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>

尽量写成:

<p><label for="name">姓名: </label><input type="text" id="name" /></p>

1. 能用css写出来的图，尽量不要用图片
2. 给区块代码及重要功能加上注释, 方便后台添加功能
3. 特殊符号使用escape 码代替
4. 书写页面过程中, 请考虑向后扩展性

**CSS书写规范**

1. 编码统一为utf-8
2. 除了初始化css外，禁止直接对标签定义样式;
3. 多个selector共用一个样式集，则多个selector必须写成多行形式
4. 属性名冒号之前不加空格，冒号之后加空格;
5. 禁止采用单行书写风格;
6. 如果有必要加入浏览器前缀，则按照-webkit- / -moz- / -ms- / -o- / std的顺序进行添加;
7. 整个网站共用的common.css放在base.html的head里面整个网站共用样式，列如：网站的header，footer，还有一些初始化的一些约定字体、字号、颜色等等
8. 类名，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，单词之间使用中划线“-”
9. 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
10. 书写代码前, 考虑并提高样式重复使用率
11. 颜色值尽量用十六进制值并且应该全部小写，尽量使用简写形式的十六进制值，例如，用 #fff 代替#ffffff
12. 与 <link> 标签相比，@import 指令要慢很多，不光增加了额外的请求次数，还会导致不可预料的问题，做好的办法是尽量合并样式表，能复用的复用
13. float、position等破坏布局导致重新渲染的属性，能不用就不用
14. 必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: 多行书写，看起来舒服，查找方便，提高代码可读性

**JavaScript书写规范**

1. 文件编码统一为utf-8, 书写过程中, 每行代码结束必须有分号

2. 库引入: 引入第三方库时需与团队协商。

3. 变量命名: 驼峰式命名. 英文字母或英文字母加数字, 首字母须小写, 如haoXiaoJun；jQuery等第三方变量前面不要加任何东西。

另, 尽量集中声明变量, 避免全局变量

4. 类命名: JavaScript首字母大写, 驼峰式命名. 如 ITaoLun

5. 函数命名: JavaScript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非注释类中文字符应该转换成unicode编码使用, 以避免编码错误时乱码显示

9. 注重与html分离

10. 代码结构明了, 加适量注释. 提高函数重用率，建议在写完一个方法在函数头部注释（参数、功能、return值），配以适当的行后注释（不可多）

11.注释最好用/\*\*/，不要用//因为当你的js比较大需要压缩的时候，那么压缩后会变成一行，//后面的会被注释掉

**图片规范**

1. 所有页面元素类图片均放入img文件夹

2. 图片格式多用于.jpg，对背景有透明需求或者其他清晰度的要求用png

3. 命名全部用小写英文字母、数字、中划线-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

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. 运用css sprite技术集中小的背景图或图标, 减小页面http请求

**注释规范**

1. html注释: 注释格式 <!–这儿是注释–>
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**

建议使用Sublime||WebStorm|| Dreameaver , 亦可根据自己喜好选择。

**其他规范**

1. 开发过程中严按分工完成页面, 以提高css复用率, 避免重复开发;

2. 减小冗余代码, 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。

**附：命名参考**

主体：main

外层：wrap

 功能条：func-bar

友情链接：friend-link

 版权：copyright

页眉：header

 页脚：footer

标题：title

主导航：main-nav

子导航：sub-nav

菜单：menu

子菜单：sub-menu

 注释：note

面包屑：breadcrumb

 容器：container

内容：content

 搜索：search

登陆：login

当前状态：current

投票：vote

页头：header

标志：logo

侧栏：sidebar

广告：banner

滚动：scroll

标签页：tab  
 文章列表：list

信息：msg

 提示技巧：tips

 指南：guide

服务：service

热点：hot

新闻：news

下载：download

注册：reg（register）

状态：status

按钮：btn

 外层：wraper

内层：inner