|  |  |
| --- | --- |
|  | **2013** |
|  | sigma  Jopix |

|  |
| --- |
| **[ RPG虚拟学习平台UI设计说明文档]** |
| UI方向 |

目 录

[1 适用环境和对象 1](#_Toc346542704)

[2 必要性 2](#_Toc346542705)

[3 技术原则 2](#_Toc346542706)

[3.1 相关技术： 2](#_Toc346542707)

[3.2 调试环境： 2](#_Toc346542708)

[3.3 代码规范化书写 2](#_Toc346542709)

[4 代码 编写规范 2](#_Toc346542710)

[4.1 目录结构及命名 规则 2](#_Toc346542711)

[4.1.1 目录结构规范 2](#_Toc346542712)

[4.1.2 文件和目录命名规范 3](#_Toc346542713)

[4.1.3 图片的命名规范 3](#_Toc346542714)

[4.1.4 其它文件命名规范 3](#_Toc346542715)

[4.2　html编写规则 4](#_Toc346542716)

[4.2.1 一般原则 4](#_Toc346542717)

[4.2.2 代码规则 5](#_Toc346542718)

[4.2.2.1 head区标识 5](#_Toc346542719)

[4.2.2.2 head区以下的标记 6](#_Toc346542720)

[4.3 css编写规则 9](#_Toc346542721)

[4.3.1 css文件 调用写法 9](#_Toc346542722)

[4.3.2 类和ID命名规则 11](#_Toc346542723)

[5　页面模版使用规范 12](#_Toc346542724)

[6 第一次迭代UI设计说明 13](#_Toc346542725)

[6.1 大体设计要求： 13](#_Toc346542726)

[6.2 页面设计描述。 14](#_Toc346542727)

[6.2.1 主页面： 14](#_Toc346542728)

[6.2.2 PK界面 15](#_Toc346542729)

[6.2.3 比赛界面 15](#_Toc346542730)

[6.2.4 挑战列表页面 16](#_Toc346542731)

[6.2.5 挑战界面 16](#_Toc346542732)

[6.2.6 用户主页 17](#_Toc346542733)

## 1 适用环境和对象

本规范适用基于浏览器的B/S版软件项目开发工作。开发流程中的模版页面编写和模版文件套用工作必须遵照此规范执行。适用对象为开发编码人员、UI设计人员、模版编写人员、界面测试人员等。

基于客户端的C/S版软件开发工作不适用本技术规范。

## 2 必要性

本规范旨在制订开发编码人员和UI模版编写人员之间在工作交叉部分的技术标准，使他们遵循同 一操作规范，利于交叉工作的平缓顺利交接。以标准化方式，提高沟通和技术协作的水平，提高工作效率。减少和改变责任不明，任务不清，和由此产生的信息沟通 不畅、反复修改、重复劳动、效率低下的现象。

## 3 技术原则

### 3.1 相关技术：

界面设计：html5 + JS + Ajax + CSS 3.0

### 3.2 调试环境：

统一使用Firefox进行调试测试

### 3.3 代码规范化书写

UI设计师设计出页面模版，程序员负责内容数据的嵌入，数据可能是从数据库中提取出来，也可能是静态写入的提示性文字，最后形成一个新的页面展示给软件操作者。

所有page文件的结构利用HTML＋XHTML标签来定义，而所有涉及表现的东西通通剥离出来，把它放到一个单独CSS文件以便后期修改。

## 4 代码 编写规范

### 4.1 目录结构及命名 规则

#### 4.1.1 目录结构规范

1、目录建立的原则：以最少的层次提供最清晰简便的访问结构。

2、根目录一般只存放index.htm以及其他必须的系统文件

3、在根目录中 原则上应该按照系统的栏目结构，给每一个栏目开设一个目录，根据需要在每一个栏目的目录中开设一个images 和media 的子目录用以放置此栏目专有的图片和多媒体文件，如果这个栏目的内容特别多，又分出很多下级栏目，可以相应的再开设其他目录。根目录下的images用于 存放各页面都要使用的公用图片，子目录下的images目录存放本栏目页面使用的私有图片

4、所有JS,ASP,PHP等脚本存放在根目录下的 scripts目录

5、所有CGI程序存放在根目录下的cgi-bin目录

6、所有CSS文件存放在根目录下style目录

7、 每个语言版本存放于独立的目录。例如：简体中文gb

8、所有flash, avi, ram, quicktime 等多媒体文件存放在根目录下的media目录

9、temp 子目录放客户提供的各种文字图片等等原始资料，以时间为名称开设目录，将客户陆续提供的资料归类整理。

#### 4.1.2 文件和目录命名规范

1、文件命名的原则：以最少的字母达到最容 易理解的意义。

2、每一个目录中包含的缺省html 文件，文件名统一用index.htm

3、文件名称统一用小写的英文字母、数字和下 划线的组合，不得包含汉字、空格和特殊字符

4、尽量按单词的英语翻译为名称。例如：feedback(信息反馈)，aboutus(关于我们) 不到万不得已不要以拼音作为目录名称

5、多个同类型文件使用英文字母加数字命名。例如：news01.htm。注意， 数字位数与文件个数成正比，不够的用0补齐。例如共有200条新闻，其中第18条命名为news018.htm

#### 4.1.3 图片的命名规范

1、名称分为头尾两两部分，接下来的单词首字母大写。

2、头部分表示此图片的大类性质。例如： 放置在页面顶部的广告、装饰图案等长方形的图片我们取名：banner ；标志性的图片我们取名为：logo ；在页面上位置不固定并且带有链接的小图片我们取名为button ；在页面上某一个位置连续出现，性质相同的链接栏目的图片我们取名：menu ；装饰用的照片我们取名：pic ；不带链接表示标题的图片我们取名：title 依照此原则类推。尾部分用来表示图片的具体含义，用英文字母表示。

3、有onmouse效果的图片，两张分别在原有文件名后加“On”和“Off”命名

#### 4.1.4 其它文件命名规范

1、 js的命名原则以功能的英语单词为名。例如：广告条的js文件名为:ad.js

2、所有的CGI文件后缀为cgi。所有CGI程序的配置文件为 config.cgi

### 4.2　html编写规则

#### 4.2.1 一般原则

1、在编写模版文件，排布结构表格之前，要思考一个最佳方案，表格的嵌套尽量控制在三层以内；要考 虑程序套用的可实现性、通用性、灵活性、预见性，所有内容均采用积木式组织，可替换和删除，并对总体结构不会造成破坏性影响

2、尽量避免 <colspan> <rowspan> 两个标记，经验表明，这两个标记会带来许多麻烦

3、一个网页要尽量避免用整个 一张大表格，所有的内容都嵌套在这个大表格之内。因为浏览器在解释页面元素时，是以表格为单位逐一显示，如果一张网页是嵌套在一个大表格之内，那么很可能 造成的后果就是，当浏览者敲入网址，他要先面对一片空白很长时间，然后所有的网页内容同时出现。如果必须这样做，请使用 <tbody>标记，以便能够使这个大表格分块显示

4、排版中经常会遇到需要进行首行缩进的处理，不要使用全角空格来达到效果，规范 的做法是在样式表中定义 p { text-indent: 2em; } 然后给每一段加上 <p> 标记，注意，一般情况下，请不要省略 </p> 结束标记

5、原则上，我们禁止用 <img width=? height=?> 来人为干预图片显示的尺寸，而且建议 <img> 标签中不要带上width 和height 两个属性，这是因为制作过程中，图片往往需要反复的修改，这样可以避免人为干预图片显示的尺寸，尽可能的发挥浏览器自身的功能；但是这样的一个副作用是当 网页还未加载图片时，不会留出图片的站位大小，可能会造成网页在加载过程中抖动（如果图片是插在一个固定大小的表格里的，不会有这个现象），尤其是当图片 的尺寸较大时，这种现象会很明显，所以当预料到这种会明显导致网页抖动的情况会发生时，务必在最后给 <img>附上 width 和 height 属性

6、为了最大程度的发挥浏览器自动排版的功能，在一段完整的文字中尽量不要使用<br> 来人工干预分段

7、 不同语种的文字之间应该有一个半角空格，但避头的符号之前和避尾的符号之后除外，汉字之间的标要用全角标点，英文字母和数字周围的括号应该使用半角括号

8、 为贯彻结构层和表现层分离的原则，严禁使用传统的HTML3.2/4.0控制表现的标签，例如<font>,<b>，还有本意用 于结构后来被滥用于控制表现的标签，例如：<h1>,<table>等。所有的字号都应该用样式表来实现，禁止在页面中出现 <font size=　> 标记，<b>　</b>　<h1>　</h1>标记，严禁在结构页面写表现标签

9、不要在网页 中连续出现多于一个的也尽量少使用全角空格（英文字符集下，全角空格会变成乱码），空白应该尽量使用 text-indent, padding, margin, hspace, vspace 以及透明的gif 图片来实现。

10、中英文混排时，我们尽可能的将英文和数字定义为 verdana 和arial 两种字体

11、行距建议用百分比来定义，常用的两个行距的值是line-height:120%/150%

12、 系统中的路径全部采用相对路径

13、为保证系统和浏览器的兼容性，当设置背景图片时，要坚持用双引号。

14、“网页大小”为网页的所有文 件大小的总和，包括HTML文件和所有的嵌入的对象。页面大小保持在34K以下为合适

#### 4.2.2 代码规则

##### 4.2.2.1 head区标识

（head区是指首页HTML代码的<head> 和</head>之间的内容）

1、公司版权注释

<!–

Generator: sigma

Creation Data: 2005-8-1

Original Author: 张三

–>

2、网页显示字符集

简体中文：

<META. HTTP-EQUIV=”Content-Type” CONTENT=”text/html; charset=gb2312″>

繁体中 文：

<META. HTTP-EQUIV=”Content-Type” CONTENT=”text/html; charset=BIG5″>

英 语：

<META. HTTP-EQUIV=”Content-Type” CONTENT=”text/html; charset=iso-8859-1″>

?

3、简介

<META. NAME=”DESCRIPTION” CONTENT=”这里填您网站的简介”>?

4、网页的css文件定义，所有css文件尽量采用外部调用

<LINK href=”style/style.css” rel=”stylesheet” type=”text/css”>

5、网页标题

<title> 这里是你的网页标题</title>

6、所有的javascript脚本尽量采取外部调用

<SCRIPT. LANGUAGE=”javascript” SRC=”script/xxxxx.js”></SCRIPT>

head区可以选择加 入的标识

7、设定网页的到期时间。一旦网页过期，必须到fwq上重新调阅。

<META. HTTP-EQUIV=”expires” CONTENT=”Wed, 26 Feb 1997 08:21:57 GMT”>

8、禁止浏览器从本地 机的缓存中调阅页面内容。

<META. HTTP-EQUIV=”Pragma” CONTENT=”no-cache”>

9、用来防止别人在框 架里调用你的页面。

<META. HTTP-EQUIV=”Window-target” CONTENT=”\_top”>

10、自动跳转。

<META. HTTP-EQUIV=”Refresh” CONTENT=”5;URL=http://www.yahoo.com”>

5指时间停留 5秒。

11、网页搜索机器人 向导.用来告诉搜索机器人哪些页面需要索引，哪些页面不需要索引。

<META. NAME=”robots” CONTENT=”none”>

CONTENT的参数有all,none,index,noindex,follow,nofollow。 默认是all。

12、收藏 夹图标

<link rel = “Shortcut Icon” href=”favicon.ico”>

13、搜索关键字

<META. NAME=”keywords” CONTENT=”关键字1,关键字2,关键字3,…”>

##### 4.2.2.2 head区以下的标记

1、body标记

为了保证浏 览器的兼容性，必须设置页面背景

<body bgcolor=”#FFFFFF”>

2、table标记

在写 <table> 互相嵌套时，严格按照的规范，对于单独的一个<table>来说，<tr>，<td>各缩进两个半角空格，结束标记 和开始标记平齐。<td> 中如果还有嵌套的表格，<table>也缩进两个半角空格，如果<td>中没有任何嵌套的表格，</td> 结束标记应该与 <td> 处于同一行，不要换行。

正确写法：

a\

<table width=”100%”? border=”0″ cellspacing=”0″ cellpadding=”0″>

<tr>

<td>&nbsp;</td>

</tr>

<tr>

<td>

<table width=”100%”? border=”0″ cellspacing=”0″ cellpadding=”0″>

<tr>

<td>&nbsp;</td>

</tr>

</table>

</td>

</tr>

</table>

b\

<td><img src=”../images/sample.gif”></td>

错误写法

a\

<table width=”100%”? border=”0″ cellspacing=”0″ cellpadding=”0″>

<tr>

<td>&nbsp;</td>

</tr>

<tr>

<td><table width=”100%”? border=”0″ cellspacing=”0″ cellpadding=”0″>

<tr>

<td>&nbsp;</td>

</tr>

</table>

</td>

</tr>

</table>

b\

<td><img src=”../images/sample.gif”>

</td>

这是因为浏览器认为换行相当于一个半角空格，以 上不规范的写法相当于无意中增加一个半角空格，如果确实有必要增加一个半角空格，也应该这样写：

<td><img src=”../images/sample.gif”> </td>

于同一个级别的 <table> 一定是左首对齐的，另外不允许没有任何内容的空的单元格存在，高度大于等于12px 的单元格应该 在 <td> 和 </td> 之间写一个&nbsp;，如果高度小于12px, 则应该 在 <td> 和 </td> 之间插入一个1\*1 大小的透明的gif 图片，这是因为某些浏览器认为空单元格非法而不会予以解释。如果代码顺序较乱，在DW中可以通过command->apply souce formatting进行重新整理！

3、Width 和height标记

一般情况下只有一列的表格，width 写在<table> 的标签内，只有一行的表格，height 写在 <table> 的标签内，多行多列的表格，width 和height 写在第一行或者第一列的 <td> 标签内。总之遵循一条原则：不出现多于一个的控制同一个单元格大小的height 和width, 保证任何一个width 和height 都是有效的，也就是改动代码中任何一个width 和height 的数值，都应该在浏览器中看到变化。

a、只一列的表格的width标记

<table width=”100%”? border=”0″ cellspacing=”0″ cellpadding=”0″>

<tr>

<td>&nbsp;</td>

</tr>

<tr>

<td>&nbsp;</td>

</tr>

</table>

b、 只一行的表格的height标记

<table width=”100%” height=”30″? border=”0″ cellpadding=”0″ cellspacing=”0″>

<tr>

<td>&nbsp;</td>

<td>&nbsp;</td>

<td>&nbsp;</td>

</tr>

</table>

c、 多行多列表格的width和height标记

<table border=”0″ cellpadding=”0″ cellspacing=”0″>

<tr>

<td width=”100″ height=”30″>&nbsp;</td>

<td width=”200″>&nbsp;</td>

<td width=”300″>&nbsp;</td>

</tr>

<tr>

<td>&nbsp;</td>

<td>&nbsp;</td>

<td>&nbsp;</td>

</tr>

</table>

4、table的 width属性

为遵循页面 结构灵活性、通用性原则，table的width属性原则上全部写成100％或者不写width属性，不推荐写成其他非100％宽度属性。留空显示效果通 过其给外部td施加style的padding属性实现。

<table width=”100%”? border=”1″ cellspacing=”0″ cellpadding=”0″>

<tr>

<td width=”200″ bgcolor=”#FF0000″ style=”padding:20px; “>

<table width=”100%” border=”1″ cellpadding=”0″ cellspacing=”0″ bgcolor=”#FFFFFF”>

<tr>

<td>table宽度 100％</td>

</tr>

<tr>

<td>&nbsp;</td>

</tr>

</table>

</td>

<td>&nbsp;</td>

</tr>

</table>

需要说明的是这里td的style虽然是css的写法，但不属于表现层的内容，而是属于结构层 的内容，所以可以直接写在html标记中间。

5、数据列表表格代码写法

表格外加上一个<div>，并对<div>赋一个 id，即<div id =list>，然后通过样式表对这个id下的所有html元素进行控制，

用<th>表示表头

设置每列宽度的位置在表头对应的<th>中，其中列表项中，字数最多或者字数不定的一项不设置宽度。

宽度的值依据列表项内容的多少来 定，2个字的30px，三个字40px，时间、年月日（比如2004-11-11）80-120px,类似于标题的列表项，表格对齐方式为左对齐 （align=left）,时间，人名一般居中对齐，数据一般右对齐（align=right）。

对齐页面内容时不要用div来对齐，直接在td 或th中写align=…

表头文字一般不折行，方法是在<th>中加上nowrap,或者通过样式表来控制

<table width=”100%” border=”0″ cellspacing=”0″ cellpadding=”0″>

<tr>

<th width=”30″ nowrap>选择</th>

<th align=”left” nowrap>标题</th>

<th width=”80″ nowrap>发表人</th>

<th width=”120″ nowrap>时间</th>

<th width=”40″ nowrap>大小</th>

</tr>

<tr>

<td>&nbsp;</td>

<td align=”left”>&nbsp;</td>

<td align=”center”>&nbsp;</td>

<td align=”center”>&nbsp;</td>

<td align=”right”>&nbsp;</td>

</tr>

<tr>

<td>&nbsp;</td>

<td align=”left”>&nbsp;</td>

<td align=”center”>&nbsp;</td>

<td align=”center”>&nbsp;</td>

<td align=”right”>&nbsp;</td>

</tr>

</table>

### 4.3 css编写规则

#### 4.3.1 css文件 调用写法

1、所有的CSS的尽量采用外部调用，特殊情况才允许使用内置html写法。

<LINK href=”style/style.css” rel=”stylesheet” type=”text/css”>

css文件结构组织

1、 文件提头，css文件名称、时间、作者

2、　将不同目的和效果的CSS选择符分别编组，编组分层级使结构清晰，便于查对。并恰当地成块注释，注释要说 明施加的页面文件的对象。顶级注释用\*TOP\*打头，次级注释不用。

.headtext {

font-size: 14px;

color: #ffffff;

font-weight: bold;

}?

.headtext a:link {

color: #ffffff;

text-decoration: none;

}

.headtext a:visited {

color: #ffffff;

text-decoration: none;

}

.headtext a:hover {

color: FED078;

text-decoration:none ;

}

#clientcard {

align:left;

}

#clientcard th {

height:20;

font-size: 12px;

color: #737373;

padding-left:10;

font-weight: bold;

background-attachment: fixed;

background:url(images/dot2.gif)

background-repeat: no-repeat;

background-position: left center;

}

#clientcard td {

height:20;

font-size: 12px;

color: #FD783A;

padding-right:25;

}

3、选择符分组的顺序是重定义的最先，伪类其次，自定义最后。便于自己和他人阅读。

例 如：

BODY {

margin-left: 0px;

…

}

TD {

font-size: 12px;

…

}

a:link {

color:?#484848;

text-decoration: none;

}

a:visited {

color: #484848;

text-decoration: none;

}

a:hover {

color:#BD0800;

text-decoration: underline;

}

.ltreename {

font-size: 14px;

…

}

#### 4.3.2 类和ID命名规则

1、 以功能和定义对象的位置，而不是外观为类和ID命名，命名的英文单词也是小写开头，中间单词首字母大写，不采用下划线。例如创建了一个 .smallBlue 类，后来打算将文字改大，颜色变为红色，这个类名就不再有任何意义了。所以，用更有描述性的名字如 .copyRight 和.infoList更加合适。

单位

1、0不用单位

2、非零值要指明单位，指定字体、边距或大小时，必须指明所用的单位

例 如： padding: 0 2px 0 1em;

3、特例：line-height不需要单位

字号大小

1、为了保证不同浏览器上字号保持一致，字号像素px 来定义，一般使用中文宋体12px 和14.7px 这是经过优化的字号，黑体字或者宋体字加粗时，一般选用14.7px 的字号比较合适

例如： font-size: 12px;

顺序

1、a:link a:visited a:hover a:actived 要按照规范顺序写

2、 边框(border)、边距(margin)和补白(padding)的简写次序为：顺时针方向从上开始，即 Top, Right, Bottom, Left。

例如： margin: 0 1px 3px 5px;表示上边距为零，右边距为1px，依此类推。

组合选择符

1、 保持CSS短小对减少下载时间非常重要。尽量为选择符分组、利用继承(inheritance)以及使用简写(shorthand)来减少冗余。

使用图片

1、 css中经常遇到使用图片的情况, 图片的路径一律采用相对路径。

例如：

.hurdlename {

font-size: 14px;

color: #0B43C2;

height:25px;

font-weight: bold;

background-image: url(images/rbar\_bg.gif);

background-repeat: repeat-y;

text-indent: 12px;

}

.but1{

background-image: url(../images/but1.gif);

font-size: 12px;

color: #000000;

border: 0;

width:85px;

height:26px;

cursor: hand;

}

2、使用图片替换技术时要考虑与系统和文件结构的亲和力。如果引用css的所有文件不在同一级相对路径，就会出 现css指定的图片无法显示的问题。在这种情况下不支持使用图片技术。建议采用filter技术

例如：

input.buttton {

filter:progid:DXImageTransform.Microsoft.Gradient(gradienttype=0, startcolorstr=#CFD1CF, endcolorstr=#EFEFEF);

border: #B5B6B5 1px solid;

font-size: 12px;

color: #000000;

cursor: hand;

height:24px;

background:#ffffff;

}

## 5　页面模版使用规范

页面模版使用规范的目 的是统一和约定UI设计、界面工程师与不同的程序编码人员的行为方式，光有了书写代码的规范还不能完全解决界面套用的统一性问题。因为不同的编码人员对同 一个模版的理解有可能不同，再加上个人工作习惯的不同，界面套用的结果往往很容易出现差异。哪怕是每人一点点差异，都可能使软件的质量收到很大的影响。

责任分工明确

1、 UI设计、界面工程师负责界面风格的设计和软件模版的编写，并监督界面套用的效果。对软件最终的界面负责。

2、 编码工程师负责软件业务逻辑的实现，软件模版的套用。对软件的数据和程序负责。

3、现实的情况是个别编码工程师在模版套用出现偏差后，在界面工程 师提出修改意见的时候拒绝修改，以开发进度、时间不够为由搪塞，甚至主观认为“这个（界面）不重要”致使界面工程师的作用得不到应有的发挥，影响软件产品 质量。

4、 解决办法是各负其责。界面问题提高重视程度，并纳入开发流程和进度管理之中。

界面模版交界过程

1、 模版文件制作完成后，在提交给编码工程师时，要简要说明模版的文件的使用说明和提醒。

2、 编码工程师和界面工程师密切合作，完全理解模版使用说明。

3、对界面结构层html的table嵌套关系理解清楚并明确和程序结合的用意。有和程 序不相配合的情况，进一步和界面工程师讨论，获得解决方案。不得在界面工程师不知情的情况下随意修改table结构、定位属性和嵌套关系。这样做的坏处是 虽然一个人解决了暂时的显示问题，但和其他编码工程师套用的结果出现不同。也不利于界面工程师控制总体界面。

4、 对表现层的界面元素和css文件选择符的对应关系。做到心中有数，理解一个模版文件，到套用其他文件时就能够举一反三。

5、编码工程师完全 理解了界面模版后，就能够顺畅的把数据层的内容放到结构层合适的位置，并指定表现层合适的选择符属性。完成界面套用工作。

模版说明的内容

1、 总体的界面结构

2、 页面板块的布局和定位table的写法

3、 table嵌套的方式的理由

4、 不同功能的界面显示单元说明和使用方法

5、 css文件中选择符的使用说明，一般的都能理解，个别特殊的要着重说明。

Css文件的版本控制

Css 文件应引入版本控制的机制，项目组中应指定专人负责css文件的上传和修改。fwq的css文件和界面工程师的css文件应该同步更新。不支持编码工程师 随意添加选择符修改css文件。谁都可以改就没有标准，没有版本控制就做很难做到界面统一。

## 6 第一次迭代UI设计说明

### 6.1 大体设计要求：

第一次迭仅完成一下UI设计，实现基本按钮操作以及之间相关连接。

设计以简约，蓝色为主

字体：默认

### 6.2 页面设计描述。

#### 6.2.1 主页面：

顶部菜单



用户控制栏

NPC对话窗口，任务领取

浮动聊天窗口

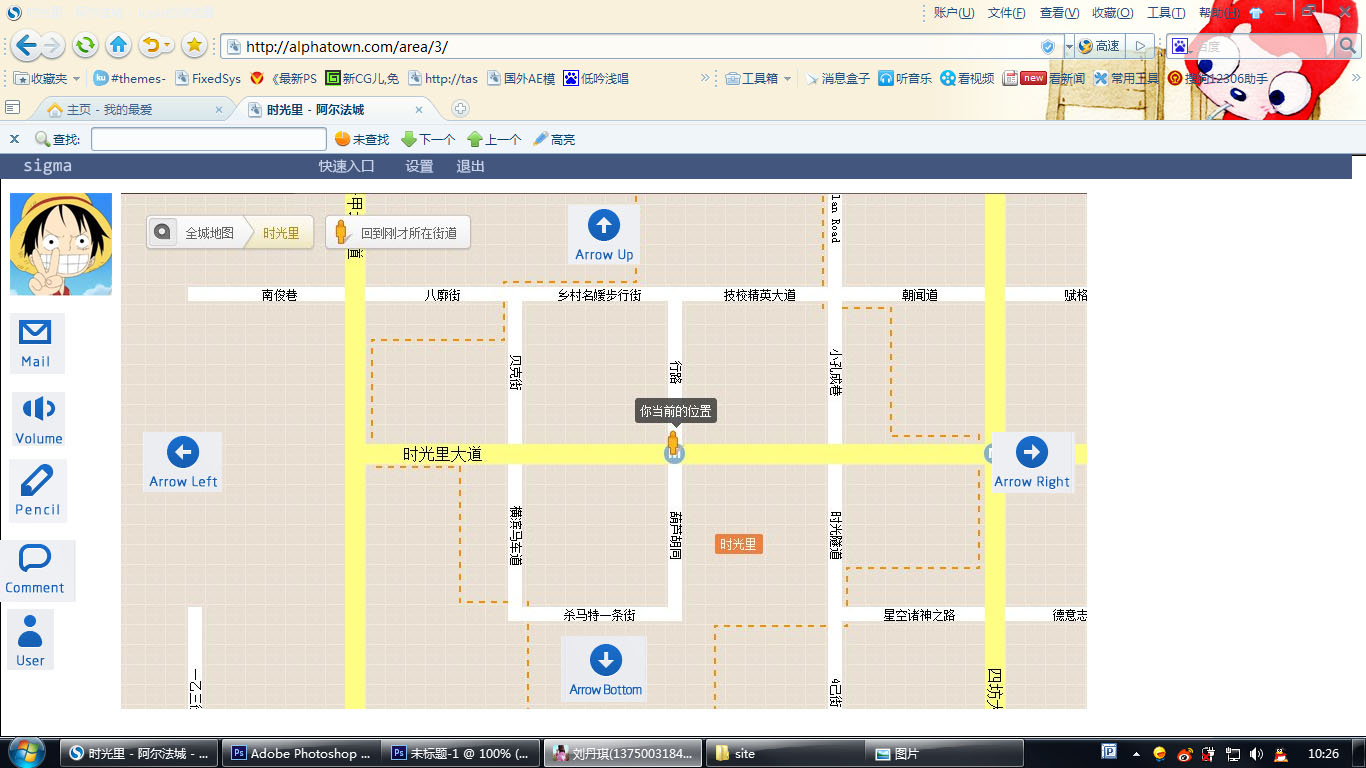
世界消息

包含：显示校园地图区域，用户控制栏，好友列表条，聊天窗口，NPC对话窗口

地图实现：可局部刷新显示地图信息，NPC信息以及用户动态信息

用户控制栏目包括用户头像，消息提示，快捷入口，退出等

好友列表显示在线好有头像



#### 6.2.2 PK界面

比赛4

比赛3

比赛2

比赛1

浮动聊天窗口

用户列表

用户控制栏

**Dfsfdsfdsfdsfsd**

包含比赛列表

用户列表里面显示参加PK的用户

每一个比赛都是通过方框颜色表示正在比赛或是等待用户加入

#### 6.2.3 比赛界面

消息框

题目详情

非模式对话框

头像

头像

头像

题目

列表

头像

头像

头像

用户控制栏

我方气球区

浮动聊天窗口

对方气球区

比赛界面大体这样：每队气球区，每做对一道题，显示该题对应颜色的气球。

对应用户发送的消息会显示对应的头像上

#### 6.2.4 挑战列表页面

浮动聊天窗口

挑战3

挑战4

挑战2

挑战1

用户列表

用户控制栏

**Dfsfdsfdsfdsfsd**

调整列表可以有两种形式，一种是列表式，一种树状式（拓扑顺序）

#### 6.2.5 挑战界面

浮动聊天窗口

气球区（挑战进度）

题目详情

非模式对话框

消息框

题目

列表

头像

头像

头像

用户控制栏

包含群挑，单挑

#### 6.2.6 用户主页

快捷输入框

基本信息

XX信息

信息列表区

功

能

区

头像

仅是用户自身的主页显示界面