



ZSYCUX 界面导航模块

说明文档

V2.1



目 录

目 录	- 1 -
1 基本界面概述.....	- 4 -
1.1 使用标准 HTML5 技术打造	- 4 -
1.2 定制登录页面.....	- 4 -
1.3 基本界面布局形式.....	- 5 -
1.4 主界面可根据屏幕宽度自适应.....	- 6 -
1.5 警报.....	- 7 -
1.6 界面自定义.....	- 8 -
2 安装模块及授权文件.....	- 9 -
2.1 安装模块.....	- 9 -
2.2 安装授权文件.....	- 9 -
3 框架模块的简单使用.....	- 10 -
3.1 打开 ZSYCNav-ux 模块的 palate	- 10 -
3.2 将 Page_Horizontal (横式布局) 或者 Page_Vertical 模块 (竖式布局) 拖入文件夹 中	- 11 -
3.3 设置 Page_Horizontal (横式布局) 或者 Page_Vertical 模块 (竖式布局), 其 displayName 即为网页标题。	- 12 -
3.4 Page_Horizontal (横式布局) 或者 Page_Vertical 模块 (竖式布局) 模块的 Nav_Link_Ord 可设置网站登陆后的默认视图	- 13 -
3.5 设置 NAV 文件.....	- 14 -

3.6	设置用户组.....	- 14 -
3.7	设置 Role.....	- 15 -
3.8	设置用户.....	- 15 -
4	按钮模块的使用.....	- 16 -
4.1	Page_Horizontal (横式布局)	- 16 -
4.2	Page_Vertical 模块 (竖式布局)	- 17 -
4.3	例子.....	- 17 -
4.4	上例效果.....	- 18 -
4.5	按钮链接属性 :	- 19 -
4.6	在 NavHttpURISTing 中输入 http:// www.sh-zs.cn 的效果.....	- 19 -
4.7	按钮报警图章显示 :	- 20 -
5	自定义图标.....	- 22 -
5.1	下载图标文件.....	- 22 -
5.2	引入图标.....	- 25 -
6	自定义背景色.....	- 27 -
6.1	设置属性.....	- 27 -
6.2	例子.....	- 27 -
6.3	结果.....	- 28 -
7	设置按钮自定义背景图片.....	- 28 -
7.1	Button_Sidebar (侧边栏按钮) 或 Button_Top (顶部按钮) 可设置背景图片 , 具体有如下两个属性 :	- 28 -
7.2	例子 :	- 29 -

7.3 结果..... - 29 -

8 自定义登陆页面..... - 30 -

8.1 设置登录模板..... - 30 -

8.2 设置登陆首页相关文件夹..... - 30 -

8.3 关于网页定制的相关说明..... - 35 -

1 基本界面概述

1.1 使用标准 HTML5 技术打造

本系统使用标准 HTML5 技术打造，使用通用的浏览器即可浏览，无需专用客户端，电脑、手机、平板均可登录使用本系统。

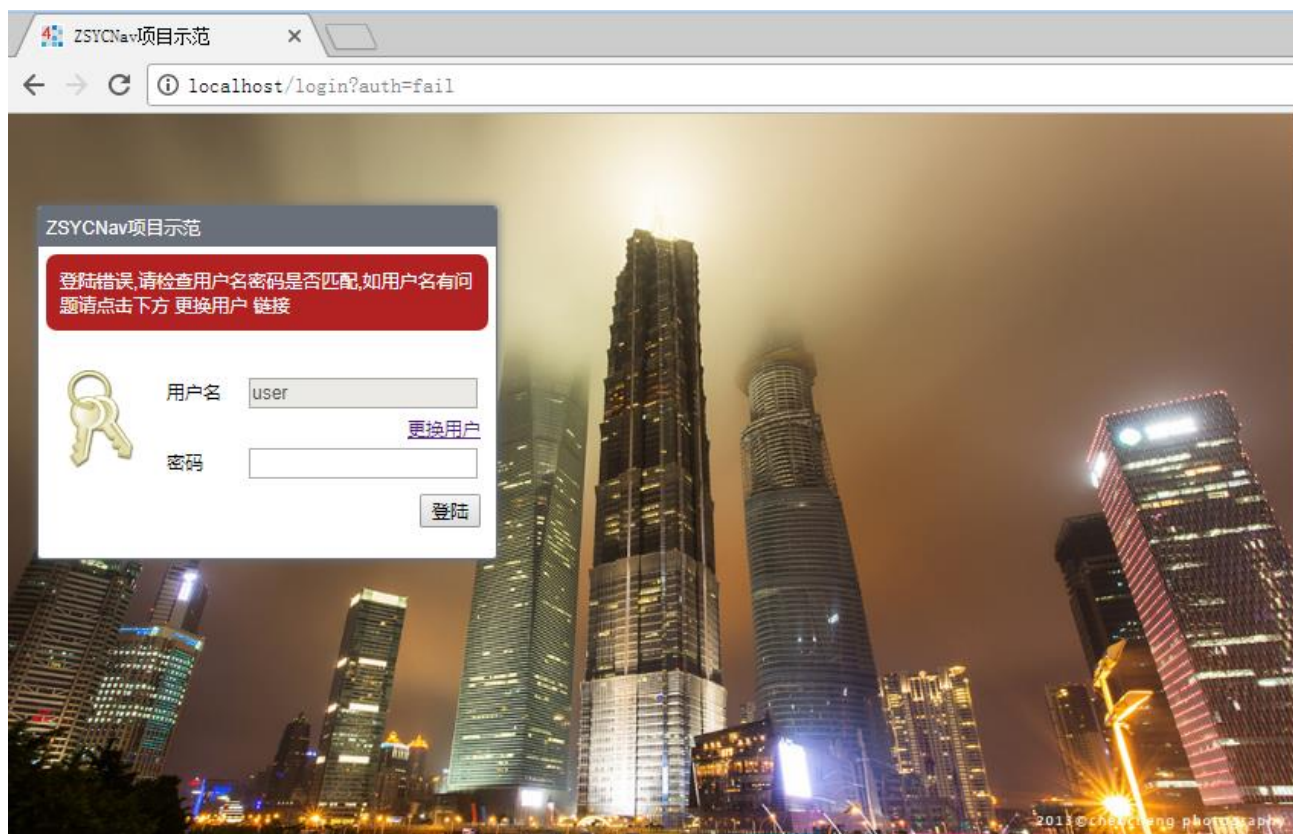
用户启动系统自带浏览器(如 IE11,Firefox 等,推荐 Firefox),并输入系统网址，即进入初次登录页面

1.2 定制登录页面

可定制登录页面。

可将已定制好的网页放入 Station 内，然后将其指定为登陆页面。

具体如何设置请参见本文第 7 章节所述，下图为简单举例：



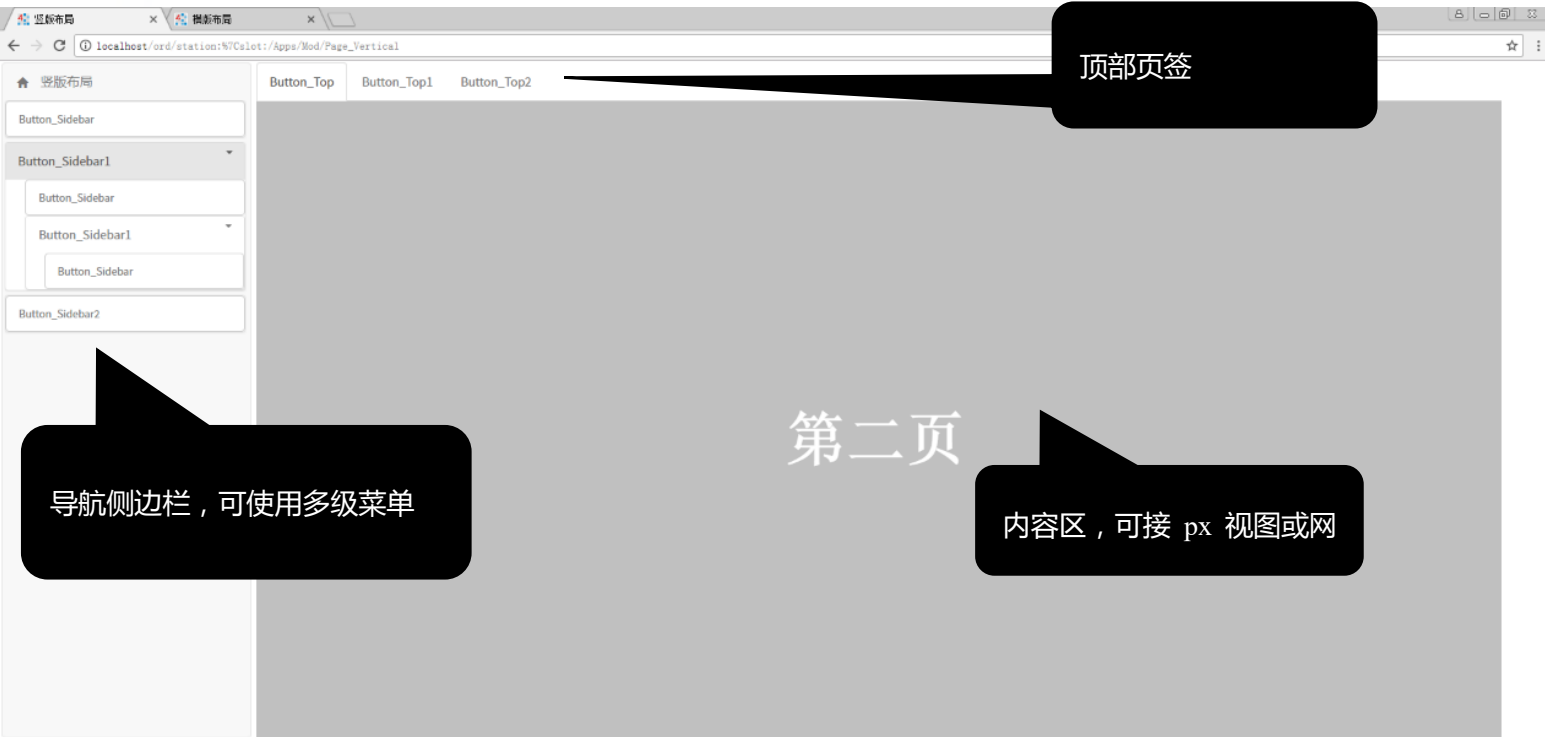
1.3 基本界面布局形式

页面基本布局使用 Bootstrap 技术，主要有两种布局：


- 横式布局：侧边栏会随顶部按钮的切换而切换



- 竖式布局：顶部页签会随侧边栏按钮切换而切换

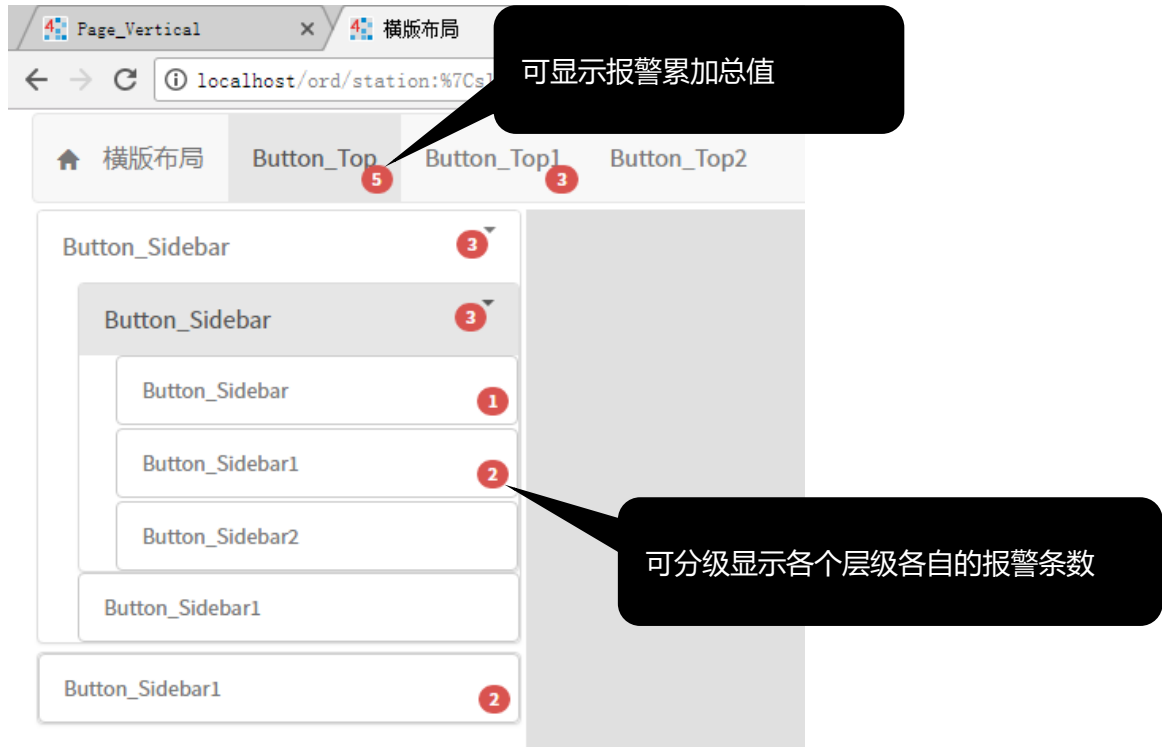


1.4主界面可根据屏幕宽度自适应

- 当浏览器窗口宽度缩小时，导航栏按钮将自动折叠。
- 当浏览器窗口宽度小于 768 像素时（比如使用手机浏览本系统网页），导航栏按钮将自动收缩成一个菜单按钮 ，点击该按钮可打开导航栏按钮列表。

1.5 警报

- 可直接在按钮中以小印章形式显示警报条数



- 可集中显示：

可在单一页面显示所有系统已发生警报的点的警报记录

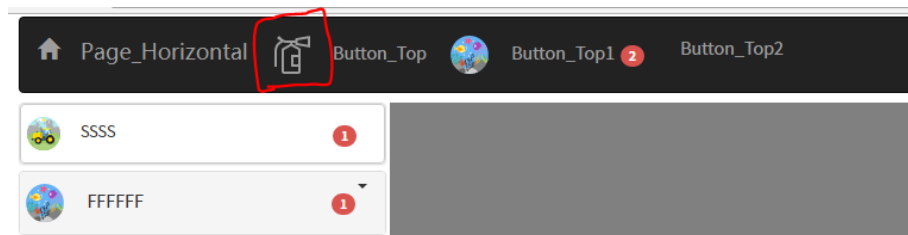
- 可分系统显示警报记录

可在各个系统内部有各自分系统的警报记录。

1.6 界面自定义

● 自定义按钮图标

可自引入外部字体按钮 JS 文件，自定义各个按钮的图标



● 自定义界面元素颜色

可自定义标题栏、按钮的背景颜色以及字体颜色，可自定义按钮按下后的背景色及字体颜色



● 自定义界面元素背景图片

可自定义按钮的背景图片，可自定义按钮按下后的背景图片



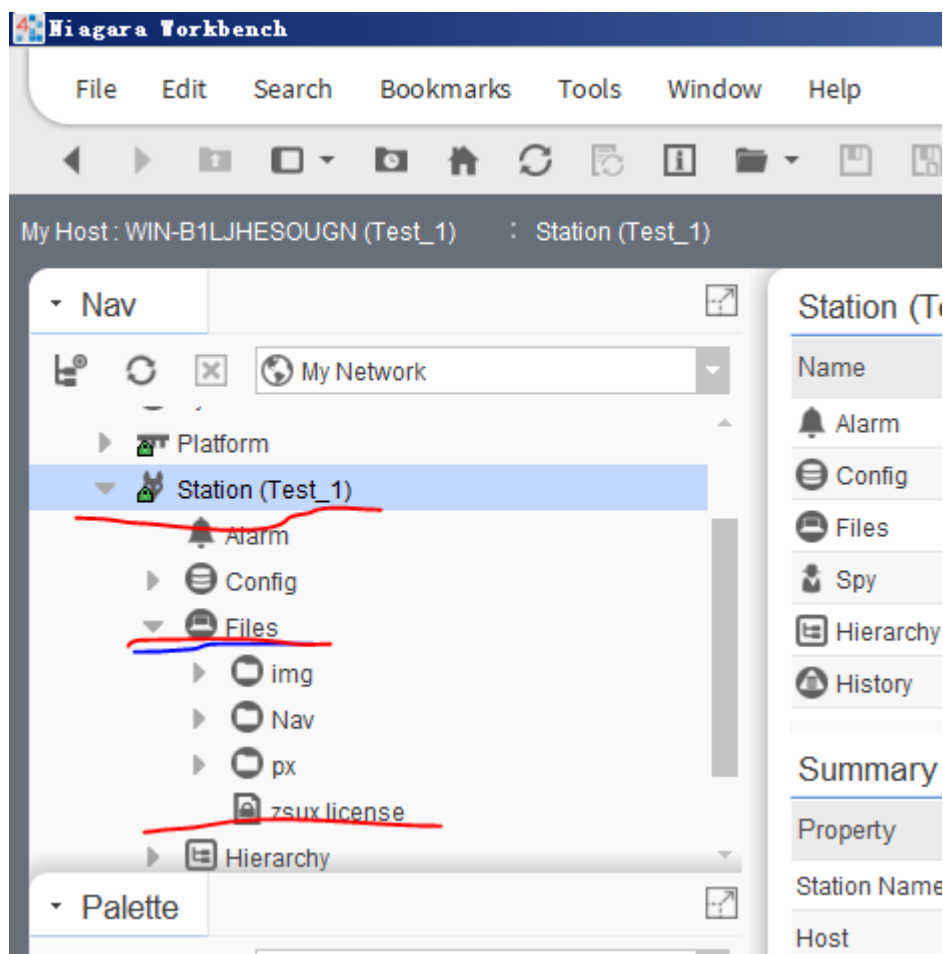
2 安装模块及授权文件

2.1 安装模块

将 ZSYCNV-ux.jar 和 ZSYCNV-rt.jar 拷贝至 niagara4 安装目录下的 modules 目录下

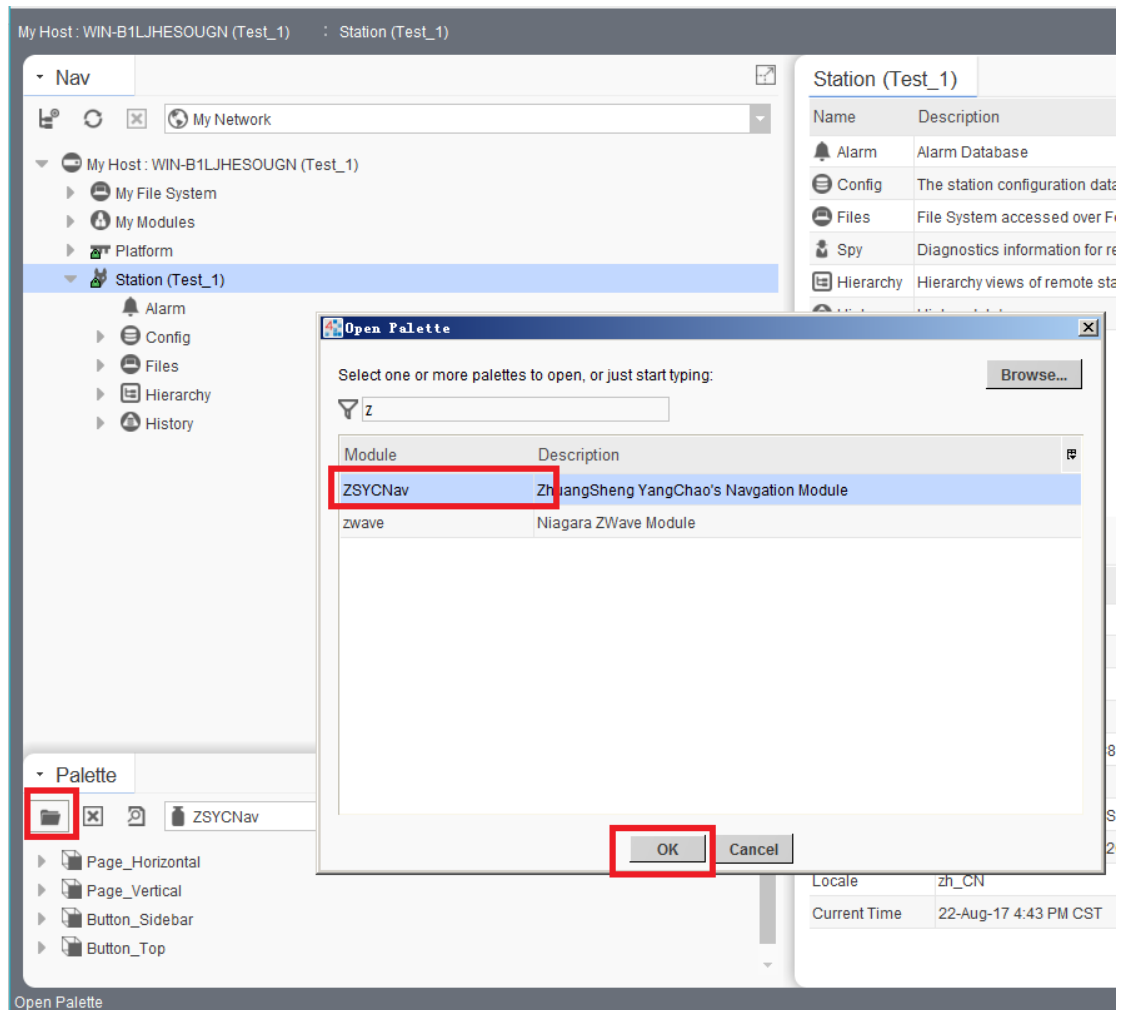
2.2 安装授权文件

在 Niagara4 的 WorkBench 界面中，将相应 Host ID 的 zsux.license 文件，拷贝至需要运行该模块的站点的 files 目录下。注意必须在 files 目录的根目录下。

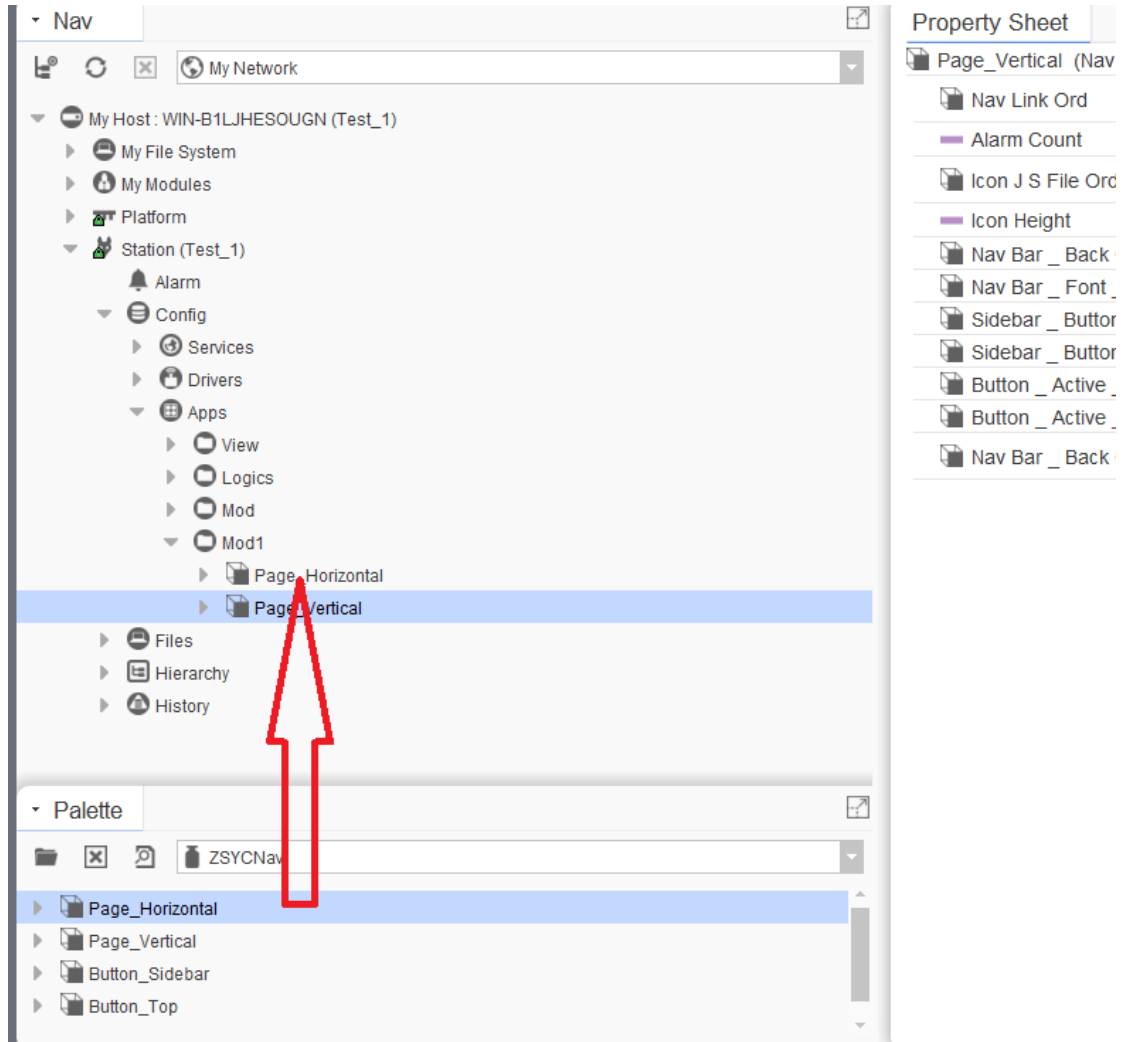


3 框架模块的简单使用

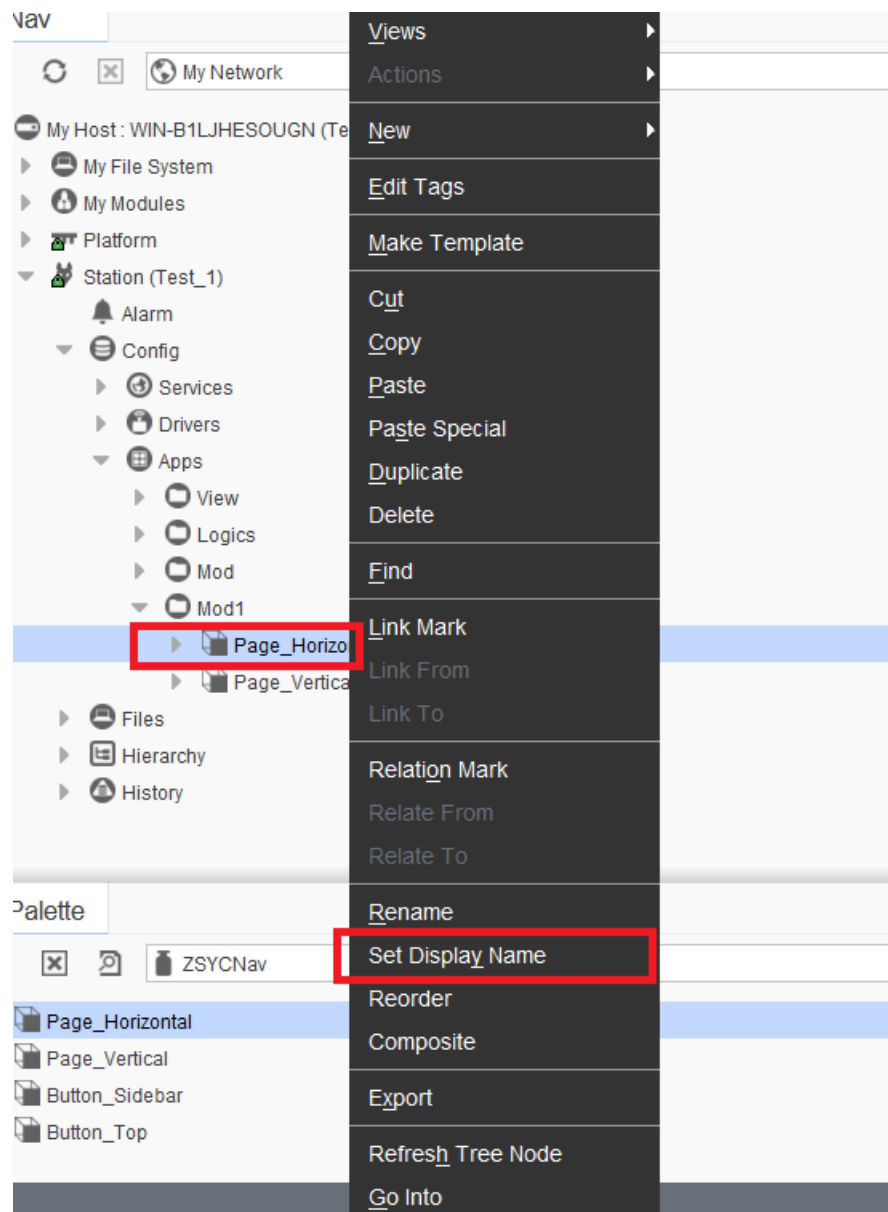
3.1 打开 ZSYCNav-ux 模块的 palate



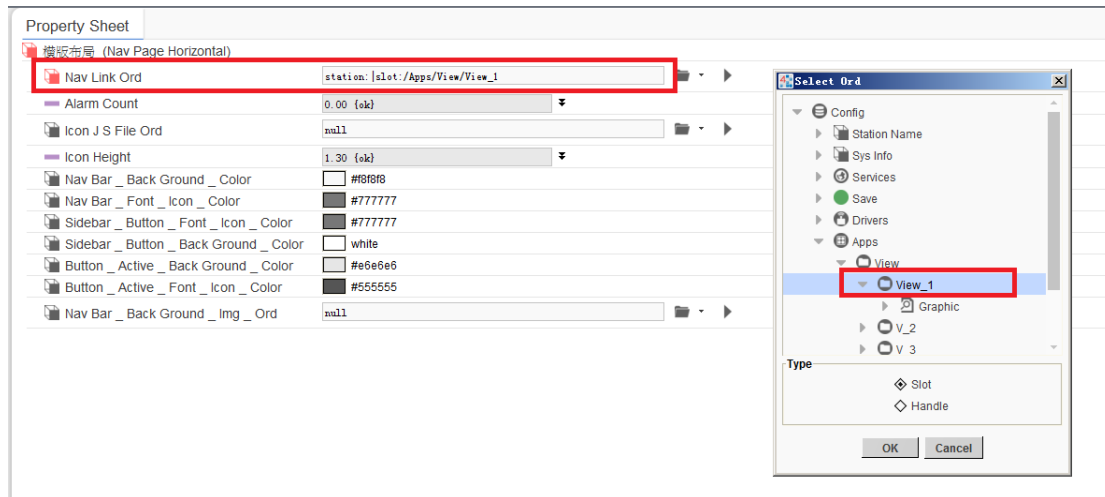
3.2 将 Page_Horizontal(横式布局)或者 Page_Vertical 模块（竖式布局）拖入文件夹中



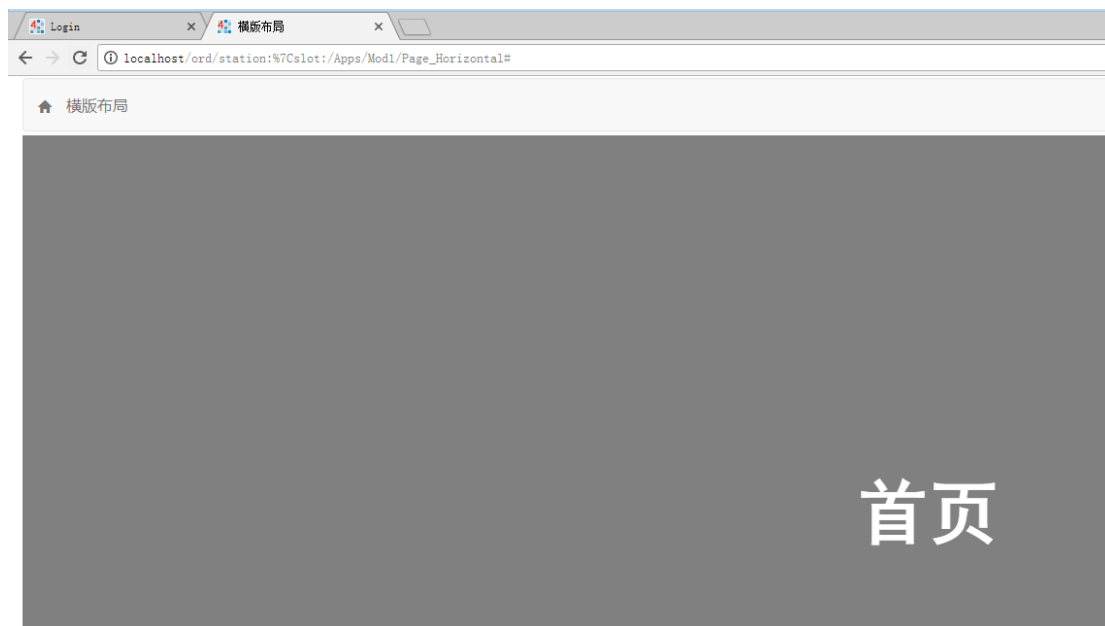
3.3 设置 Page_Horizontal （横式布局）或者 Page_Vertical 模块（竖式布局），其 displayName 即为网页标题。



3.4 Page_Horizontal（横式布局）或者 Page_Vertical 模块（竖式布局）模块的 Nav_Link_Ord 可设置网站 登陆后的默认视图

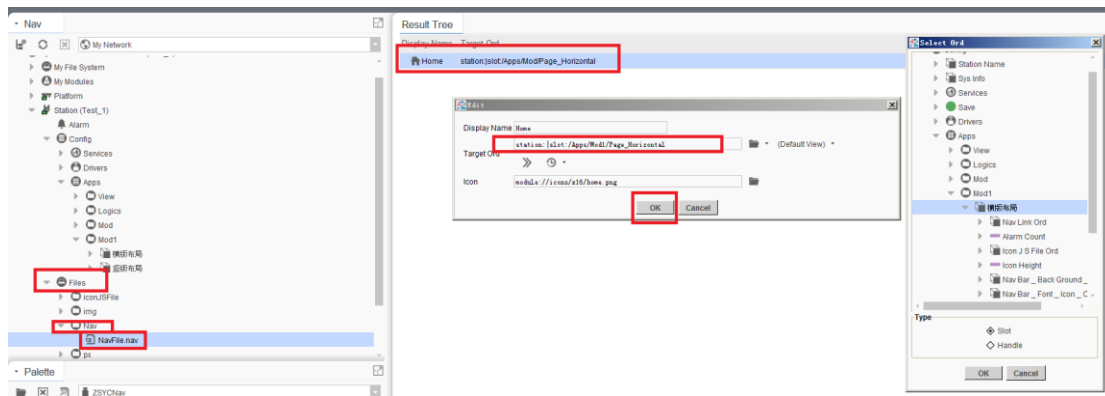


上述动作设置完后的首页效果如下

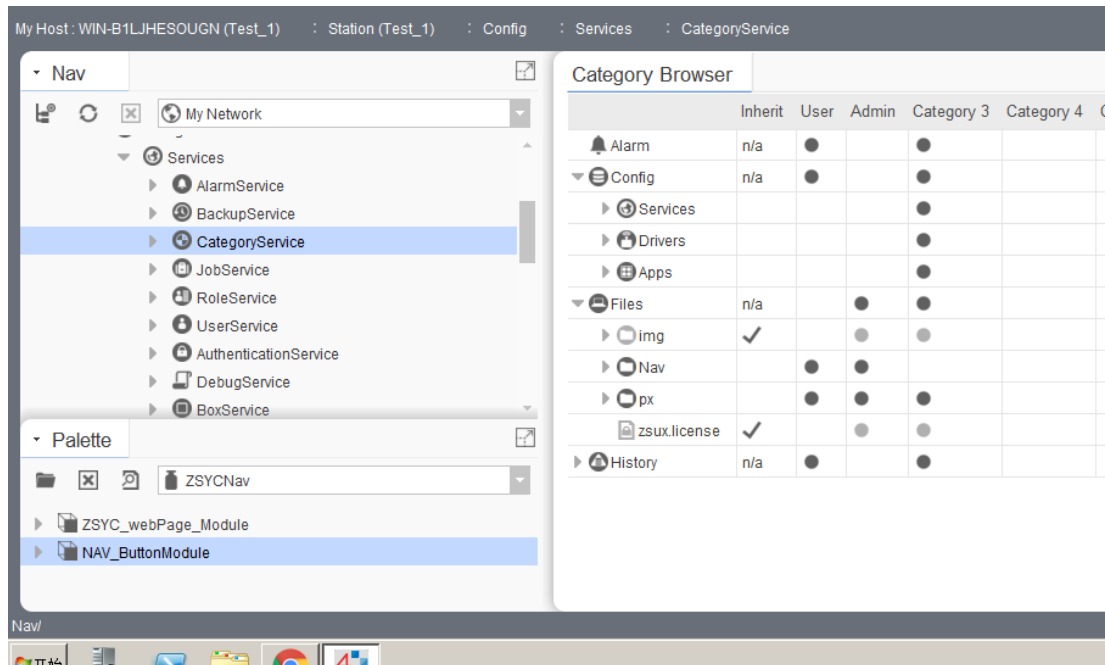


3.5 设置 NAV 文件

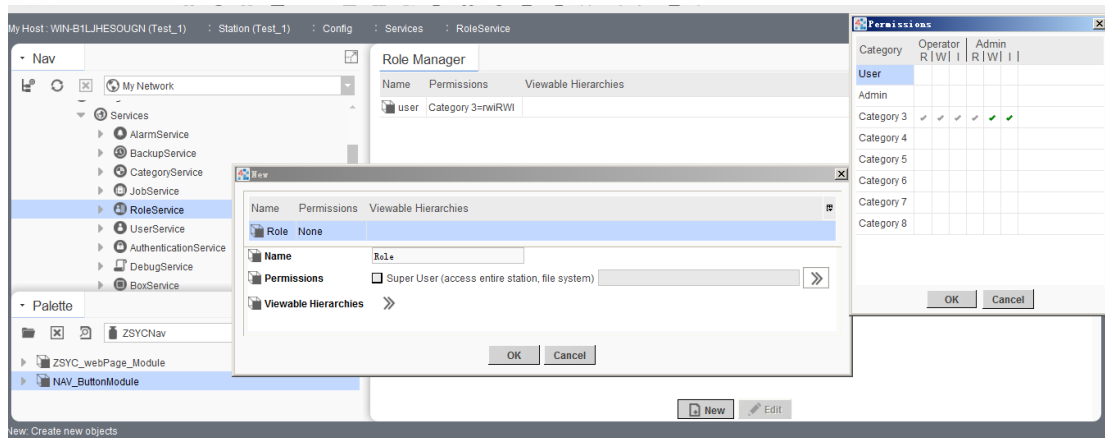
要想实现一登陆 Niagara4 就显示该网页框架，需设置用户首页指向，首先得设置 NAV 文件：在 Station 下新建 Nav 文件夹，在该文件夹下新建 NavFile.nav 文件，并修改该文件的 Home 属性



3.6 设置用户组



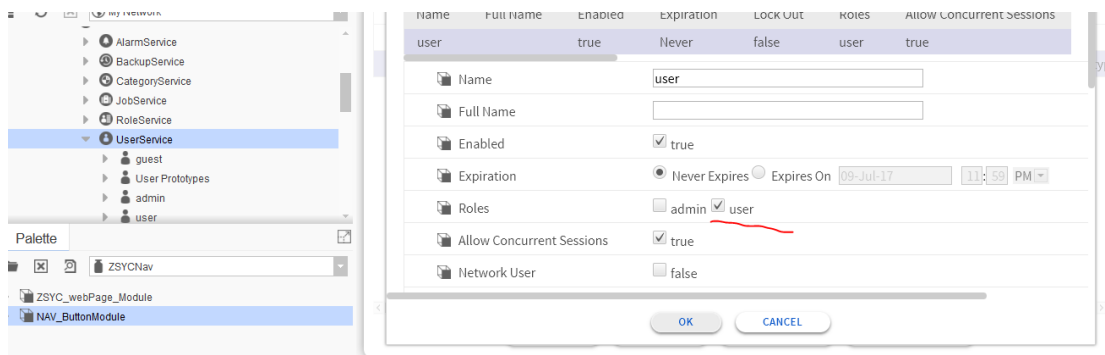
3.7 设置 Role



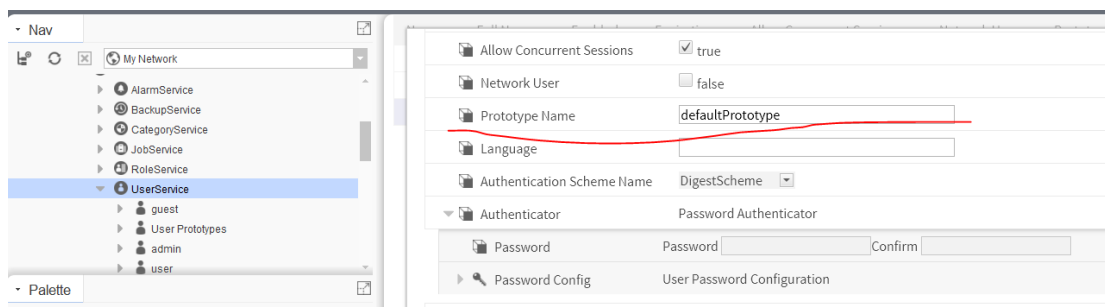
3.8 设置用户

3.8.1 新建用户，注意最好不要给用户用默认的 admin 用户，因为该用户为默认的管理员用户，拥有所有操纵站点的权限

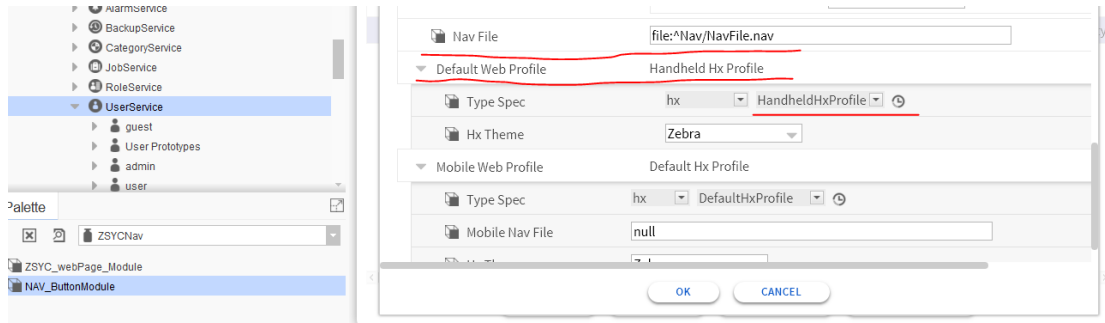
3.8.2 用户 role 的选择



3.8.3 用户 protoType 的设置



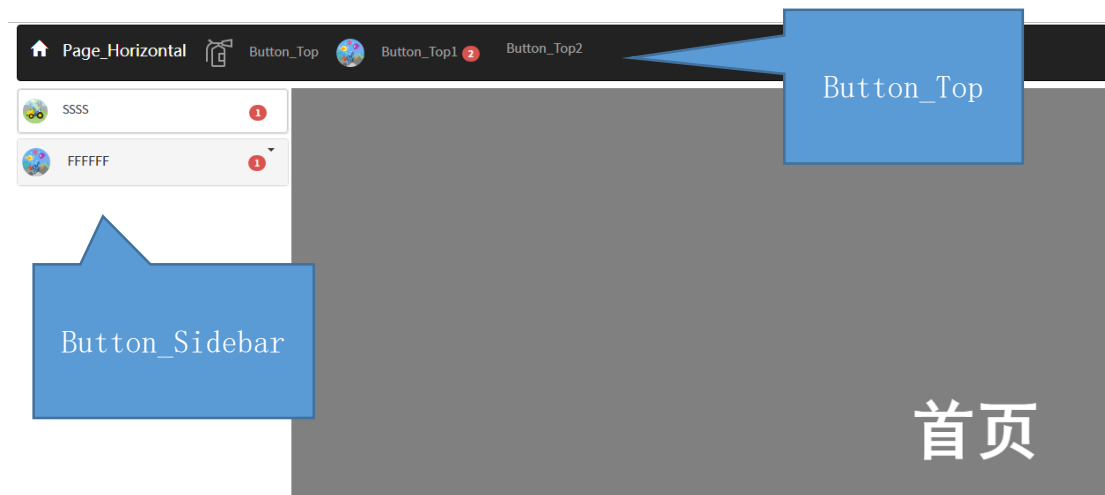
3.8.4 用户 NAV 文件和网页类型的选择



4 按钮模块的使用

将 Button_Sidebar (侧边栏按钮) 或 Button_Top (顶部按钮) 拖入 Page_Horizontal (横式布局) 或者 Page_Vertical 模块 (竖式布局) 模块下。

4.1 Page_Horizontal (横式布局)



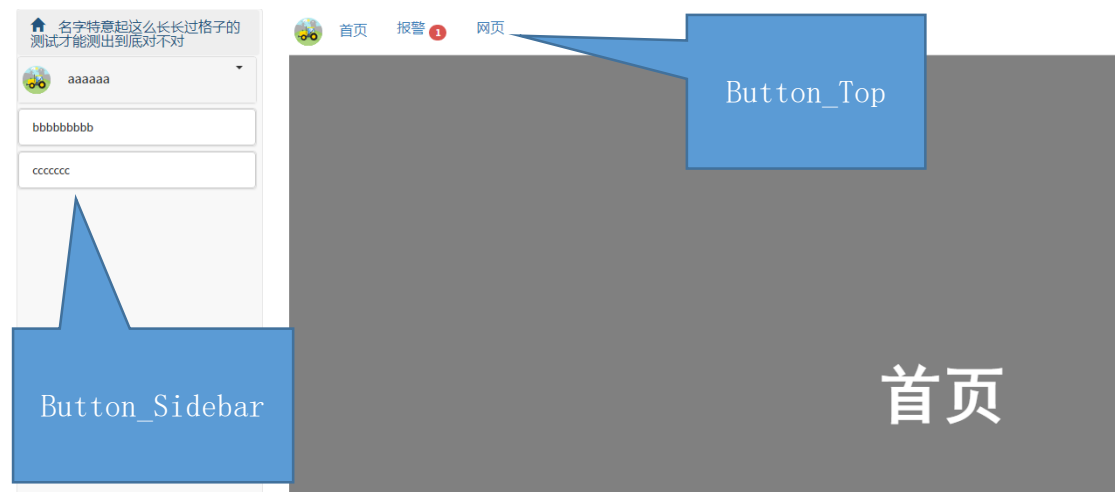
需要注意的是, 在 Page_Horizontal (横式布局) 中, Button_Top 对象只能在 Page_Horizontal 下有直属的一层, Button_Sidebar (侧边栏按钮) 下只能有嵌套的 Button_Sidebar (侧边栏按钮), 不能再嵌入 Button_Top (顶部按钮)。

在 Page_Horizontal 下直属的 Button_Sidebar (侧边栏按钮) 将是首页默认的侧边栏按钮。

在 Button_Top (顶部按钮) 对象下的多层 Button_Sidebar (侧边栏按钮) 将表现为按下相应

顶部按钮时切换出现的侧边栏按钮。

4.2 Page_Vertical 模块（竖式布局）



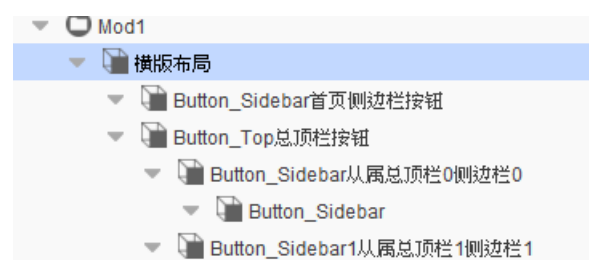
在 Page_Vertical 模块（竖式布局）中，Button_Sidebar（侧边栏按钮）下既可以有嵌套的 Button_Sidebar（侧边栏按钮），也能有 Button_Top（顶部按钮）。

在 Page_Vertical 模块（竖式布局）下直属的 Button_Top（顶部按钮）将是首页默认的页签按钮。

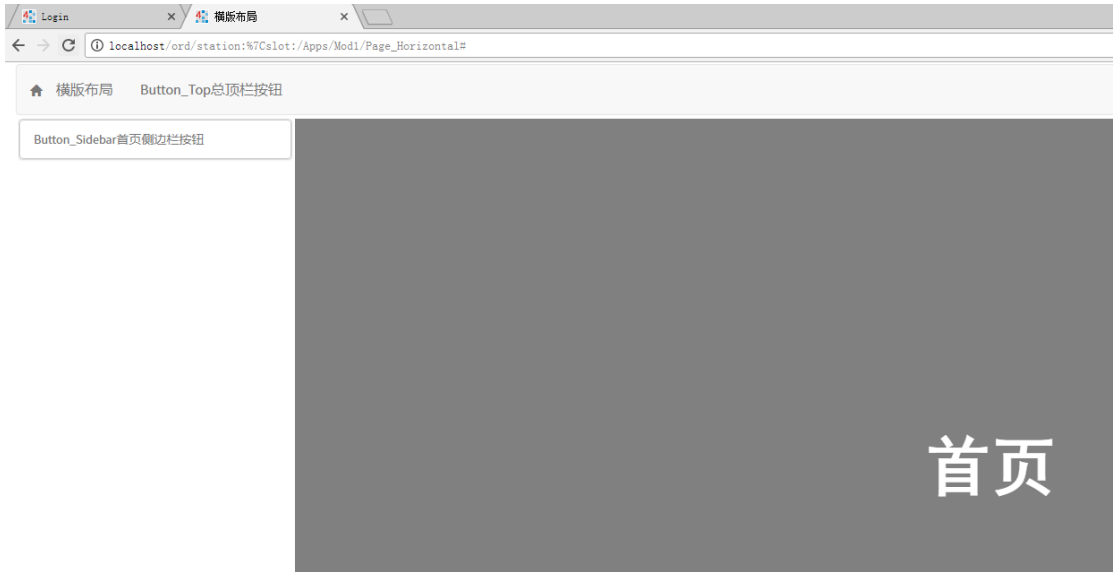
在 Page_Vertical 模块（竖式布局）下直属的多层 Button_Sidebar（侧边栏按钮）将表现为整个网站的固定的侧边导航栏的导航按钮。

在 Button_Sidebar（侧边栏按钮）下的 Button_Top（顶部按钮）对象将表现为按下相应侧边栏按钮时切换出现的顶部页签按钮。

4.3 例子



4.4 上例效果



4.5 按钮链接属性：

Button_Sidebar（侧边栏按钮）或 Button_Top（顶部按钮）的 NavHttpURISTing 属性与 Nav_Link_Ord 属性，这两个属性制定了点击该按钮后工作区显示哪个页面。

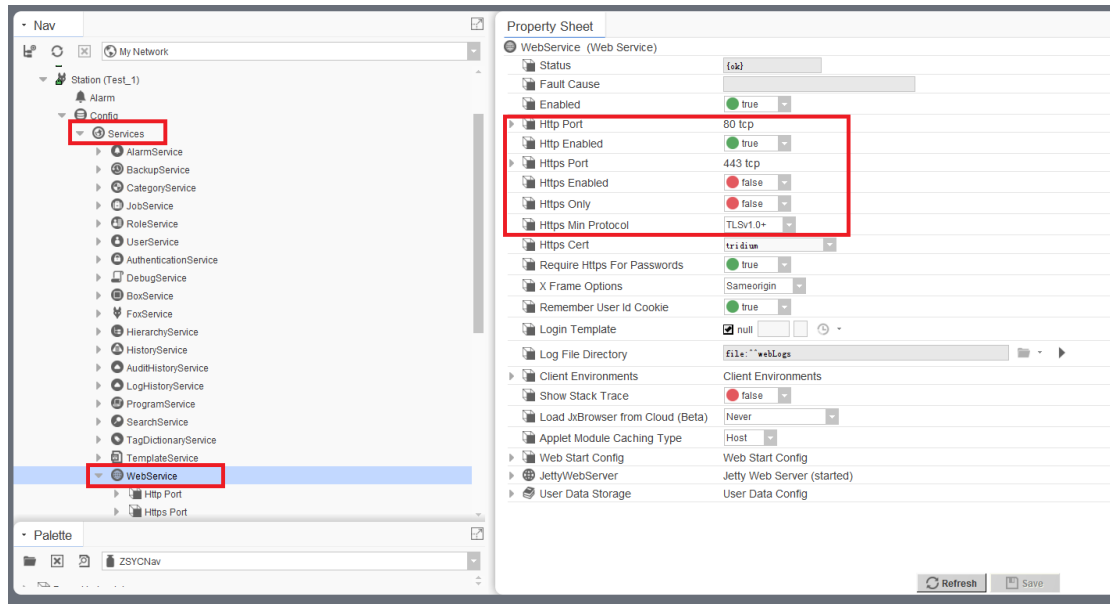
Property Sheet	
Button_Top总顶栏按钮 (Nav Button Top)	
Nav Http U R I String	http://www.sh-zs.cn {ok}
	<input type="checkbox"/> null http://www.sh-zs.cn
Nav Link Ord	null
Alarm Count	0.00 {ok}
Alarm Source Active Count	0
Alarm Source Ord	null
Icon String	{ok}
Button _ Back Ground _ Img _ Ord	null
Button _ Active _ Back Ground _ Img _ Ord	null

注意，NavHttpURISTing 必须包含其 Http 或 Https 的地址头部，比如如果想将 www.sh-zs.cn 整合至系统中，则必须在 NavHttpURISTing 中输入 http:// www.sh-zs.cn

4.6 在 NavHttpURISTing 中输入 http:// www.sh-zs.cn 的效果



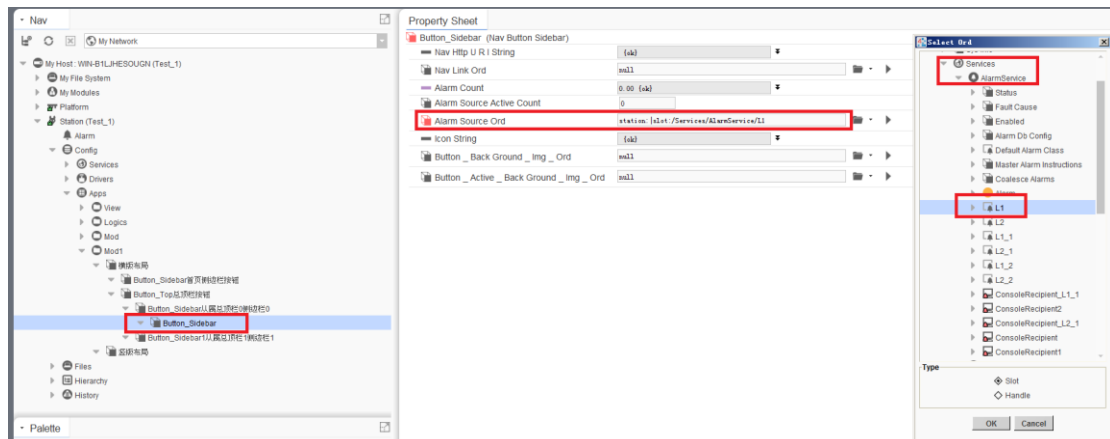
注意，如果引入的是网络安全协议的 `Https://` 开头网址，无需特殊设定，但如果需要在 `NavHttpURISting` 中整合进 `http://` 开头网址，则同时必须将 Niagara4 站点设置为 http 网站，设置方法如下：



注意，若这两个同时有值时，工作区将显示 `NavHttpURISting` 所指向网页，建议这两个值只有一个有值。

4.7 按钮报警图章显示：

`Alarmsourceord` 属性可以绑定站点的 `Services/AlarmService/` 下的某个 `AlarmService` 类，将指定的 `alarmService` 里的未确认报警条数绑定到该按钮的右下角小印章中。



注意，底层按钮的报警条数将自动汇总到其上层按钮中，前提是它上层的按钮的

Alarmsourceord 属性为空。

注意，若某个按钮的 Alarmsourceord 指向了某个 alarmService，则其报警值就是该

AlarmService 的报警数，而不会统计其下子模块的报警数目。



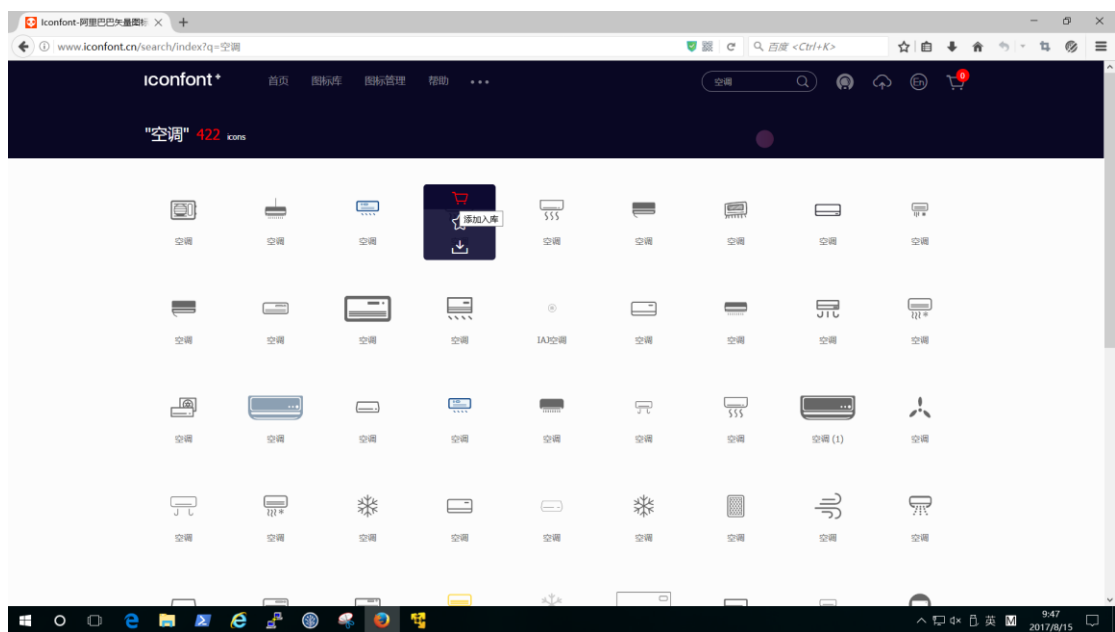
5 自定义图标

5.1 下载图标文件

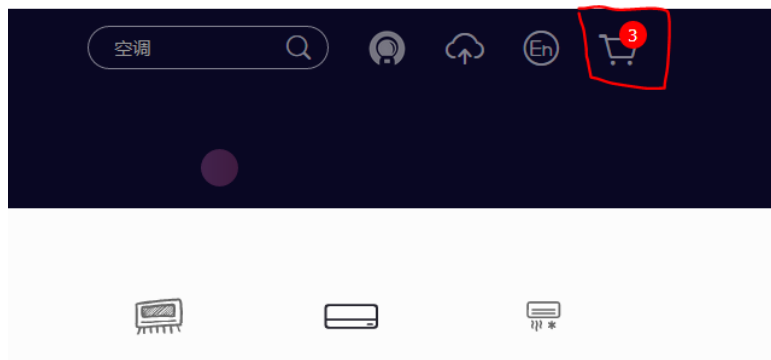
5.1.1 至 <http://www.iconfont.cn/> 上下载图标，登陆 <http://www.iconfont.cn/> 并使用注册使用账号登陆

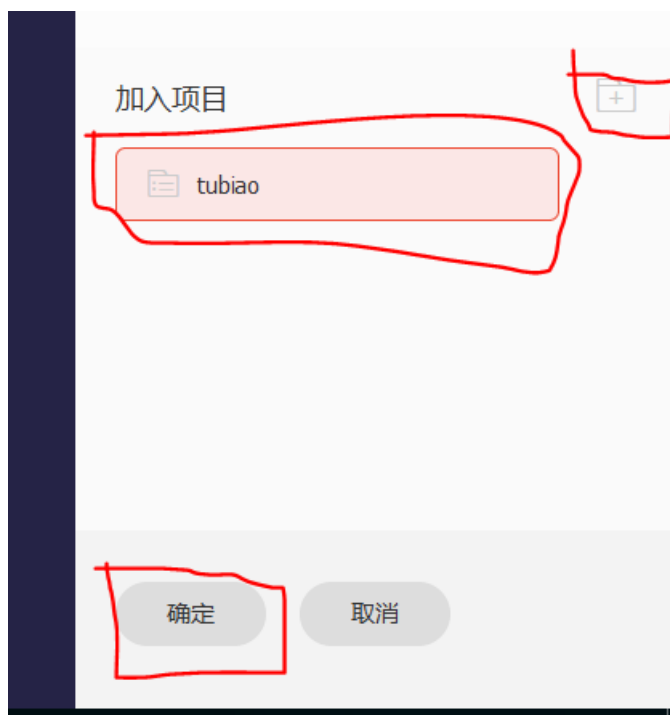
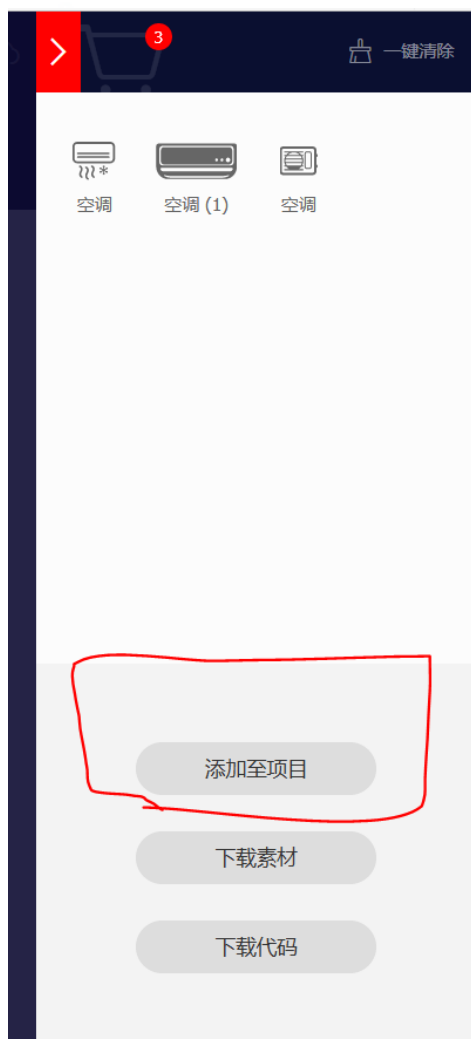
5.1.2 在 <http://www.iconfont.cn/> 搜索要使用的图标

5.1.3 将中意的图标添加至购物车里（基本都是免费）

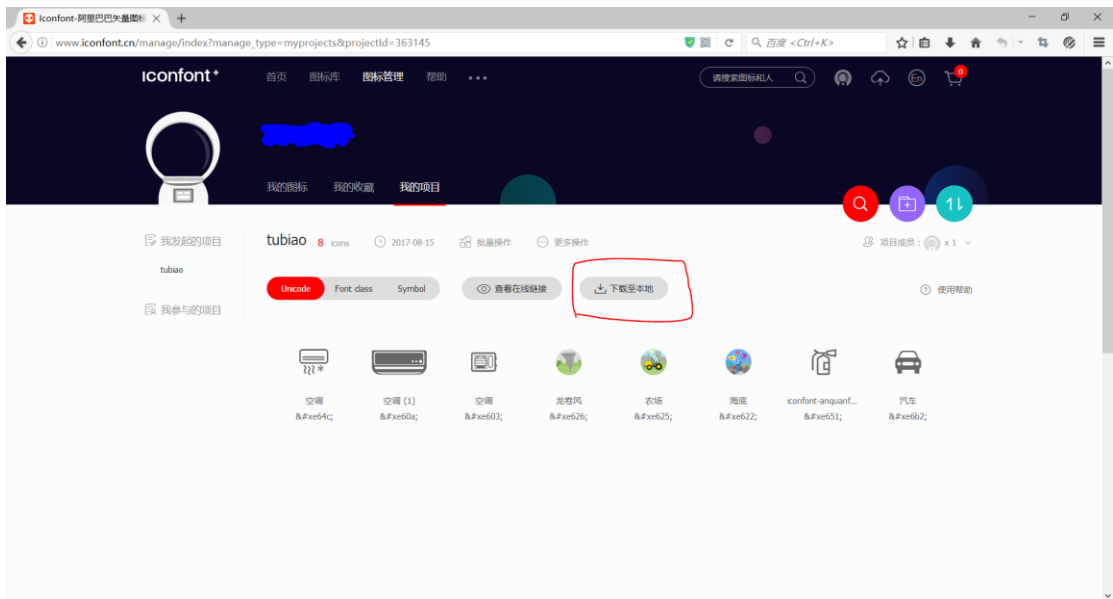


5.1.4 进入购物车，将选择好的几个图标添加至项目。

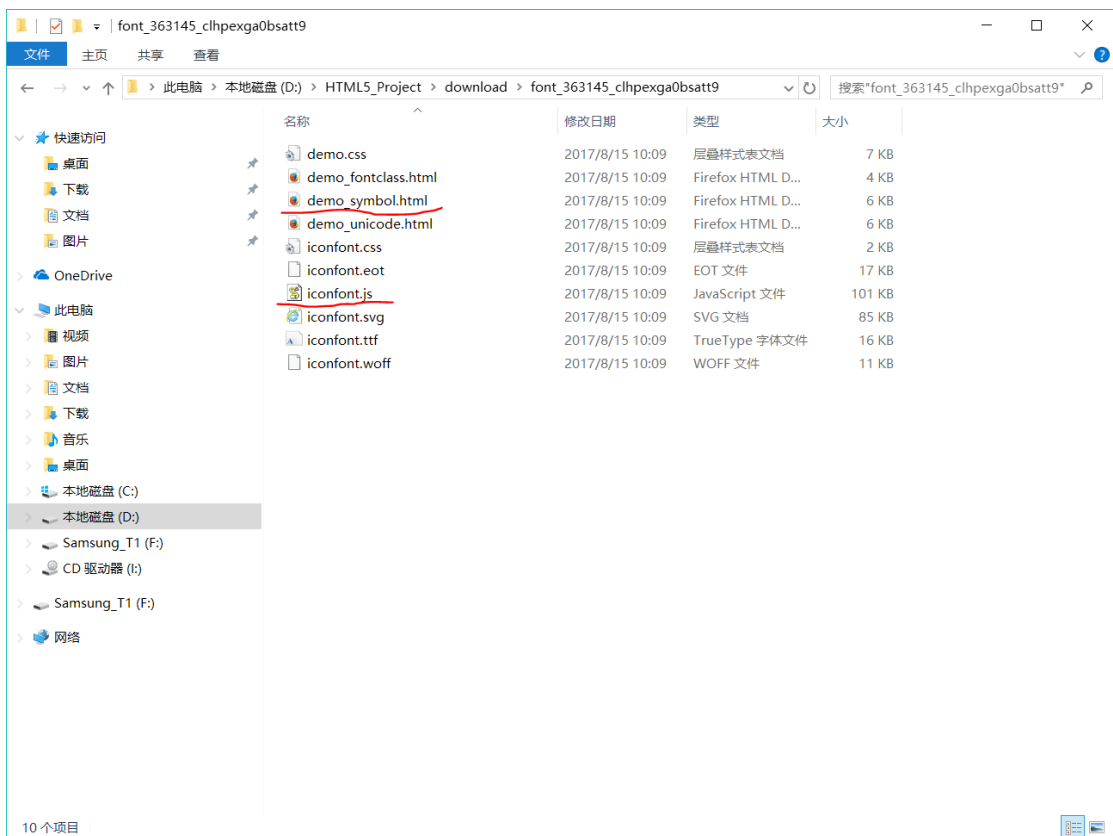




5.1.5 下载项目。



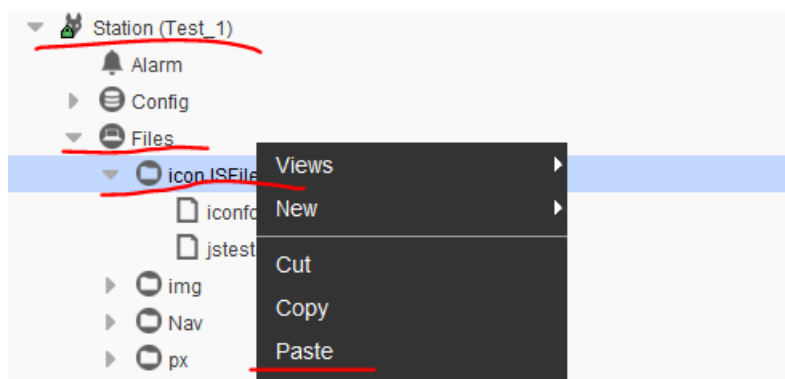
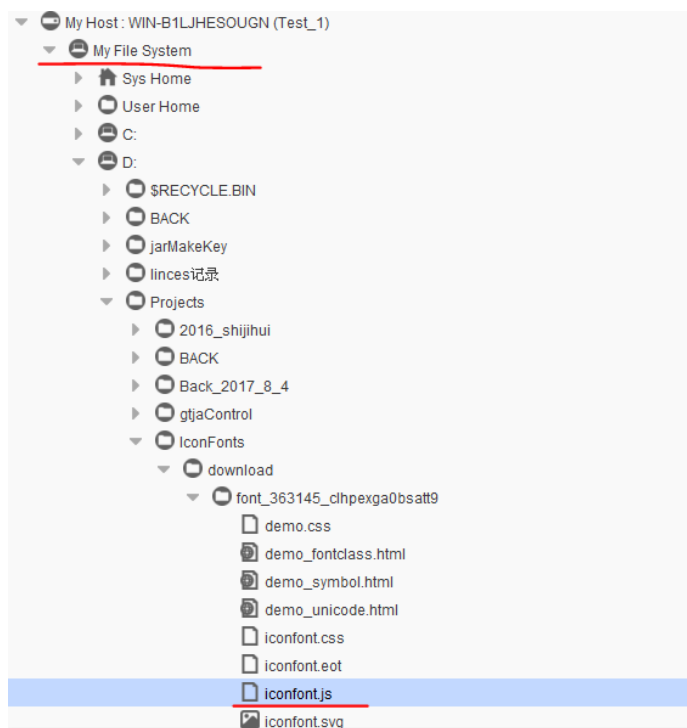
5.1.6 解压下载好的项目，浏览其中的 demo_symbol.html 网页，建议将各个图标下的文字标识都拷贝出来放到一个文本文件中，原因是因为有些图标的名称太长导致网页上两个图标的文字会重叠在一起，放到文本文件里就分开了。





5.2 引入图标

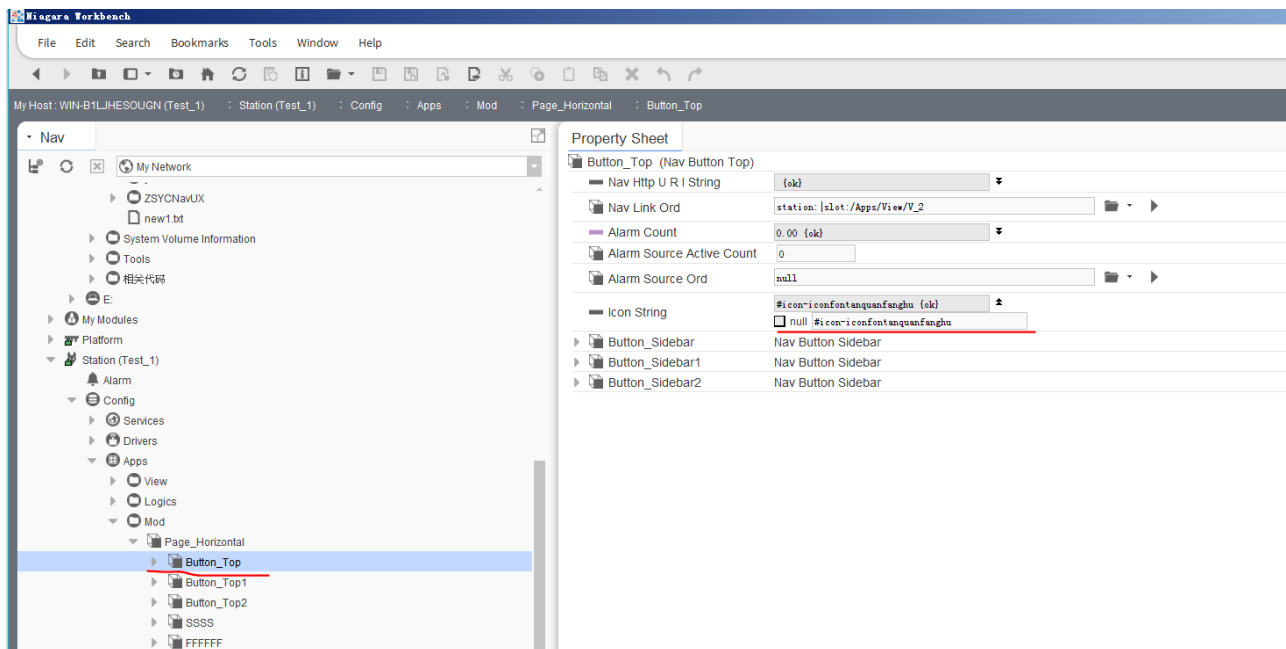
5.2.1 将 iconfont.js 拷贝至 station 的 files 下。



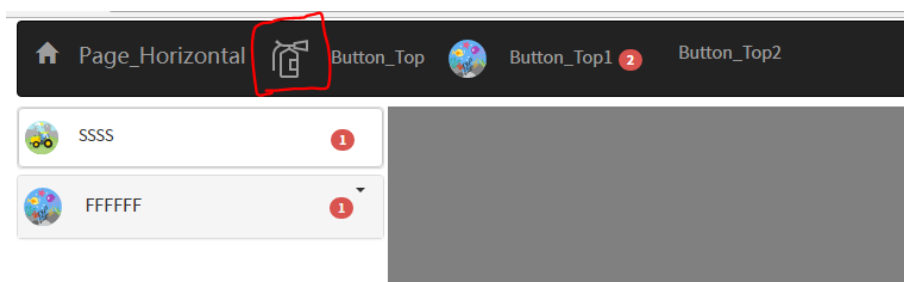
5.2.2 设置 Page_Horizontal 或 Page_Vertical 的 iconjsfileord 属性，引入 iconfont.js 文件。

5.2.3 设置 Page_Horizontal 或 Page_Vertical 的 iconHeight 属性，该属性设置整个系统的图标大小为默认字体大小的几倍，因为该属性会关系到按钮的高度，所以建议一般不要超过 3。

5.2.4 设置需要图标的 Button_Sidebar 或 Button_Top 的 iconString 属性，填入需要的图标的具体代号，即刚才拷入文本文件中的相应图标的以#号开头的字符串，例如本例中#icon-iconfontanquanfanghu 对应的是个消防灭火器图标。



5.2.5 结果如下。



需要注意的是，该方法既可以引入纯色图标，也可以引入彩色图标，纯色图标将和按钮文字一起变色，而彩色图标则只有一部分会随文字变色。

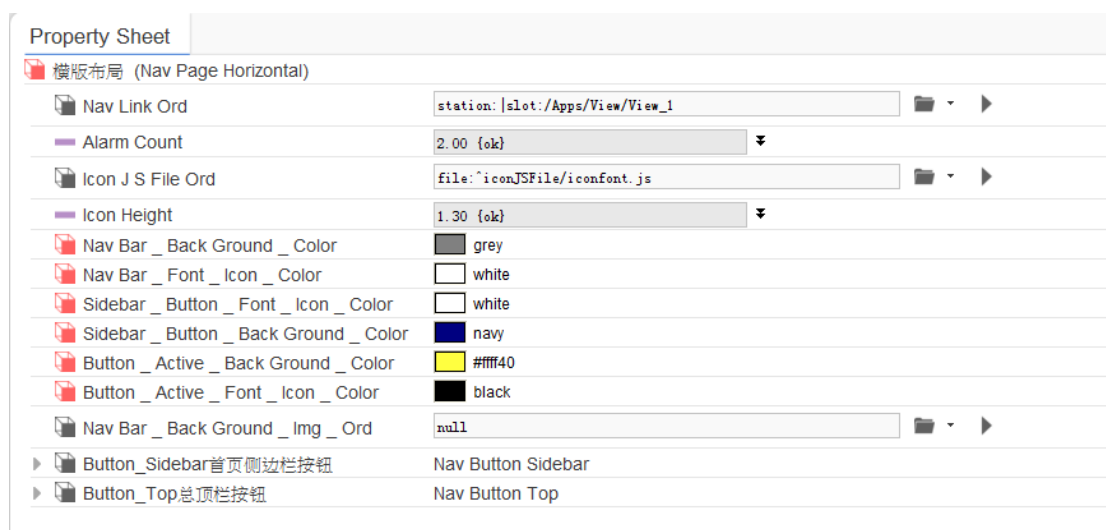
6 自定义背景色

6.1 设置属性

Page_Horizontal (横式布局) 或者 Page_Vertical 模块 (竖式布局) 模块下，都有自定义颜色属性，如下所示：

- navBar_BackGround_Color：该属性设定导航栏底色
- navBar_Font_Icon_Color：该属性设定导航栏上的文字与图标颜色。
- sidebar_Button_Font_Icon_Color：该属性设定侧边栏按钮未按下时的文字与图标颜色。
- sidebar_Button_BackGround_Color：该属性设定侧边栏按钮未按下时的背景颜色。
- button_Active_BackGround_Color：该属性设定侧边栏与导航栏按钮按下后的背景色。
- button_Active_Font_Icon_Color：该属性设定侧边栏与导航栏按钮按下后的文字与图标颜色。
- navBar_BackGround_Img_Ord：该属性设定导航栏背景图片。

6.2 例子



6.3 结果



7 设置按钮自定义背景图片

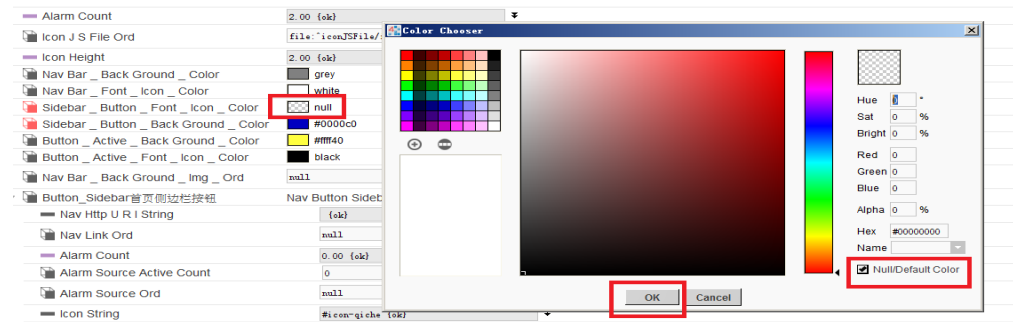
7.1 Button_Sidebar（侧边栏按钮）或 Button_Top（顶部按钮）可设置背景图片，具体有如下两个属性：

- Button _ Back Ground _ Img _ Ord 属性：设置按钮未按下时的背景图片。
- Button _ Active _ Back Ground _ Img _ Ord 属性：设置按钮按下后的背景图片。

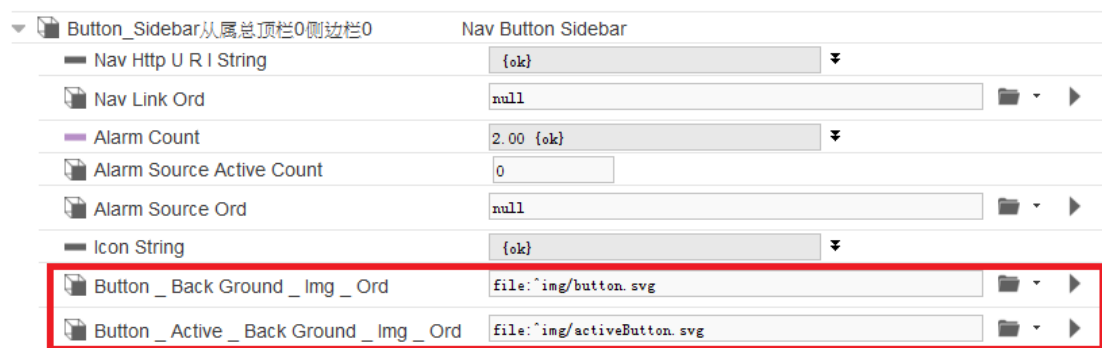
注意：

如需隐藏按钮本身的文字，使用图片里的文字替代，可设置 Page_Horizontal（横式布局）或者 Page_Vertical 模块（竖式布局）模块的 sidebar_Button_Font_Icon_Color 及 button_Active_Font_Icon_Color 为 null，这样这些文字会变透明，但这样做将同时意味着所有按钮的文字都将透明了，您将不得不为每一个按钮设置图片。该方法适用于您需要定制化

个性化的按钮以及按钮上的字体。



7.2 例子：



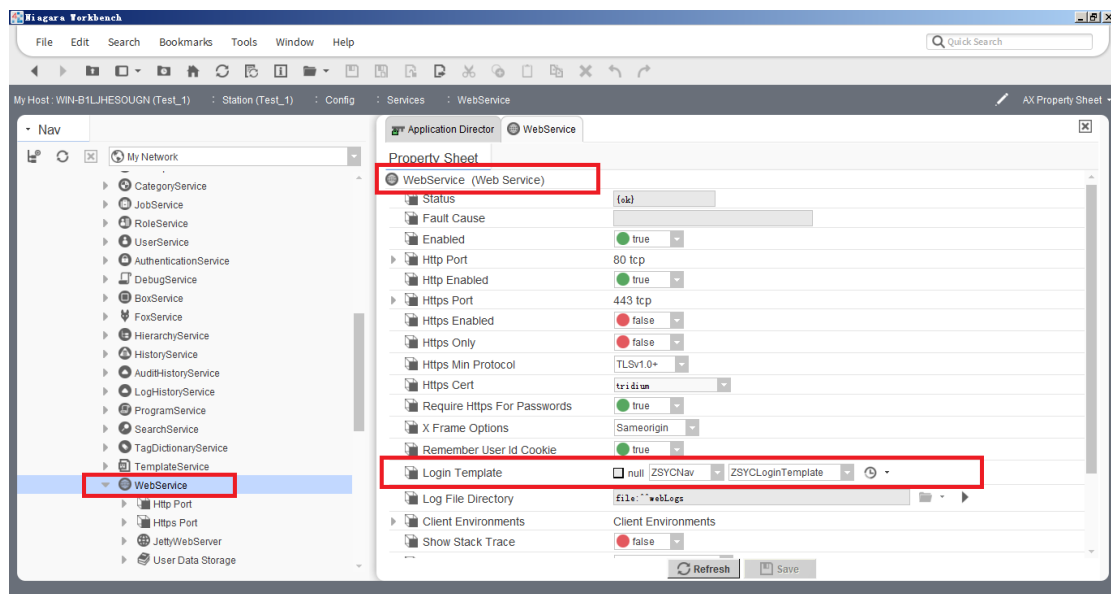
7.3 结果



8 自定义登陆页面

8.1 设置登录模板

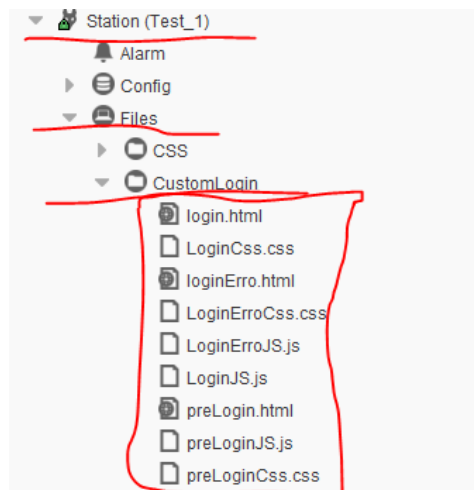
在站点的 webService 里设置该站点的登陆模板。



8.2 设置登陆首页相关文件夹

8.2.1 在站点的 files 下新建一个叫 CustomLogin 的文件夹，注意文件夹名必须为 CustomLogin

8.2.2 将自定义文件放入该文件中，自定义文件必须为如下文件名的几个文件：



- preLogin.html : 登陆时的第一个页面, 输入用户名。
- preLoginCss.css : preLogin.html 网页的 Css 文件, 内容会被插入至 preLogin.html 的 ID 为 preLginCSS 的 Dom 块中, 可以为空。
- preLoginJS.js : preLogin.html 网页的 javascript 文件, 内容会被插入至 preLogin.html 的 ID 为 preLginJS 的 Dom 块中, 可以为空。
- login.html : 登陆时的第二个页面, 输入密码。
- LoginCss.css : login.html 网页的 Css 文件, 内容会被插入至 login.html 的 ID 为 LginCSS 的 Dom 块中, 可以为空。
- LoginJS.js : login.html 网页的 javascript 文件, 内容会被插入至 login.html 的 ID 为 LginJS 的 Dom 块中
- loginErro.html : 如登陆错误, 则显示该网页, 如果是用户名没输入对, 请按切换用户链接回到 preLogin 界面重新输入用户名。
- LoginErroCss.css : loginErro.html 1 网页的 Css 文件, 内容会被插入至 loginErro.html 的 ID 为 loginErro CSS 的 Dom 块中
- LoginErroJS.js : loginErro.html 网页的 JavaScript 文件, 内容会被插入至 loginErro.html 的 ID 为 loginErroJS 的 Dom 块中

8.2.3 这三个网页最少必须包括如下代码（可添加 DOM 以及各种格式和属性，可

以修改如下示例中中文写明的可修改的文字的地方，但如下代码不可删减）：

- 预登陆网页 preLogin.html

```
<!DOCTYPE html>
<html>
<head>
  <meta name="viewport" content="width=device-width initial-scale=1.0
maximum-scale=1.0 target-densityDpi=medium-dpi">
  <title>登陆标题请在此修改</title>
  <script type="text/javascript" src="login/loginN4.js"></script>
  <style type="text/css" id="preLoginCSS">$cssContent</style>
  <script type="text/javascript" id="preLoginJS">$jsContent</script>
</head>
<body onload="checkFail()">
  <script type='text/javascript'>
    if ('ontouchstart' in window) {
      document.body.className += ' touch-enabled';
    }
  </script>
  <div>
    <noscript>JavaScript must be enabled to login</noscript>
  </div>
  <script type="text/javascript" src="login/digestLoginN4.js"></script>
  <script type="text/javascript" src="login/core/auth.min.js"></script>
  <form method='POST' action='prelogin'>
    <input id="username" class="login-input" type="text"
name="j_username" autofocus="">
    <input type='submit' name='submit' value="登陆按钮文字请在此修改"
onclick="return doLogin();" />
  </form>
</body>
</html>
```

● 登陆网页 login.html

```

<!DOCTYPE html>
<html>
<head>
  <meta name="viewport" content="width=device-width initial-scale=1.0
maximum-scale=1.0 target-densityDpi=medium-dpi">
  <title>登陆标题文字请在此修改</title>
  <script type="text/javascript" src="login/loginN4.js"></script>
  <style type="text/css" id="loginCSS">$cssContent</style>
  <script type="text/javascript" id="loginJS">$jsContent</script>
</head>
<body onload="checkFail()">
  <script type='text/javascript'>
    if ('ontouchstart' in window) {
      document.body.className += ' touch-enabled';
    }
  </script>
  <div>
    <noscript>JavaScript must be enabled to login</noscript>
  </div>
  <script type="text/javascript" src="login/digestLoginN4.js"></script>
  <script type="text/javascript" src="login/core/auth.min.js"></script>
  <form method='POST' action='j_security_check'>
    <input id="username" class="login-input" type="text"
name="j_username" value="$initUsername" readonly disabled>
    <input id="password" class="login-input" type="password"
name="j_password" autocomplete="off" autofocus="">
    <input type='submit' name='submit' value="登陆按钮文字请在此修
改" onclick="return doLogin();" />
  </form>
  <div class='login-change-user'>
    <a href="/prelogin?clear=true">更换用户文字请在此修改</a>
  </div>
</body>
</html>

```

- 登陆错误网页 loginErro.heml

```

<!DOCTYPE html>
<html>
<head>
  <meta name="viewport" content="width=device-width initial-
scale=1.0 maximum-scale=1.0 target-densityDpi=medium-dpi">
  <title>登陆标题文字请在此修改</title>
  <script type="text/javascript" src="login/loginN4.js"></script>
  <style type="text/css" id="loginErroCSS">$cssContent</style>
  <script type="text/javascript" id="loginErroJS">$jsContent</script>
</head>
<body onload="checkFail()">
<script type='text/javascript'>
  if ('ontouchstart' in window) {
    document.body.className += ' touch-enabled';
  }
</script>
<div>
  <noscript>JavaScript must be enabled to login</noscript>
</div>
<div id="login-failed">
  登陆错误文字请在此修改
</div>
<script type="text/javascript" src="login/digestLoginN4.js"></script>
<script type="text/javascript" src="login/core/auth.min.js"></script>
<form method='POST' action='j_security_check'>
  <input id="username" class="login-input" type="text"
name="j_username" value="$initUsername" readonly disabled>
  <input id="password" class="login-input" type="password"
name="j_password" autocomplete="off" autofocus="">
  <input type='submit' name='submit' value="登陆按钮文字请在此修
改" onclick="return doLogin();" />
</form>
<div class='login-change-user'>
  <a href="/prelogin?clear=true">更换用户链接文字请在此修改
</a>
</div>
</body>
</html>

```

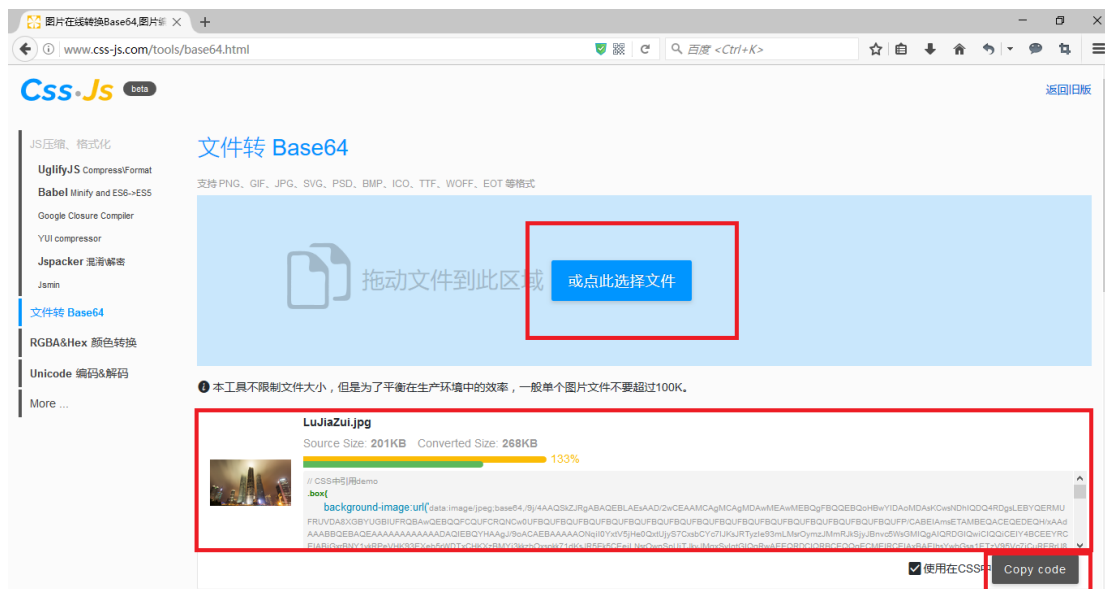
8.3 关于网页定制的相关说明

8.3.1 必须使用预登陆网页先登陆用户名，也必须使用“更换用户”的链接才能回到预登陆页面，更换用户必须使用预登陆网页，这个是 Niagara4 的新机制，没得商量。

8.3.2 登陆网页无法引入站点内 files 下的动态文件资源，这是 Niagara4 里的安全机制规定的，这意味着您或者使用 Internet 网上的线上资源作为图片来源，或者以 Data URI scheme 形式直接将图片转换成的 BASE64 字符串写入 HTML 网页中或 CSS 网页中。

8.3.3 将图片转换为 BASE64 字符串的方法如下：

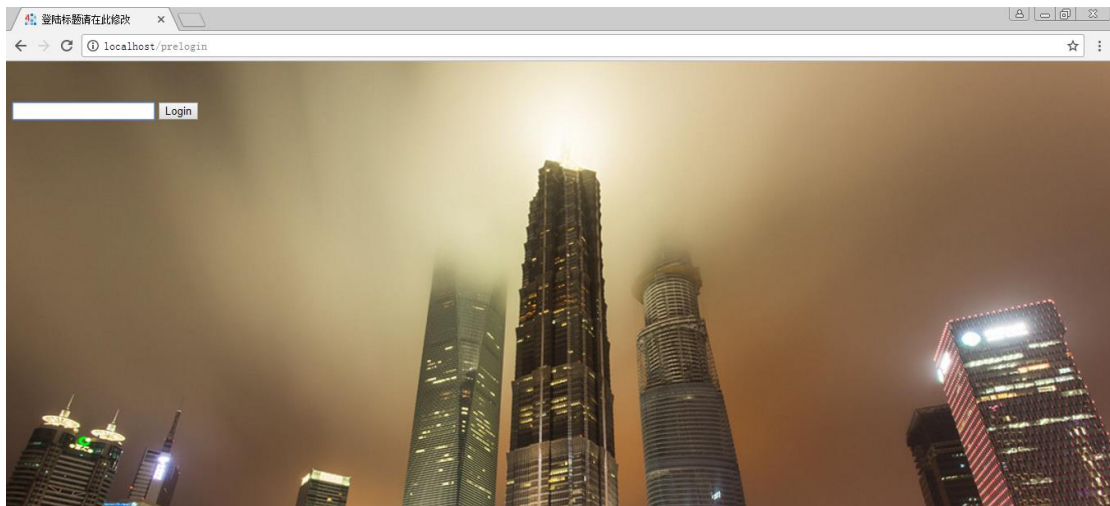
- 进入 <http://www.css-js.com/tools/base64.html> 网站
- 将转换得到的代码拷贝到需要使用的 CSS 文件或 HTML 文件的相应位置中



8.3.4 例如，我可以在 preLoginCss.css 里添加如下代码

```
body {
    padding-top: 40px;
    padding-bottom: 40px;
    background-repeat: no-repeat;
    background-size: 100%, 100%;
    /*background-position: center;*/
    background-image: url("data:image/jpg;base64,/9j/4AAQSkZJRg...
上例所得该图片的BASE64代码...");
}
```

8.3.5 则该站点的预登陆页面如下所示



更多如何美化网页的知识请参考 HTML 以及 CSS 以及 JavaScript 等网页相关内容，或者干脆外包找专门的网页制作人员来完成，只要网页内容包含了上文所说的那些必要的 DOM，其他怎么改本文就不赘述了。当然，也可参考本人提供的“自定义模板登陆网页修改示范案例”，该案例将上文中的几个必备 Form 和 Input 部件进行了组织，并添加了一些说明性文字，添加 CSS 格式，并在 CSS 文件中添加了背景图，具体可见案例中文件。