**报表开发规范**

作者：Merrier

版本：1.0

更新时间：2017-3-16

写在开头：

1. 所有的页面截图都是后台工程的页面，可能会和前端工程所看到的页面有一些不同，不过了解整个报表系统的逻辑就够了，不需要对页面过分纠结。
2. 以下提到的所有“模块”均是一个图表，而报表是模块拖拽拼装之后的多个图表。
3. 报表系统的所有页面都在web\_design/contents\_folder里，所以下面写的路径都是在这个目录下的相对路径

页面：

1. 模块展示页面，这里展示了所有的模块，用户可以对其进行排序和查看图例：（form\_center\_final/form\_center\_module.html）

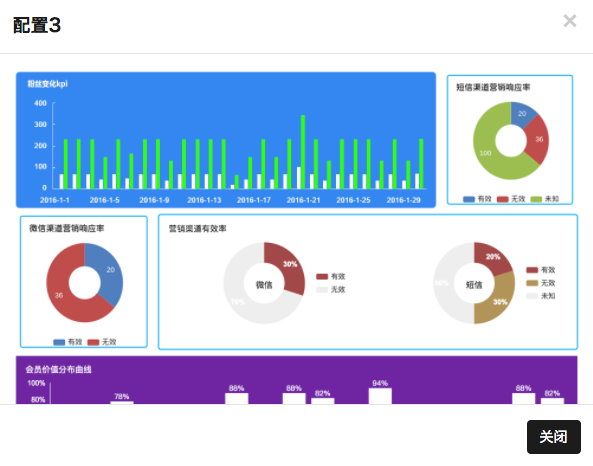


1. 报表展示页面，这里展示了所有用户配置好的报表，用户可以创建新报表、设为首页、看其图例、编辑名称和序号、查看（跳转到“报表查看”页面）、配置（跳转到“报表配置”页面）以及删除

（form\_center\_final/form\_center\_form.html）

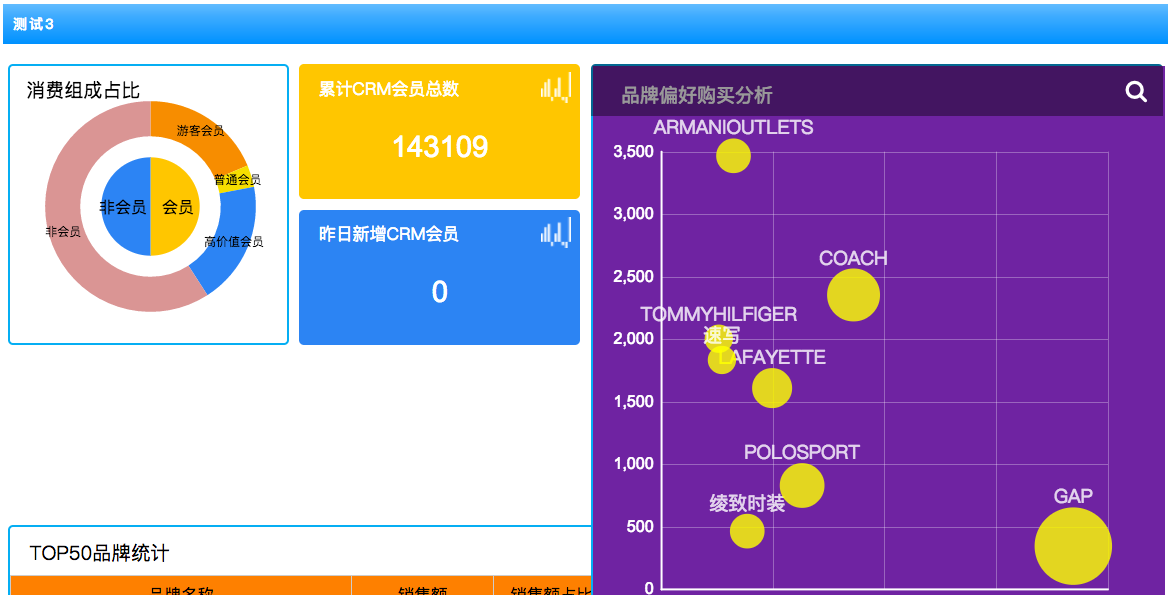


点击图例会弹出蒙版，并且显示这张报表的示例图：



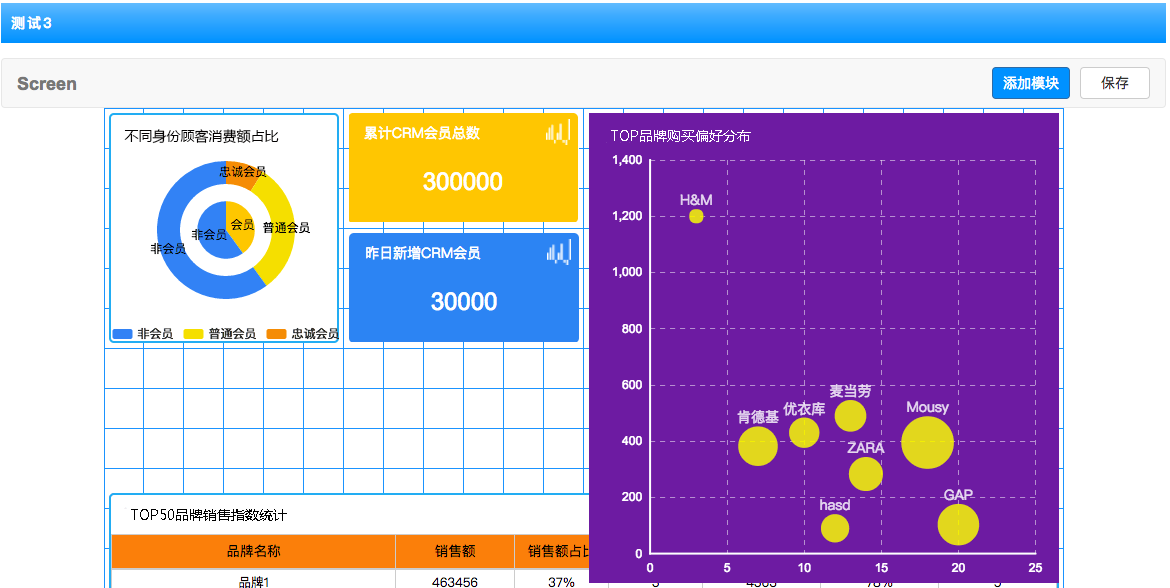
1. 报表查看页面，这里用户可以看到其之前配置好的报表，其位置是之前其在“报表拖拽”页面配置好的，同时会根据屏幕宽度进行适配（关于适配，会在下面讲到），这个页面在前端的工程可能会看不到图表，因为没有数据的原因，但是只需要理解逻辑就可以了，这部分不需要我们写静态页面，一切都是按照写好的js绘制的。

（form\_center\_final/form\_modules\_show.html）



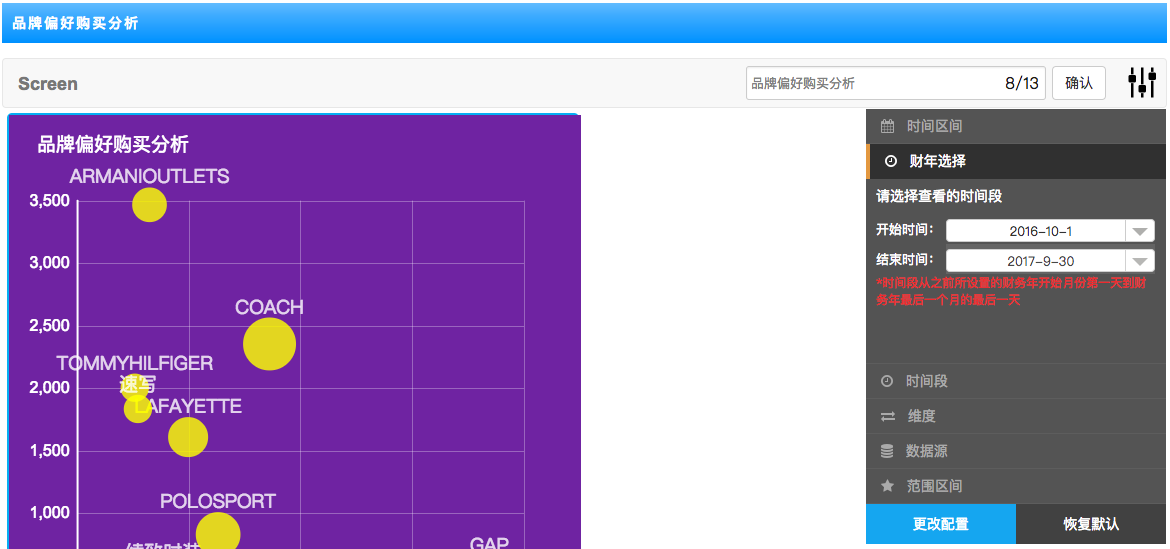
1. 报表配置页面，用户可以在这里对某个报表进行拖拽重定位，或点击添加模块按钮导入新的模块到该报表；用户可以点击保存按钮保存对该报表的配置，然后前端会将配置好的报表生成一张图片（也就是在“报表展示”页面用户点击图例时出现的那张图片），同时传给后台存储到数据库

（form\_center\_final/form\_draggable\_show.html）



1. 模块配置页面，用户可以在这里对某个模块进行重新“绘制”，从下面的截图中可以看到，用户可以设置时间、维度、数据源、范围区间等选项，这些选项的更改会影响到改模块绘制所用到的数据，从而会改变模块的代表意义（比如从年度变成了月度），需要注意的是，该页面也不需要前端书写，后台有相应模板，可以自己套用，所以在前端工程中查看也有可能看不到模块，只会看到右侧配置项列表。

（modules\_details下的所有html）



1. 模块静态页面，这里是唯一需要前端书写的页面，具体书写规则如下（截图为modules\_folder/modules\_folder\_template.html，该页面也是前端报表开发人员会经常用到的一个页面，后面会讲）：

（1）title需要修改为模块的中文名称



（2）div.form\_brick的修改（module\_id为所开发模块的英文名称，可直接将中文名称进行百度翻译或有道翻译，然后将其格式化为全部小写字母+“\_”组合）：

a.首先需要先把注释修改对，就是把模块的中文名称填上

b.然后把所有为module\_id的id或class替换成模块的英文名称（如果该模块内部只有最多一个div会进行绘制初始化，则其id为module\_id\_canvas，就像截图中那样，如果有多于一个进行绘制的容器，就命名为module\_id\_first或module\_id\_left类似这种，具体可参见其他比较复杂的模块命名方式）

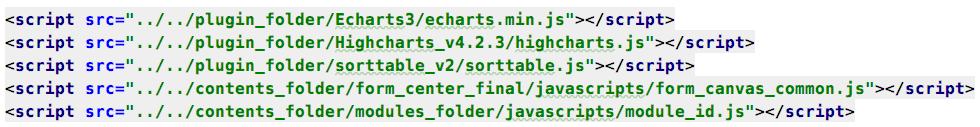
c.将id为module\_id的div的类名进行修改，其中module\_id\_load为必须类，form\_bg\_类型的类为可选类，其css为整个模块添加一个2px的border，有些模块可能需要添加，视具体情况而定，form\_class类型的类为可选类，因为有些模块可能样式类似，统一加一个类的话方便书写样式，具体可参见document\_folder下的报表总结.doc

d.将id为module\_id的div的其他属性进行修改，其中包括data-width为模块的宽度，其值为小于等于24的数字，计算规则为模块在960宽度盒子所占比例（举个例子，比如模块宽度为710px，则其所占比例为（710+10）/ 40 = 18，加的10px为每个模块最外层的5pxborder宽度，），data-height同理，而style中可以写其在960px宽度盒子中的宽高，参见下图

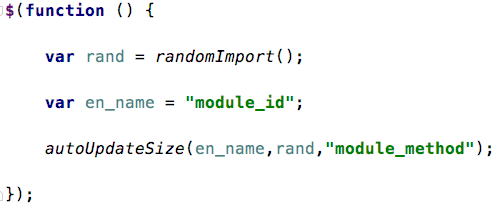
e.修改id为module\_id的div下的所有div的data-radio，从下图可以看到，如果id为module\_id的div下有多个进行展示的div（比如下图中一个用来绘制的div.module\_id\_canvas，还有一个表格div.module\_id\_table），那么他们两个div在这个大的盒子中的比例就需要以data-radio的形式确定，data-height\_radio为0.4就以为着该div的高度占整个盒子高度的40%，同理其他data-width\_radio，data-top\_radio等等，需要注意的是，data-width\_radio和data-height\_radio的默认值为1，而data-top\_radio、data-bottom\_radio、data-left\_radio、data-right\_radio的默认值为0



f.然后还需要在body的后面添加自己的js，js的命名为模块的英文名



g.在module\_id.js中进行相应修改（module\_method为该模块绘制所用的方法，如果之前有类似模块则可以重用，若没有则需要新建）



h.若该模块并没有可以重用的方法（可以去报表总结.doc查看有没有类似的模块），则必须到form\_center\_final/javascripts/form\_canvas\_common.js中新建模块所用方法，需要修改的内容包括：作者（自己的名字）、图表类型（较为具体的描述一下该方法所绘制的报表的样式，具体可查看其他方法）、所用插件、图表个数、图表名称，还有方法名字，然后就是方法内部的内容了，被注释掉得“// var dataArray =…”是不需要删除的，这是后台获取数据的方式，而“//假数据”下面的“var dataArray…”是前端根据图表设定的假数据，为json对象格式；form\_opt配置项中的有些内容是公用的，比如title、grid等等，可以查看其他方法



总结：如果需要做新报表，负责开发的前端人员只需要负责modules\_folder目录下的对应页面，然后按照上面步骤进行修改和书写，最后一定要更新报表总结.doc，如果有任何变动，请记得更新此文档。

补充：

1. 所有的报表系统专用样式都在form\_draggable\_show.less中
2. 关于适配，我在上面没有介绍，主要思路是按照.content\_mb这个div的宽度进行调整，而由于左侧导航菜单的宽度为220px，而.content\_mb这个div左右分别有27px的margin，所以最后的最大宽度是window.width() – (220 + 27)，这也是好多js中出现的win\_width – 274的来由
3. 在我书写这份文档之前，已经完成了好多种样式的模块，可以去报表总结.doc查看，所以在做模块之前请一定要确定之前是否做过，以免麻烦自己
4. 拖拽页面的适配由于时间关系还没有完成，但是其实现可以模仿form\_canvas\_common中的autoUpdateSize（）方法。