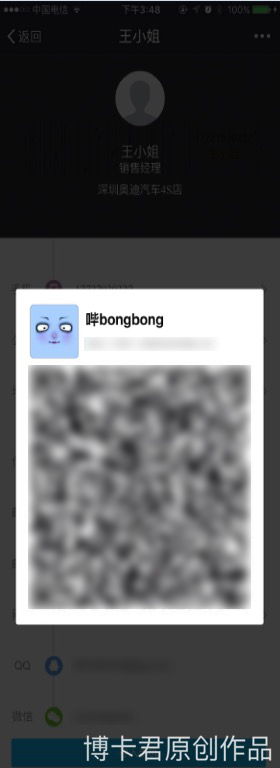
第五章：微信小程序名片夹详情页开发





备注下大致需求：顶部背后是轮播图，二维码按钮弹出模态框信息、点击微信栏、点击存入手机，地址栏需要地图展示，名片分享也是模态框指引。

首先是轮播图，autoplay 自动播放，interval 轮播的时间，duration 切换速度，可以根据自己需求去添加。

Delete：是删除按钮，加载进来是隐藏的，需用户点轮播图进去后，轮播图全屏显示才出来。  
noClickImg 与 ClickImg：切换全屏与非全屏轮播图，绑定了点击事件 changeClick 来切换，只是改变样式即可。  
Block：图片列表。  
Number\_img：当前轮播 index（currentNumber），与图片 length 集合（cardnum）。  
其中 currentNumber：

//轮播图发生改变时改变数字

//初始化数据

Data:{

currentNumber:1

}

slidechange:function(e){

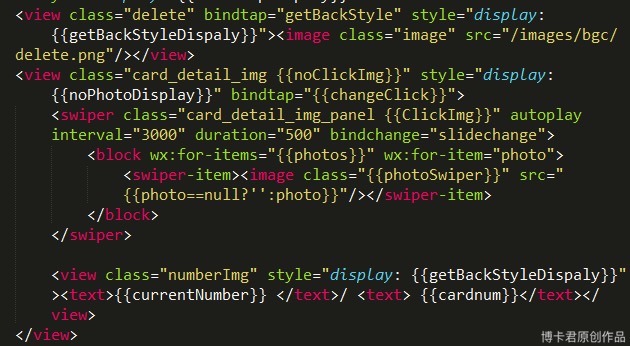
var number = e.detail.current;

this.setData({

currentNumber:number+1

})

},



这里可以看到全屏状态下当关闭按钮被点击后 getBackStyle，把 changeClick 切换到 imgFullScrenn 待命。



再次点击返回原样式，

https://static.oschina.net/uploads/img/201609/29140957_bgzd.jpg  
  
切换后事件又走回到 getBackStyle 了，灵活运用。



刷新下开发者工具可以看到具体效果如下：



详情页可以看到信息基本都是样式一样，可以使用微信提供的循环 block。   
下面是详情页里面的个人信息数据，

如果有信息就显示出来，没有数据的不显示，这里使用

https://static.oschina.net/uploads/img/201609/29140957_5jtp.jpg

//中文信息

               var chinaMessage = res.card.groups[0].fields;

var personMessage= []

               for(var i = 0;i<chinaMessage.length;i++){

                personMessage.push(chinaMessage[i])

               }

               //为空或者null是不显示判断

              for(var k in personMessage){

                if(personMessage[k].value==null || personMessage[k].value==""){

                personMessage[k]["display"] = "none";

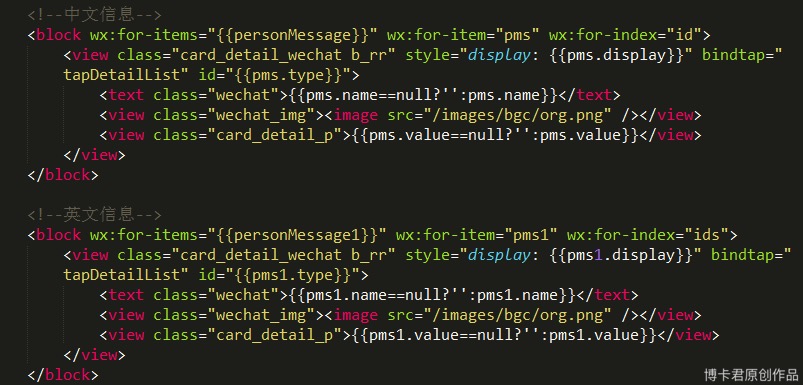
                }else{

                personMessage[k]["display"] = "block";

                }

               }

具体以 json 数据格式来处理，我们需要做的就是给它绑定 display 的值，然后我们调用即可。



微信此版本的 setData 不支持异步更新数据，故而我们在发生真实网络数据请求时一定要在后面加上 forceUpdata()，强制触发视图渲染，否则会出现很多莫名其妙的 bug。



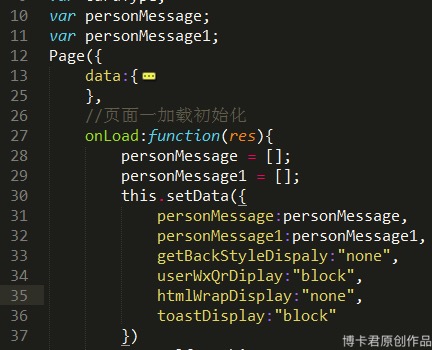
这里说明下：如是服务器真实数据。



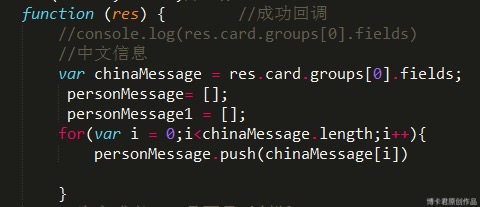
可以看到会报错，可能是 js 的执行顺序，依次往下走，此时网络数据还在请求中。



定义一个变量即可。



当然这里的数据都是 push 上来的。



以下是二维码弹出信息。



这是弹出模态框二维码信息，布好局初始化是none状态。那里需要它直接绑定数据即可：

This.setData({

//模态框名字:”显示？隐藏”

})

方法是让它显示。

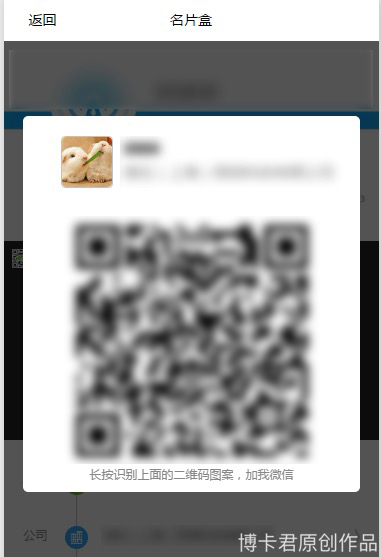


需要用他的地方调用方法即可。(支持重复调用)

详情页公司位置地图直接调用微信提供的接口实现（群里的 demo 有实现方式）。

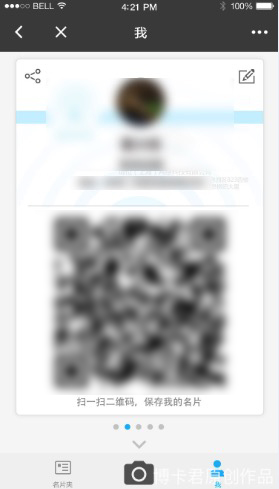
可以看下实现的效果：



  
下一章：微信（小程序）我的页面开发。

第六章：微信小程序「」「我」的页面开发

效果图与需求：



用户有多张名片，需要左右切换查看，往下切换是菜单按钮。

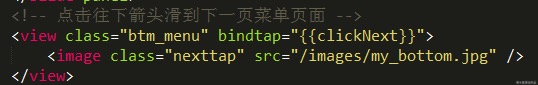
这里需求两处滑动，用到了微信提供给我们的滑动组件 swiper，并且进行了嵌套使用，第一层是名片展示与菜单按钮的上下滑动，第二层是名片展示的左右滑动（支持互相嵌套使用的，可以放心使用）。

Vertical 加上就是纵向滑动，去掉即是左右滑动。

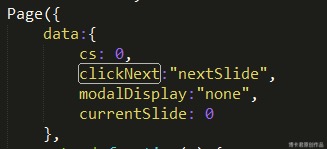
整体结构如下所示：



点击事件绑定的是数据切换方式，因为需要支持多次点击切换。



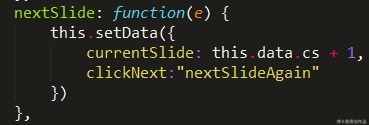
初始化数据是 nextSlide：



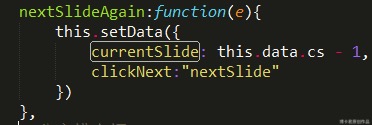
再看下 nextSlide 事件。currentSlide 是当前页面的 index，改变它即可完成切换效果，可以看上图初始化数据时设置了 cs 是 0。

https://static.oschina.net/uploads/img/201609/29140959_bnUT.jpg

因而赋值当前 data.cs+1 即可，再把绑定点击事件 clickNext 切换成 nextSlideAgain。



再看下 nextSlideAgain 事件，执行的减去 1 个索引，实现多次点击切换效果。



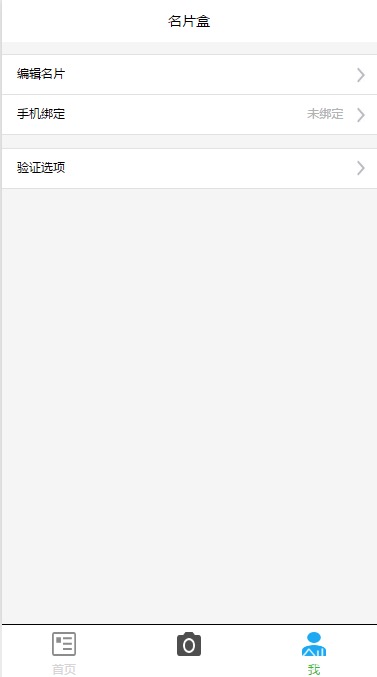
具体效果可以看到。

点击个人名片进去编辑名片页面，由于需要带参，故而使用的是 wx.navgateTo。



可以看下效果：

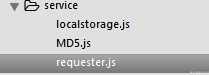




最后上点干货：

我们发出第一篇教程的时候有人就注意到这点了，怎么做真实数据交互，下面大家可以了解下。

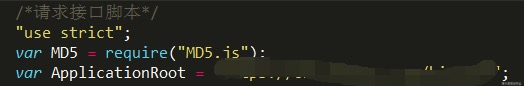
首先进去是 MD5 加密，requster 交互层。



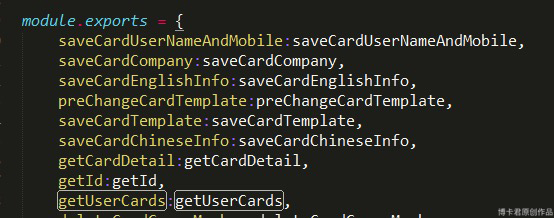
怎么引用 MD5.js？当然是模块化 require，被引用的 js 不要忘记 module.exports 出来。

下面是 requester.js 引用 MD5.js。

ApplicationRoot 是服务器地址（配置服务器时在开发设置页面查看 AppID 和 AppSecret，配置服务器域名）。



Require.js 这里 module.exports 是暴露方法出去。



这时候在全局 app.js 里面引入 require.js 映射到全局 global，这个 global 是全局的。



这时候那个页面需要那个页面就直接去接受吧，模块化是不是很好用？

https://static.oschina.net/uploads/img/201609/29141001_Ybkw.jpg

可以完整的看下与后台做数据交互的一个请求实现方式如下：

图一是 requester.js 里面的封装。

图二是需要调用数据的页面渲染。





大家有什么疑问可以多指点，会在下一章说明一些已知疑问。

应该有一些人喜欢 sublime 编辑器，有人问怎么切换高亮，在你的右下角直接切换成 html 即可。

