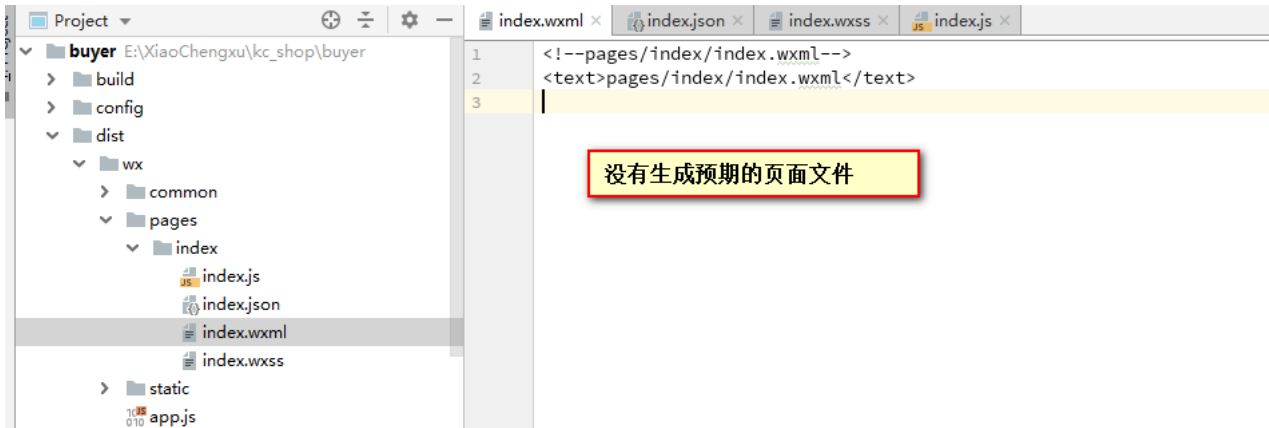


mpvue小程序-项目初始化-首页基本功能

Created 星期一 31 十二月 2018

1. 根据mpvue QuickStart文档要求，全局安装vue-cli工具，创建一个mpvue初始项目。创建过程中按照要求输入appid等项目基本信息，启用vuex（后面要用），禁用eslint（避免后续因为语法检查报错，打乱思路）。
2. 切换到新建项目所在目录下，运行npm install命令安装依赖，npm run build将mpvue源码编译为标准小程序项目。小程序项目生成目录为dist/wx。（新版mpvue好像支持了什么swan百度小程序，所以为了区分将微信小程序生成到了wx目录中）。
3. 删除新建项目中src/pages, src/components, src/utlis目录下的所有默认文件。将src/main.js文件中export default{ config: {...}}中部分内容copy到新项目中src/app.json文件中（这是不同版本mpvue引起的差异，新版本的mpvue将app.json相关配置独立出来，保持与微信小程序标准要求一致，而不是像老版本一样需要写到main.js中）。由于本次只需要实现首页部分功能，所以仅copy与index相关的内容。对于出现在app.json中tabbar部分，而本次尚未实现的页面，使用pages/index/index进行占位。
4. 将app.json文件中引用到的所有pages、静态文件等copy到新项目。
5. 尝试重新运行npm run build，发现生成的小程序页面文件dist/wx/pages/index.wxml等仅包含如下内容，并没有如预期根据项目中的mpvue源码正确生成页面，需要继续配置。



6. 由于参考项目中pages没有使用mpvue要求的xx.vue+main.js写法，而是使用了mpvue-entry插件省略了每个页面配套的main.js文件。所以为了使页面能够正常编译，我们需要配置mpvue-entry插件。（参考1: <https://www.npmjs.com/package/mpvue-entry?activeTab=readme> 参考2: <https://github.com/F-loat/mpvue-entry>）
 - a. 在package.json中增加对mpvue-entry插件的依赖（"mpvue-entry": "^2.0.0-rc.8"，要用2.0.0版本的mpvue-entry插件，不要照抄参考项目的依赖，会报错！！），执行npm install命令安装该依赖。
 - b. 参考<https://github.com/F-loat/mpvue-entry> 和 <https://www.npmjs.com/package/mpvue-entry?activeTab=readme> 对mpvue-entry插件进行配置。

```
// webpack.base.conf.js
const MpvueEntry = require('mpvue-entry') step 1

module.exports = {
  entry: MpvueEntry.getEntry('src/app.json'), step 2
  ...
  plugins: [
    new MpvueEntry(), step 3
    ...
  ]
}
```

```
// app.json - 支持 js 格式，兼容微信小程序官方配置规范
{
  "pages": [
    {
      "path": "pages/news/list", // 页面路径，同时是 vue 文件相对于 src 的路径，必填
      "config": { // 页面配置，即 page.json 的内容，可选
        "navigationBarTitleText": "文章列表",
        "enablePullDownRefresh": true
      }
    }
  ],
  "window": {}
}
```

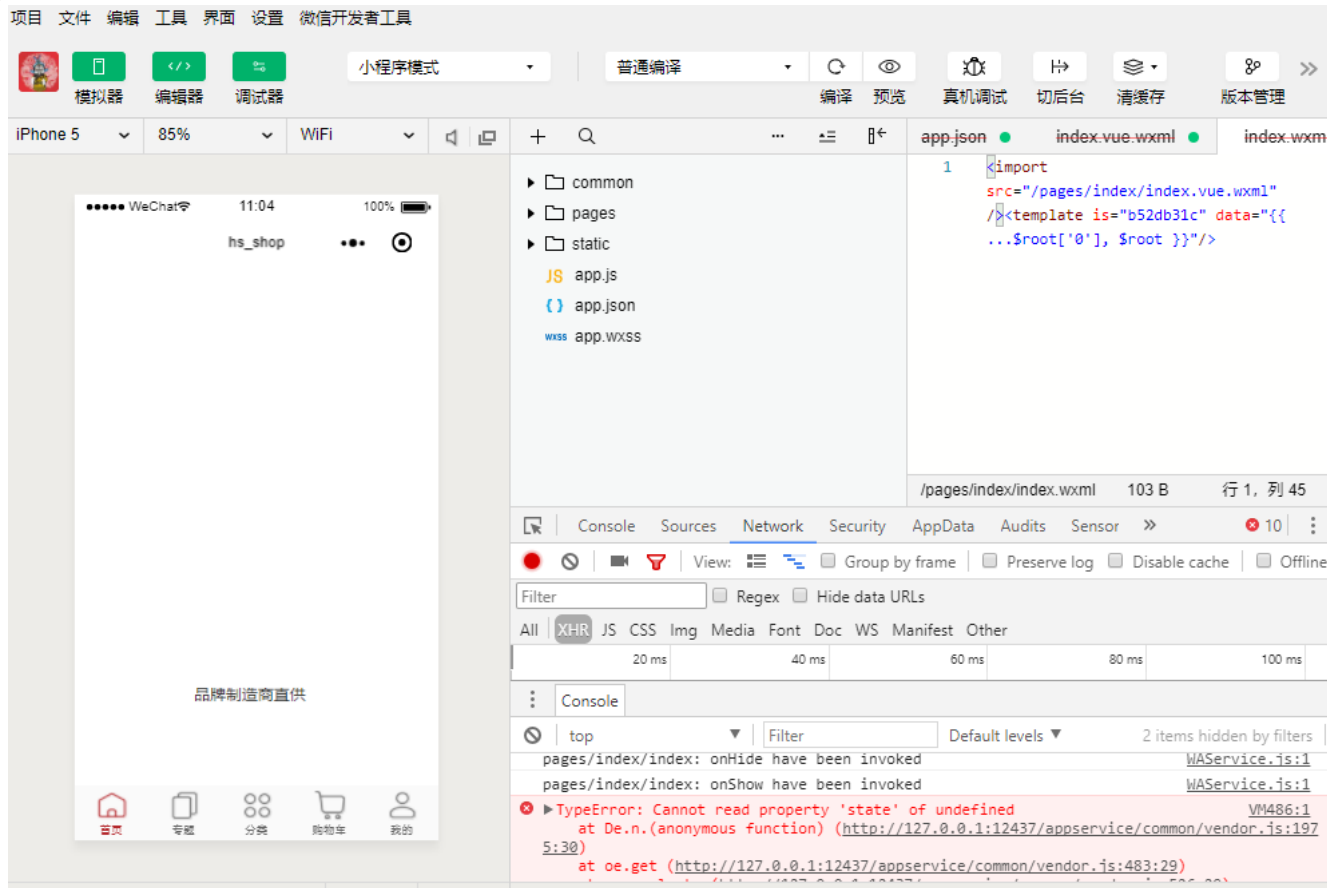
这一部分不要参考，不需要修改app.json文件

c.

```
14 // 引入mpvue-entry插件
15 const MpvueEntry = require( id: 'mpvue-entry')
16
17 function resolve (dir) {
18   return path.join(__dirname, '..', dir)
19 }
20
21 function getEntry (rootSrc) {
22   var map = {};
23   glob.sync(rootSrc + '/pages/**/*.main.js')
24   .forEach(file => {
25     var key = relative(rootSrc, file).replace('.js', '');
26     map[key] = file;
27   })
28   return map;
29 }
30
31 // 配置mpvue能够查找到各pages作为entries
32 const entry = MpvueEntry.getEntry( customPaths: 'src/app.json')
33 // 使用mpvueEntry插件后以下3行代码功能通过mpvueEntry实现，所以注释掉
34 // const appEntry = { app: resolve('./src/main.js') }
35 // const pagesEntry = getEntry(resolve('./src'), 'pages/**/*.main.js')
36 // const entry = Object.assign({}, appEntry, pagesEntry)
37
38 let baseWebpackConfig = {
39   // 如果要自定义生成的 dist 目录里面的文件路径，
40   // 可以将 entry 写成 { 'toPath': 'fromPath' } 的形式，
41   // toPath 为相对于 dist 的路径，例: index/demo，则生成的文件地址为 dist/index/demo.js
42   entry,
43   target: require( id: 'mpvue-webpack-target'),
44   output: {
```

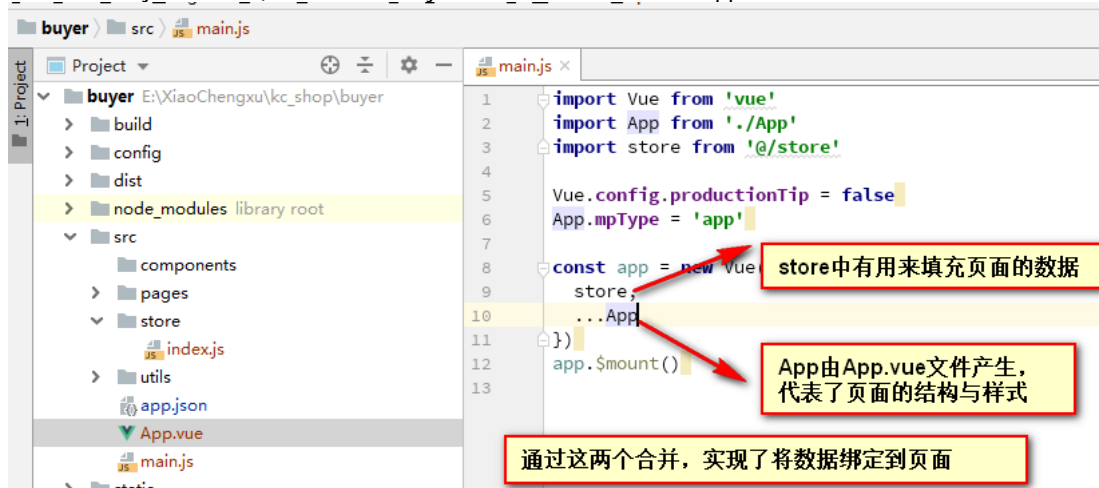
- d. mpvue-entry插件配置总结: mpvue-entry插件一定要使用"^2.0.0-rc.8"版本，旧版本多有不兼容。不需要再新建一个routes.js文件，直接从app.json文件可以解析到entries。网上文档有很多错误，参考时候一定要多思考。
- e. 至此，重新运行npm run build命令，mpvue项目中页面已经可以成功编译为标准小程序页面，只是运行过程中还有很多报错:

f.



7. 让我们重新回到pages/index/index.vue, 页面文件中多处出现的banner、channel、brands、newGoods等变量的绑定通过vuex插件实现。
vuex参考: <https://www.jianshu.com/p/a804606ad8e9> 所以接下来需要配置vuex插件。

- 初始化项目时已经选择启用vuex, 此处不用再修改package.json
- 从旧项目复制src/store文件夹及文件夹中文件到新项目对应路径
- 将src/store/index.js文件依赖或者间接依赖的src/utls/api.js request.js wx.js等copy到新项目 (将整个src/utls文件夹copy过来)
- 将request.js依赖的flyio (用来发送http请求的库, 功能相当于axios库) 添加到package.json, 重新npm install
- 修改src/main.js文件如下 (使用store/index.js文件中导出的对象对默认的App对象进行扩充, 请自行了解ES6的...操作符):



f.

g. 重新执行npm run build, 报如下错误:

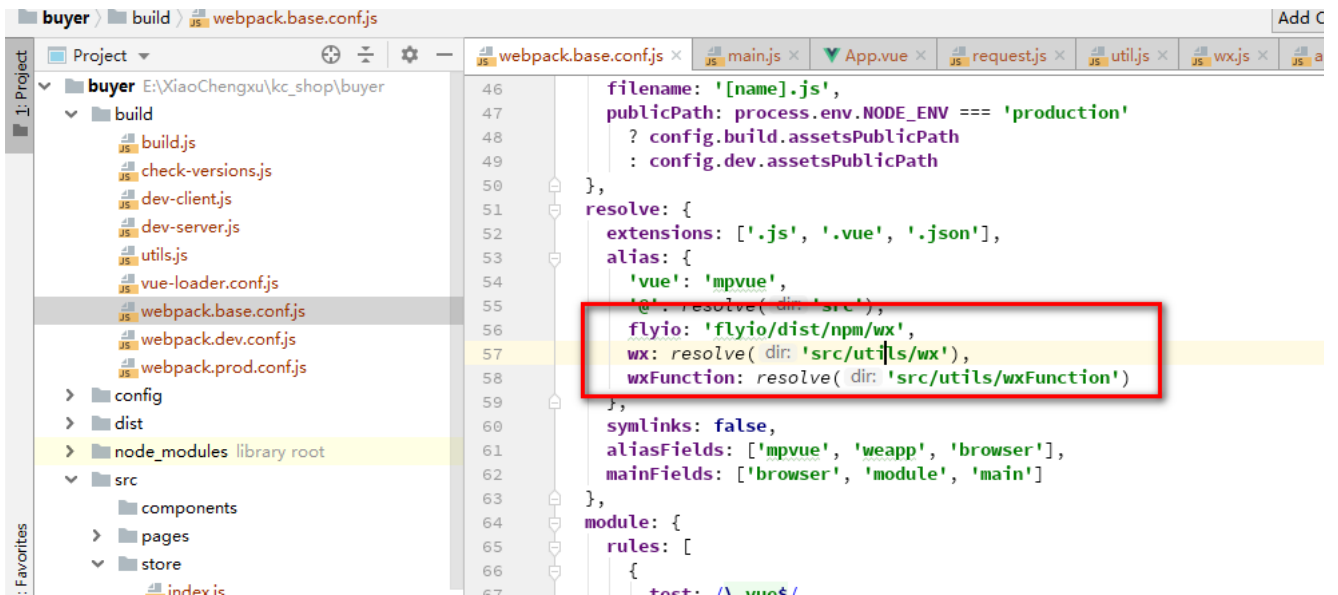
```
ERROR in ./src/utls/request.js
Module not found: Error: Can't resolve 'wx' in 'E:\XiaoChengxu\kc_shop\buyer\src\utls'
 @ ./src/utls/request.js 1:0-20
 @ ./src/utls/api.js
 @ ./src/store/index.js
 @ ./src/main.js

Build failed with errors.
```

h. npm ERR! code ELIFECYCLE

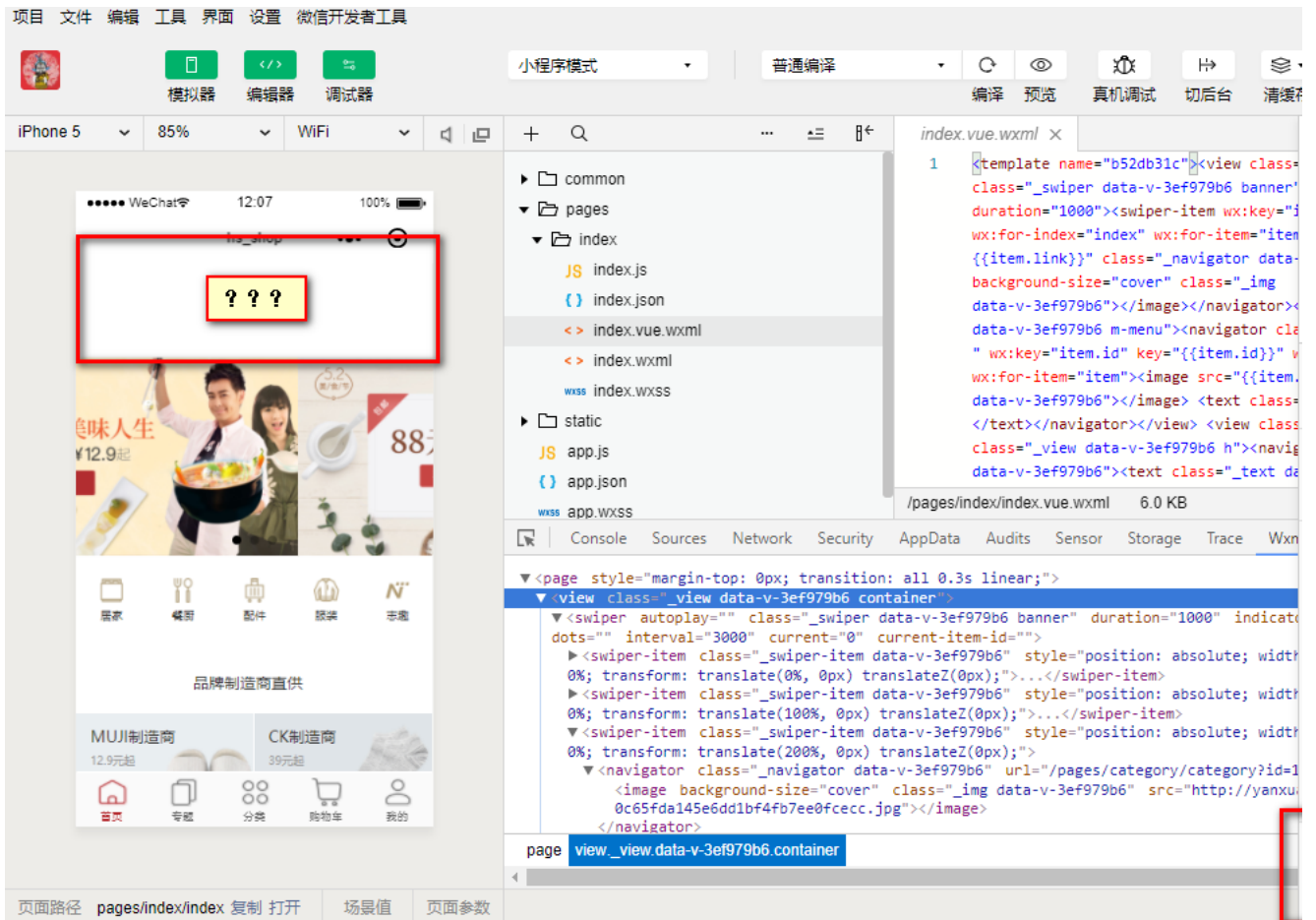
i. 在项目webpack.base.conf.js配置文件中增加如下内容:

j.



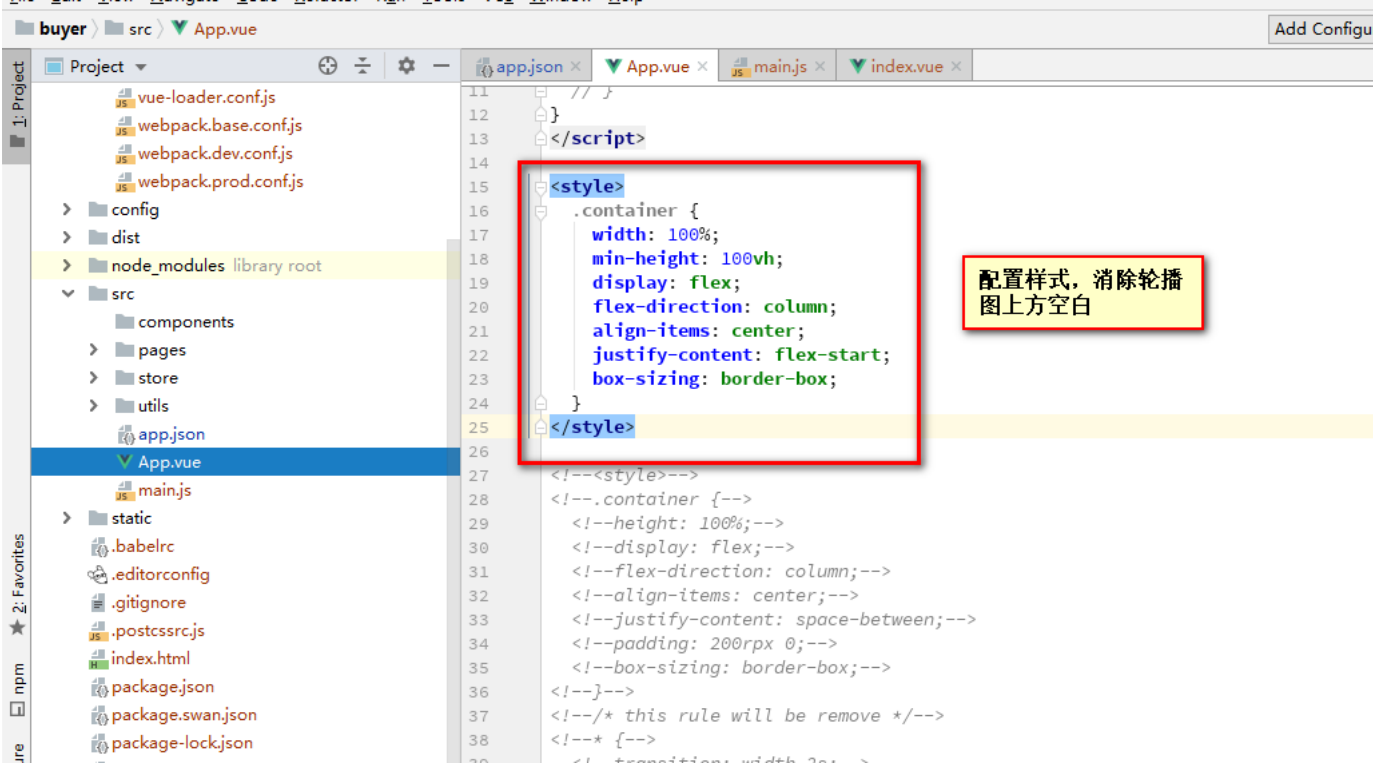
k. 重新执行npm run build命令, 启动server项目, 在微信小程序开发工具中设置允许访问非https后台, 刷新小程序, 可以看到小程序商城首页 (但是首页轮播图上方便多了一片空白区域):

l.



8. 复制参考项目src/App.vue文件中style标签内所有内容覆盖新项目style标签, 即可解决轮播图上方空白区域问题。至此小程序项目初始化及首页实现完成。

9.



10. 需要补充了解的知识：mpvue与vue的异同、vuex功能及原理、webpack打包工具原理、mpvue-entry插件功能及原理。更新后的项目代码及本文档已同步至github，请自行参考。

Attachments:

- [pasted_image.png](#) 23.5kb
- [pasted_image001.png](#) 11.7kb
- [pasted_image002.png](#) 74.2kb
- [pasted_image003.png](#) 86.3kb
- [pasted_image004.png](#) 87.7kb
- [pasted_image005.png](#) 87.7kb
- [pasted_image006.png](#) 49.7kb
- [pasted_image007.png](#) 17.9kb
- [pasted_image008.png](#) 80.8kb
- [pasted_image009.png](#) 175kb
- [pasted_image010.png](#) 247kb
- [pasted_image011.png](#) 92.2kb