2019

冉鹏峰

聚贤庄

2019-3-6

微信小程序开发指南

目录

[微信小程序开发指南 2](#_Toc2765523)

[前期准备工作 2](#_Toc2765524)

[小程序开发之前端开发 3](#_Toc2765525)

[开发基础知识储备 3](#_Toc2765526)

[调试技巧 5](#_Toc2765527)

[小程序开发之后台开发 6](#_Toc2765528)

[技术选型 6](#_Toc2765529)

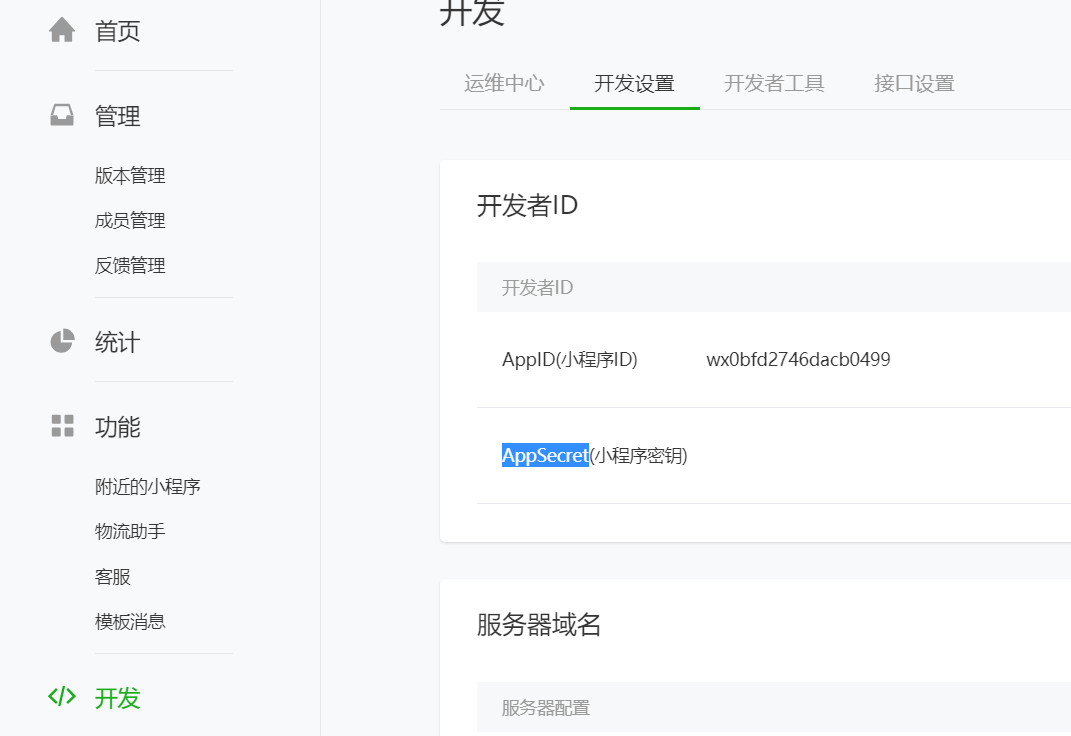
[数据交互处理 6](#_Toc2765530)

[小程序开发之数据库设计 7](#_Toc2765531)

# 微信小程序开发指南

## 前期准备工作

1. 官网注册账号：<https://mp.weixin.qq.com/>
2. 在主页》开发》开发设置中保存自己的AppID和AppSecret



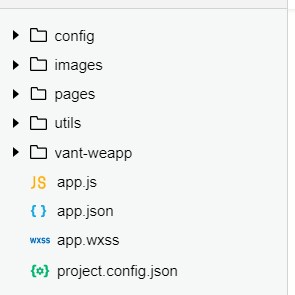
其中AppSecret只有一次明文保存机会，忘记后只有重置

1. 小程序不支持svn，初次导入项目需要在微信开发工具以外操作，提交项目亦是如此
2. 在主页》成员管理》项目成员中配置开发成员微信

## 小程序开发之前端开发

### 开发基础知识储备

#### 项目结构介绍



Config：存放后台服务器的各种URL地址

Image：存放图片

Pages：小程序中真实页面代码文件

Utils：前端工具类，可无限迭代

Vant-weapp：插件

App.js/json/wxss/wxml：小程序全局配置入口，用于保存全局信息、函数、登陆操作、页面注册、菜单配置、全局样式

Project.config.json：开发工具在本项目的配置信息，相当于eclispe的.settings的东西，保存开发者的个性设置。

#### Pages内容介绍

* 新建一个微信页面：1.鼠标右键新选择新建目录，2.在新建目录上右键新建page。

这样的好处在于：自动将新建页面信息注册到app.json中，不用人为配置；自动创建\*.js/json/wsxx/wxml文件。

* 微信中js语法遵循ES5，同时数据双向绑定，具体参考vue写法
* wxml文件中常用的数据遍历方法介绍

示例代码：

<view wx:for="{{noteList}}" wx:key="{{item.noteId}}" class='pannel'>

<text class='header'>{{item.header}}</text>

<view class='content-view'>

<text class='content'>{{item.createTime}}:</text>

<text class='content'>{{item.noteContent}}</text>

<view class='icon-block'>

<image class='icon' mode="widthFix" src='../../images/icon/readNote.png'></image>

<text class='content' bindtap='meetRecentMethod' data-id='{{item.meetId}}'>查看</text>

<image class='icon' mode="widthFix" src='../../images/icon/delNote.png'></image>

<text class='content' bindtap='bindButtonDelete' data-id='{{index}}' data-noteid='{{item.noteId}}'>删除</text>

</view>

</view>

</view>

说明：

wx:for="{{noteList}}" 中的noteList是对应page下js中data定义的数据，为空或者长度为0将不会显示任何内容

wx:key="{{item.noteId}}" 这个内容作为该数据行的唯一标识，用于页面数据动态变化时锁定数据行，具体介绍百度:wx:key作用

在微信标签中或者标签属性中直接用{{item.属性}}去显示即可

示例中的noteList数据：

{"statusCode":"200","data":[{"noteId":"e2406e87-ec61-4a89-b2a1-d37c8d9b1a15","userId":"cd95c364-797f-445e-b739-04c58b9b708d","meetId":"51e2e5b7-6d4d-4680-8af0-7cbd3c86f078","noteContent":"this is a test for Noting and Update","createTime":"2019-01-30 10:37:32","header":"测试"}]}

显示出来的效果：



* 全局变量保存

有两种方式可以实现：1. wx.getStorageSync/ setStorageSync（推荐） 2.操作app的globalData

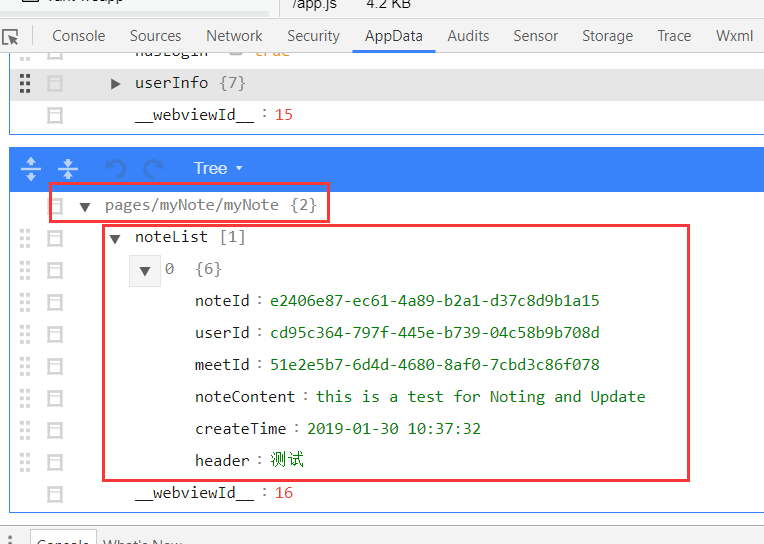
### 调试技巧

**首先去掉开发工具默认的小程序的https校验**

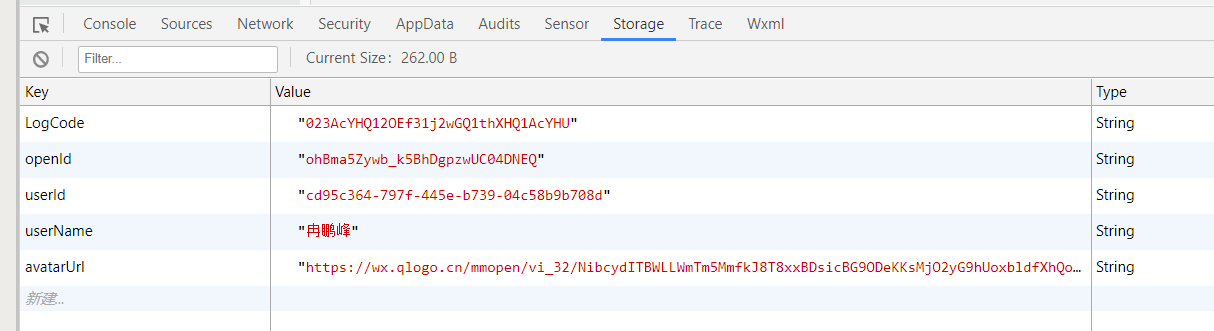


**其次善于通过调试工具的Network去判断后台数据的请求状态**

**通过appData去检查各个页面的数据:**



**通过Storage去检查全局变量**



登陆部分可以最后在做，因为登陆校验时候开发者工具的appid需要同后台数据程序的appid保持一致

## 小程序开发之后台开发

### 技术选型

后台开发的标准架构是一个提供webservice服务器即可，实现方案：springboot+REST(配置更简单，灵活)、spring+jersey\_REST(更加熟练/成熟)

### 数据交互处理

为了更好的遵循微信开发规范，在获取敏感的用户信息时，需要对小程序提交的信息经行解密。

比如获取用户的昵称、城市、头像等信息需要将用户在小程序中做点击登陆操作后获取的getuserinfo中的encryptedData（AES加密的数据暗文）和iv（解密初始向量，理解成加盐）提交到后台，后台通过session\_key+IV去解密encryptedData获取昵称、城市、头像等信息。

小程序中辨识用户只能通过openId去辨识，具体做法如下：

让用户授权登陆小程序，在wx.login方法的success回调中获取一个code，有效时间5分钟，此code建议只调用并保存一次，此时让服务端回调小程序的API

传入的参数必须包含code，反回参数获取session\_key和openId，只要code不变，session\_key和openid就一直有效，服务端获取到session\_key后将openid

返回到小程序中，此openid作为客户端的唯一标识，以后在做数据验证的时候，可以通过验证此openid的有效性来验证用户合法性。

## 小程序开发之数据库设计

由于小程序的特殊性，如登陆/注册等操作需要配合小程序的开发api共同完成，因此辨识用户的openid和解密用户信息的session\_key需要在用一张表专门维护

