# 微信统一平台使用文档

## 公众后台配置:

### 一 微信配置:

登录微信公众号后台配置服务器: <https://mp.weixin.qq.com/>





服务器的URL路径填写统一平台的地址:<http://itapl.com/weixin-web/public/callback>

这里请选择明文模式, 暂不支持加密.

在公众号中注册网页授权URL .

1.

2.

3.



网页授权域名

### 二 管理平台配置

登录安普利微信管理平台: <http://itapl.com/weixin-mgr/>

用户名: admin, 密码 admin.

添加公众号:



登录微信管理平台,添加将自己的公众号添加进去



参数说明:

**公众号名称:** 公众号的名称

**原始ID:** 注意不是公众号的账号, 在公众号的账号信息页面最下面的ID, 即创建公众号的 时候,自动生成的原始ID

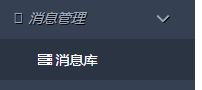
**公众号类型:** 分为服务号和订阅号

公众号appid : appid

公众号密匙: secret

**子系统入口**: 注意, 这里只有在需要微信网页授权 和 微信JSAPI使用的时候需要填写,具体的填写内容见下文.

2.1 配置消息:



可以创建三种不同的消息类型

2.2 事件管理

用户关注公众号推送消息:

消息创建完成后, 在公众号页面修改公众号信息中添加.



2.3 菜单管理



创建菜单栏目: 创建菜单栏目的时候,当菜单的响应类型是 [点击跳转URL]

地址类型[常规地址]: 跳转到这个地址对应的URL.

[获取openid]: 获取微信用户的openid, 这里需要在公众号的[子系统入口] 配 置子系统入口的URL(如何配置子系统入口,请看下文的子系统对接)

[授权地址]: 同上

创建默认菜单: 首先在[微信菜单] 栏目中,创建菜单的栏目. 接下来在微信菜单组中创建菜 单组,将菜单栏目添加到里面, 选择同步到微信, 那么默认的菜单就创建好 了

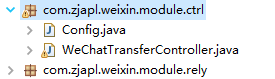
创建个性化菜单: 首先创建个性化规则,如果需要对用户进行标签分组, 那么需要在标签页面创建标签, 并且同步到微信上面,这时候就会获得微信给我们的 标签ID, 这个ID在后面和子系统的对接中需要用到.





## 子系统对接

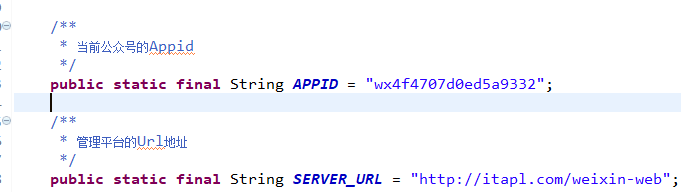
子系统需要在自己的项目中加入两个包.



只需要关心ctrl 包下面的内容即可



Config.java 是一些配置信息:



WeChatTransferController.java 是授权回调的入口,比较重要

它是一个Controller, 当平台获取到微信用户信息的时候,会重定向到这个Controller的auth方法中.



我们在上面配置公众号子系统入口的时候配置的地址就是能够访问到这个/auth目录的上层地址.

例如: <http://itapl.com/office-web/public/wechat/auth> 这是个子系统的地址,可以访问到 子系统的/auth 目录, 在配置子系入口的时候配置http://itapl.com/office-web/public/wechat/ 即可, 配置到auth的上层目录, 平台在重定向的时候会自动在该地址后面加上auth.

auth 接口在接受参数的时候, openid 就是微信用户的openid, state参数就是我们在配置菜单的时候配置的参数:



这里配置的是myinfo, 那么当用户在点击这个按钮的时候,进入这个地址, state就是myinfo.

data 只有当地址类型是授权地址的时候才有值, 是微信用户的详细信息. 是JSON 类型.

将data转为对象:

WeiXinUser user = JSON.parseObject(data, WeiXinUser.class); //将data 解析成用户数据, 如果Scope为base, data是空

## 微信页面JSAPI授权

需要在页面中打开用户相册, 打开摄像头, 分享到朋友圈等等功能的时候, 详见https://mp.weixin.qq.com/wiki JS-SDK说明文档

需要引入以下JS:

<script type="text/javascript" src="http://res.wx.qq.com/open/js/jweixin-1.2.0.js"></script>



在页面中引入上面的JS

<script type="text/javascript">

$.initJsApiConf(

['scanQRCode'],

function() {

//JSAPI 初始化成功

$("#btn").click(function(){

wx.scanQRCode({

needResult: 0, // 默认为0，扫描结果由微信处理，1则直接返回扫描结果，

scanType: ["qrCode","barCode"], // 可以指定扫二维码还是一维码，默认二者都有

success: function (res) {

var result = res.resultStr; // 当needResult 为 1 时，扫码返回的结果

}

});

});

},

function(res) {

//JSAPi 初始化失败

alert(res);

}

);