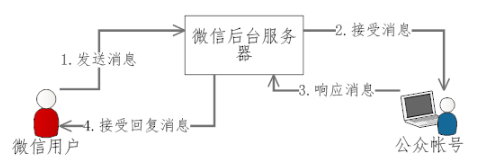
# 微信公众号通讯机制



# 微信公众号简介

微信公众号分为服务号、订阅号、企业号（现为企业微信），订阅号可以个人申请，服务号和企业号要有企业资质才可以。

**订阅号和服务器的区别**

服务号：主要偏向于服务交互（功能类似12315，114，银行，提供绑定信息，服务交互），每月可群发4条消息；服务号适用人群：媒体、企业、政府或其他组织。

订阅号：主要偏向于为用户传达资讯，（功能类似报纸杂志，为用户提供新闻信息或娱乐趣事），每天可群发1条消息；订阅号适用人群：个人、媒体、企业、政府或其他组织。

# 开发前配置：

## 注册微信公众号/测试号

### 个人订阅号

进入微信公众号注册页面<https://mp.weixin.qq.com/>，此处我选择的是个人订阅号



注册成功之后进入微信公众平台后台,然后完善微信号名称和微信号ID





微信号名称默认是新注册公众号,还需要修改微信号名称, 修改的时候需要经过微信认证,并且审核通过之后才可以使用该公众号.·

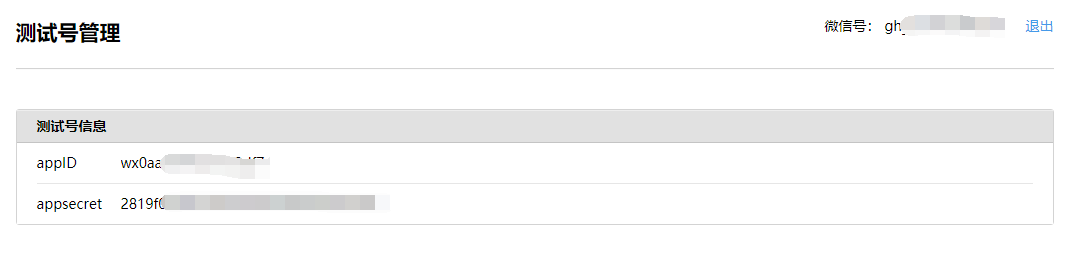
参考网址：<https://blog.csdn.net/a1786223749/article/details/80787379#1_28>

### 测试号

个人订阅号权限较低，可注册测试号（权限搞）进行测试

测试号注册地址：

<http://mp.weixin.qq.com/debug/cgi-bin/sandbox?t=sandbox/login>

用微信扫描页面中的二维码进行登录，登录成功后，就可以看到腾讯分配给我们的测试公众号的信息了，如下图所示, 接下来我们就可以搭建环境,进行开发测试了

参考网址：<https://blog.csdn.net/a1786223749/article/details/80787379#2_51>

## 搭建微信本地调试环境

注册、本地安装NATAPP外网穿透

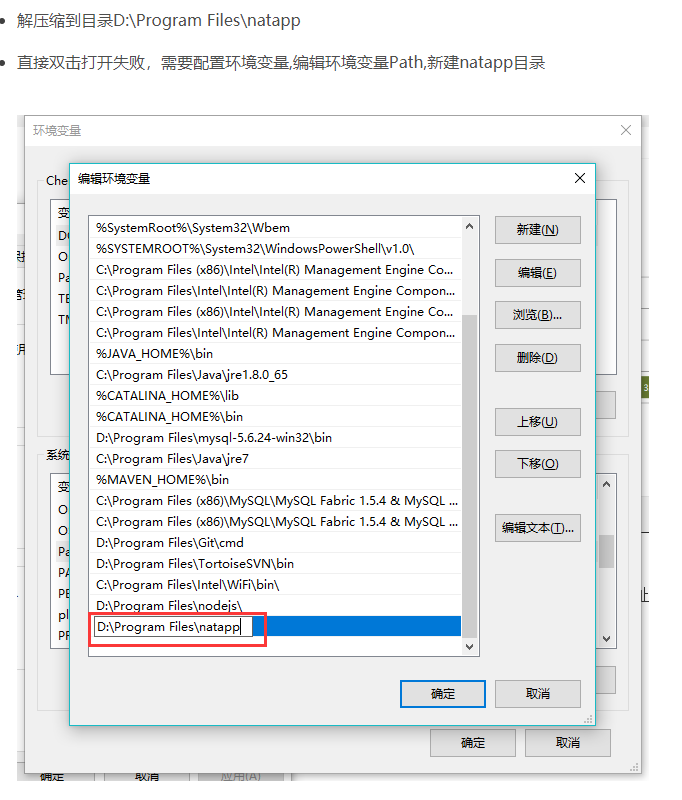
使用外网穿透，映射本地服务端口，完成通过外部http请求访问本机服务器

注册官网：<https://natapp.cn/>

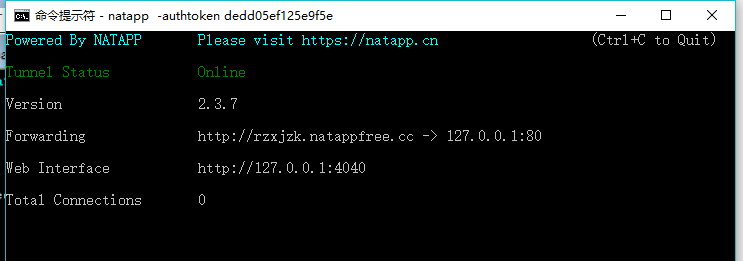
1. 下载natapp客户端



1. 解压、安装、配置环境变量



1. 注册购买免费隧道，获取authtoken
2. cmd进入natapp目录执行 natapp -authtoken yourauthtoken



此时外网用户可以直接使用http://rzxjzk.natappfree.cc这个域名访问到我内网的127.0.0.1:8080服务器了

注：8080端口为natapp配置时设置的本地端口号；

打开上方图片命令时，不可关闭，否正url可能会失效

参考网址：<https://blog.csdn.net/a1786223749/article/details/80787379#3_68>

## 安装本地redis

安装redis，用于存放AccessToken

参考<https://www.runoob.com/redis/redis-install.html>

# 开始开发（初步开发）

微信公众号申请成为开发者，进入开发者模式

## 微信公众号接入（校验签名）

接入公众号需要3个步骤，分别是：

1. 填写服务器配置
2. 验证服务器地址的有效性
3. 依据接口文档实现业务逻辑

### 填写服务器配置

**第1步中服务器配置包含服务器地址（URL）、令牌(Token) 和 消息加解密密钥(EncodingAESKey)。**

Token可由开发者可以任意填写，用作生成签名（该Token会和接口URL中包含的Token进行比对，从而验证安全性）；

EncodingAESKey由开发者手动填写或随机生成，将用作消息体加解密密钥。本例中全部以未加密的明文消息方式，不涉及此配置项

### 验证服务器地址的有效性

**第2步，验证服务器地址的有效性，当点击“提交”按钮后，微信服务器将发送一个http的get请求到刚刚填写的服务器地址，并且携带四个参数：**



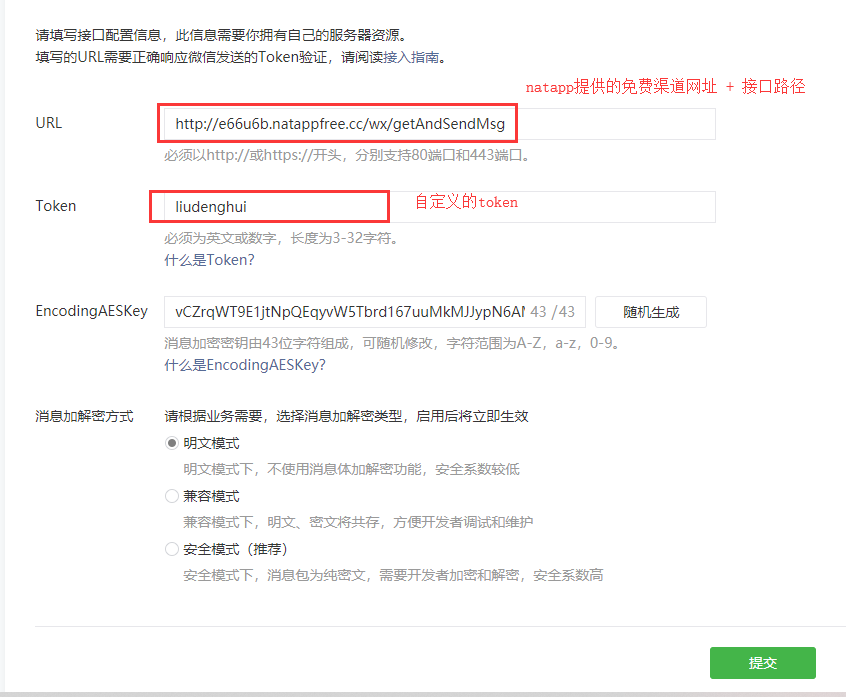
控制层中@RequestMapping，接收方式为GET

接到请求后，我们需要做如下三步，若确认此次GET请求来自微信服务器，原样返回echostr参数内容，则接入生效，否则接入失败。

　　1. 将token、timestamp、nonce三个参数进行字典序排序

　　2. 将三个参数字符串拼接成一个字符串进行sha1加密 (可逆加密解密函数)

　　3. 开发者获得加密后的字符串可与signature对比，标识该请求来源于微信



点击提交，会在本机服务器进行验证签名

**校验签名通过后，此url为公众号接受消息的地址**

注：每次更换natapp地址时，都需修改一下配置url

### 参考

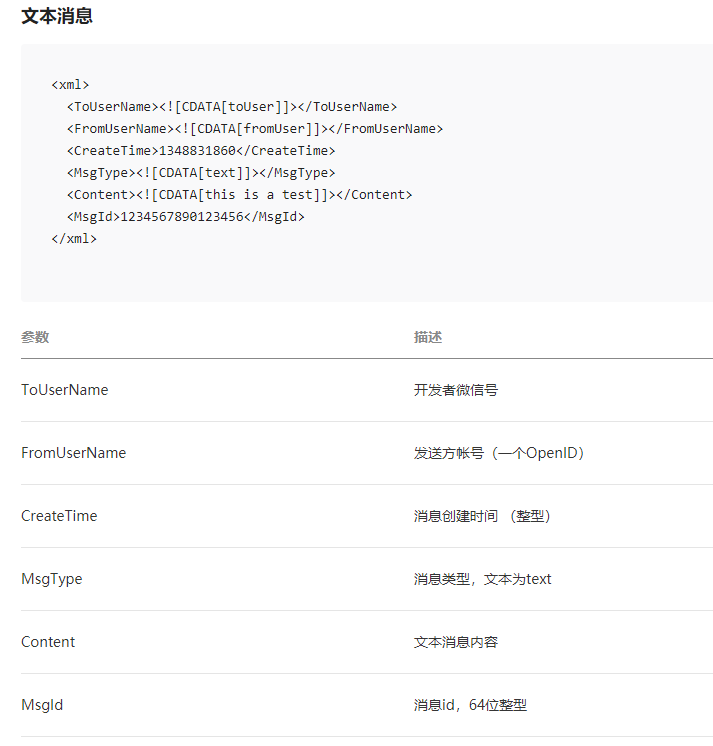
参考网址：<https://blog.csdn.net/shenbug/article/details/78748346>

<https://blog.csdn.net/a1786223749/article/details/80787379#4_127>

## 消息回复，实现与用户交互

### 消息接收的数据与回复数据结构分析

**接收文本消息**



**回复文本消息**



### 获取信息请求，将xml转为信息格式

控制层中@RequestMapping value值为上一步接入时的url路径，接收方式为POST

通过MessageUtil. xmlToMap()将接收消息的xml转换为Map格式

获取map中的信息

### 处理post请求，将信息返回到微信平台

根据用户发送的信息，进行逻辑处理，

通过MessageTextUtil. initMessage()方法，将返回的信息写入，转换发送者/接收者的位置

MessageTextUtil. messageToXml()方法，将信息转为xml返回到微信平台

### 参考

参考网址：<https://blog.csdn.net/shenbug/article/details/78749732>

微信API：

接收普通消息：

<https://developers.weixin.qq.com/doc/offiaccount/Message_Management/Receiving_standard_messages.html>

被动回复用户消息

<https://developers.weixin.qq.com/doc/offiaccount/Message_Management/Passive_user_reply_message.html>

## Access\_token的获取

### Access\_token介绍

access\_token是公众号的全局唯一接口调用凭据，公众号调用各接口时都需使用access\_toke。access\_token的有效时间时2个小时，并且调用的次数是有限的。建议在使用采用缓存处理。通过访问接口的方式获取access\_token.在之后的开发中会很常见，调用接口。

### 获取Access\_token

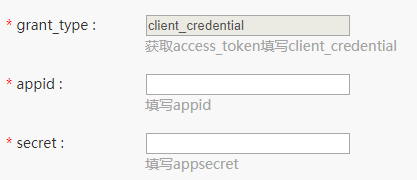
向微信发送get请求获取Access\_token

url：<https://api.weixin.qq.com/cgi-bin/token>

? grant\_type=client\_credential

&appid=APP\_ID

&secret=APP\_SECRET



先在redis中查询是否存在Access\_token，若不存在，再向微信请求获取

获取后，存入redis，并设置失效时间

### 参考

参考网址：

Access\_token获取：<https://blog.csdn.net/shenbug/article/details/78752888>

redis配置： <https://www.cnblogs.com/-flq/p/9068605.html>

## 图片、语音消息回复

### 新增**永久**图文素材

因为测试的公众号为个人未认证订阅号，无法使用临时图文素材

将需上传文件以流形式上传至微信

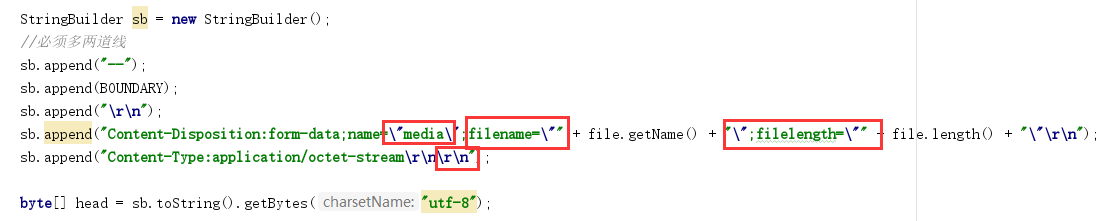
流最后需加 /r/n

还需填写media

微信接口文档提示：



代码如下：



上传结果会返回xml，携带media\_id素材文件id，发送消息时需使用次id

### 回复图片、语音信息





根据xml格式创建图片、语音消息实体类

创建工具方法（写入信息、转为xml）

发送信息：controller.WeChatMsgController. getAndSendMsg()方法中修改if..else…信息，返回不同信息

### 参考

参考网址：<https://blog.csdn.net/shenbug/article/details/78754540#commentBox>

微信API：

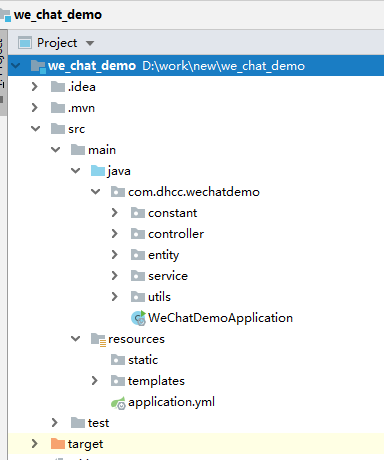
新增永久素材“

<https://developers.weixin.qq.com/doc/offiaccount/Asset_Management/Adding_Permanent_Assets.html>

被动回复用户消息

<https://developers.weixin.qq.com/doc/offiaccount/Message_Management/Passive_user_reply_message.html>

# 代码结构



**src/main/java/com.dhcc.wechatdemo**

│

├── constant – 常量、微信接口url存放地址

│

├── controller – 控制层

│

├── entity – 实体类层

│ │

│ ├─ material -- 素材相关实体类包

│ ├─ menu -- 菜单相关实体类包

│ ├─ message -- 消息相关实体类包（接收、回复、客服、群发、模板消息）

│ ├─ user -- 用户相关实体类包

│ ├─ AccessToken.java -- accessToken

│ ├─ CreateQRCode.java -- 创建二维码

│ └─ KfAccount.java -- 客服账号

│

├── service – service层

│

├── utils – 工具类

│ │

│ ├─ message -- 接收/发送消息工具类包（xml转json。json转xml）

│ ├─ EncryptionUtil.java -- 加密工具工具类（微信服务器验证时使用）

│ ├─ HttpHelper.java -- 发送HTTP请求工具类（GET、POST、上传/下载文件）

│ ├─ ImageUtil.java -- 图片处理工具类（暂未使用）

│ └─ WxErrorCodeUtil.java -- 微信错误码工具类，（根据微信返回错误码，提示错误信息）

│

└──WeChatDemoApplication – 入口文件

**src/main/resources**

│

├── static – 静态文件

│

├── templates – HTML页面（暂时测试使用）

│

└── application.yml – 配置文件

# 接口API

详细请参考微信公众号开发文档：

<https://developers.weixin.qq.com/doc/offiaccount/Getting_Started/Overview.html>

完成以下相关API接口，需先配置好微信服务器（微信公众号接入）

## 基础设置（/base）

### 获取access\_token

**接口地址**：/base/getAccessToken

**请求方式**：GET

**输入**：无

**输出**：json格式的AccessToken信息（包含access\_token值，有效时间）

**介绍：**

access\_token是公众号的全局唯一接口调用凭据，公众号调用各接口时都需使用access\_token。开发者需要进行妥善保存。

access\_token的存储至少要保留512个字符空间。

access\_token的有效期目前为2个小时，需定时刷新，重复获取将导致上次获取的access\_token失效。（因此需使用redis存放，且设置有效时间）

**流程：**

* 检查redis中是否存在access\_token
* 存在，则获取redis中数据；
* 不存在，向微信发起GET请求获取，并将数据存放在resia中，设置时效2小时
* 返回access\_token

### 获取微信callback IP地址

**接口地址**：/base/getCallbackIp

**请求方式**：GET

**输入**：无

**输出**：json格式的服务器ip地址

**介绍：**

callback IP即微信调用开发者服务器所使用的出口IP

### 获取微信API接口 IP地址

**接口地址**：/base/getApiDomainIp

**请求方式**：GET

**输入**：无

**输出**：json格式的api.weixin.qq.com的解析地址

**介绍：**

API接口IP即api.weixin.qq.com的解析地址，由开发者调用微信侧的接入IP

## 自定义菜单（/menu）

注意：

1.微信自定义菜单最多包括3个一级菜单，每个一级菜单最多包含5个二级菜单。

2.一级菜单最多4个汉字，二级菜单最多7个汉字，多出来的部分将会以“...”代替。

3.创建自定义菜单后，菜单的刷新策略是，在用户进入公众号会话页或公众号profile页时，如果发现上一次拉取菜单的请求在5分钟以前，就会拉取一下菜单，如果菜单有更新，就会刷新客户端的菜单。测试时可以尝试取消关注公众账号后再次关注，则可以看到创建后的效果

### 创建菜单

**接口地址**：/menu/createMenu

**请求方式**：POST

**输入**：无

**输出**：json（errcode、errmsg）（errcode = 0，表示创建成功）

**介绍：**

暂时在MenuController.createMenu()方法中指定自定义菜单信息

实际需传入自定义菜单的json信息，包含不同级别菜单，菜单类型、名称、对应事件

需注意创建菜单的格式问题，（不同类型，不同点击事件）

菜单按钮，可设置：点击类型、网页类型、小程序跳转、微信应用调用（相机、位置）

具体格式参考：微信开放平台：

<https://developers.weixin.qq.com/doc/offiaccount/Custom_Menus/Creating_Custom-Defined_Menu.html>

实体类：entity.menu.base包下

### 获取菜单

**接口地址**：/menu/getMenu

**请求方式**：GET

**输入**：无

**输出**：json格式的自定义菜单信息（包含个性化菜单）

**介绍：**

获取创建的个性化菜单json格式信息

### 删除菜单

**接口地址**：/menu/delMenu

**请求方式**：GET

**输入**：无

**输出**：json（errcode、errmsg）（errcode = 0，表示删除成功）

**介绍：**

删除当前使用的自定义菜单。

请注意，在个性化菜单时，调用此接口会删除默认菜单及全部个性化菜单。

### 自定义菜单的事件推送

用户点击菜单按钮，微信会发送给服务器，创建菜单时配置的信息，服务器可根据事件信息，进行反馈给用户消息（文字、图文、图片、视频等）

详见消息管理 - 接收消息

### 个性化菜单

**接口地址**：未配置

**介绍：**

开发者可以通过该接口，让公众号的不同用户群体看到不一样的自定义菜单。该接口开放给已认证订阅号和已认证服务号。

和自定义菜单相比，个性化菜单，加入了额外的配置项（指定用户群体）

具体格式参考：微信开放平台：

<https://developers.weixin.qq.com/doc/offiaccount/Custom_Menus/Personalized_menu_interface.html>

实体类：entity.menu.conditional包下

## 消息管理（/message）

当普通微信用户向公众账号发消息时，微信服务器将POST消息的XML数据包到开发者填写的URL上。

### 服务器配置

**接口地址**：/message/getAndSendMsg（GET请求）

**请求方式**：GET

**输入**：HttpServletRequest 读取request中的信息

**输出**：返会微信验证服务器传来的随机字符串echostr

**介绍：**

开发者自定义配置服务器url、token，微信根据url发送验证请求

对传来的时间戳、随机数、开发者定义的TOKEN进行排序、sha1加密处理，加密后与微信传来的加密签名signature对比，一致表示服务器验证成功，否正验证失败。

### 接收消息并返回

**接口地址**：/message/getAndSendMsg（POST请求） （与服务器配置接口地址一致）

**请求方式**：POST

**输入**：HttpServletRequest 读取request中的信息

**输出**：json（errcode、errmsg）（errcode = 0，表示删除成功）

**介绍：**

接收用户发送的消息（文字、图片、位置、点击菜单按钮等）

微信会将用户发送的详细封装成xml格式，存放到request中，开发者需进行解析xml为map，取出相关数据。

根据信息类型（text、image、event等）、具体信息（content、EventKey等），处理逻辑，返回给用户相关信息，返回的信息格式应遵循微信规定，将信息转为xml格式返回

实体类：entity.message.request接收的消息类型

entity.message.response发送的消息类型

工具类：

utils.message.RequestMessageUtil.java 将微信发来的xml转为map格式，根据不同类型进行取map中的值

utils.message.response包 将逻辑处理完后，要返回的信息转换成xml格式

### 群发消息

#### 根据标签群发

**接口地址**：/message/sendMassByTag

**请求方式**：POST

**输入**：int tagId 标签id

**输出**：json（errcode、errmsg、msg\_id消息id、msg\_data\_id）（errcode = 0，表示删除成功）

**介绍：**

根据标签群发分为两大块：

群发用户（若选定标签id，则只群发标签id对应的用户；否则群发所有用户）

群发信息（包括图文、视频、语音、文字等，具体所需信息，参考微信开放平台）

暂测试成功群发文本（群发图文测试公众号无此权限，但可进行预览群发图文，详见下方 **群发预览API**）

具体格式参考：微信开放平台：

<https://developers.weixin.qq.com/doc/offiaccount/Message_Management/Batch_Sends_and_Originality_Checks.html>

#### 根据OpenId群发【仅认证服务号可用】

**接口地址**：/message/sendMassByOpenId

**请求方式**：POST

**输入**：int openIds 用户openId

**输出**：json（errcode、errmsg、msg\_id消息id、msg\_data\_id）（errcode = 0，表示删除成功）

**介绍：**

与根据标签群发类似，根据用户ids群发分为两大块：

群发用户（用户openId的列表）

群发信息（包括图文、视频、语音、文字等，具体所需信息，参考微信开放平台）

#### 删除群发

**接口地址**：/message/delMass

**请求方式**：POST

**输入**：int msgId 群发的消息id

int[] articleIdx 要删除的文章在图文消息中的位置

**输出**：json（errcode、errmsg）（errcode = 0，表示删除成功）

**介绍：**

根据群发的消息id，删除群发消息，

msgid：必须包含

articleIdx：可不填，不填或填0，表示删除本图文消息的全部文章（仅在删除图文消息时使用）

#### 群发预览

**接口地址**：/message/previewMass

**请求方式**：POST

**输入**：无

**输出**：json（errcode、errmsg）（errcode = 0，表示删除成功）

**介绍：**

指定发送预览消息的用户openid，及指定群发消息（类型、信息）

#### 查询群发消息发送状态

**接口地址**：/message/getMassState

**请求方式**：POST

**输入**：int msgid 要查询群发消息发送状态的消息id

**输出**：json（msg\_id消息id、msg\_status消息状态）

**介绍：**

根据消息id，查询出此群发消息的发送状态。

#### 获取群发速度

**接口地址**：/message/getMassSpeed

**请求方式**：POST

**输入**：无

**输出**：json（speed速度级别、realspeed具体速度，单位：万/分钟）

**介绍：**

获取当前群发的速度级别，及具体速度信息

#### 设置群发速度

**接口地址**：/message/setMassSpeed

**请求方式**：POST

**输入**：int speed 速度级别（0-4之间）

**输出**：json（errcode、errmsg）

**介绍：**

设定群发的速度

### 模板消息

#### 设置公众号所属行业

**接口地址**：/message/setIndustry

**请求方式**：POST

**输入**：int industryId1行业编号1

int industryId2行业编号2

**输出**：json（errcode、errmsg）

**介绍：**

设定公众号所属的行业，根据行业获取相关模板

公众号只可绑定两个行业，每月仅可修改行业1次

#### 获取公众号所属行业信息

**接口地址**：/message/getIndustry

**请求方式**：POST

**输入**：无

**输出**：json（行业信息）

**介绍：**

获取设定的公众号所属行业信息，

#### 获取模板ID

**接口地址**：/message/getIndustry

**请求方式**：POST

**输入**：int template\_id\_short 模板库中的模板id

**输出**：json（获取成功/失败信息，成功返回模板template\_id消息id）

**介绍：**

根据设定的行业，选择行业相关模板库中的模板id，获取使用

（暂未测试、测试公众号需自定义模板）

#### 获取模板列表

**接口地址**：/message/getAllPrivateTemplate

**请求方式**：POST

**输入**：无

**输出**：json（模板列表信息）

**介绍：**

获取当前公众号可用的模板列表，

#### 删除模板

**接口地址**：/message/delPrivateTemplate

**请求方式**：POST

**输入**：int templateId 获取模板id时，获得的模板id

**输出**：json（删除结果）

**介绍：**

删除指定的当前可用的模板id，

#### 发送模板消息

**接口地址**：/message/sendTemplate

**请求方式**：POST

**输入**：int userOpenId 需要发送的userOpenId

**输出**：json（发送结果）

**介绍：**

发送给用户模板消息，需指定发送用户的userOpenId，模板id，具体消息信息（需按照模板消息内容设定好）

可设置点击此消息跳转url，或者跳转小程序页面

目前测试号添加了一个模板消息，后台直接写死模板id，消息信息

## 客服管理（/kf）

**注：由于测试公众号/个人未认证订阅号不存在客服权限，暂时无法测试**

### 客服账号管理

**注：必须先在公众平台官网为公众号设置微信号后才能使用该能力。**

#### 添加客服账号

**接口地址**：/kf/addKfAccount

**请求方式**：POST

**输入**：应输入客服信息（kf\_account客服账号，nicjname昵称，password密码）

（单客服时可不设置密码）

**输出**：json结果（errcode、errmsg）（errcode = 0，表示删除成功）

**介绍：**

开发者进行添加客服，须先在公众平台官网为公众号设置微信号

kf\_account格式为：账号前缀@公众号微信号

#### 修改客服账号

**接口地址**：/kf/ updateKfAccount

**请求方式**：POST

**输入**：应输入客服信息（kf\_account客服账号，nicjname昵称，password密码）

（单客服时可不设置密码）

**输出**：json结果（errcode、errmsg）（errcode = 0，表示删除成功）

**介绍：**

开发者进行修改客服账号

#### 删除客服账号

**接口地址**：/kf/ delKfAccount

**请求方式**：POST

**输入**：应输入客服信息（kf\_account客服账号，nicjname昵称，password密码）

（单客服时可不设置密码）

**输出**：json结果（errcode、errmsg）（errcode = 0，表示删除成功）

**介绍：**

开发者进行删除客服账号

#### 设置客服账号头像

**接口地址**：/kf/ updateKfHeadImg

**请求方式**：POST

**输入**：kf\_account 要设置的客服账号；file 图片文件

**输出**：json结果（errcode、errmsg）（errcode = 0，表示删除成功）

**介绍：**

为指定的客服账号上传图片文件，作为客服头像。

#### 获取所有客服账号

**接口地址**：/kf/ getKfList

**请求方式**：GET

**输入**：无

**输出**：json结果（所有客服账号信息）

**介绍：**

为指定的客服账号上传图片文件，作为客服头像。

### 客服发消息

**接口地址**：/kf/ sendKfMsg

**请求方式**：POST

**输入**：touser要发给的用户，msgtype消息类型，具体消息信息

**输出**：json发送结果

**介绍：**

客服为指定用户发送客服消息

设定消息类型（文字、图片、语音、图文、卡券、菜单消息等），具体消息信息（根据微信公众平台规定设置）。

发送文本消息时，可插入小程序文字链，进行小程序跳转。

### 客服输入状态

**接口地址**：/kf/ typingKfMsg

**请求方式**：POST

**输入**：touser要发给的用户，command输入状态

**输出**：json发送结果

**介绍：**

此接口需要客服消息接口权限。

如果不满足发送客服消息的触发条件，则无法下发输入状态。

下发输入状态，需要客服之前30秒内跟用户有过消息交互。

在输入状态中（持续15s），不可重复下发输入态。

在输入状态中，如果向用户下发消息，会同时取消输入状态。

输入状态command："Typing"：对用户下发“正在输入"状态 "CancelTyping"：取消对用户的”正在输入"状态

## 素材管理（/ material）

请注意：

1、最近更新：永久图片素材新增后，将带有URL返回给开发者，开发者可以在腾讯系域名内使用（腾讯系域名外使用，图片将被屏蔽）。

2、公众号的素材库保存总数量有上限：图文消息素材、图片素材上限为100000，其他类型为1000。

3、素材的格式大小等要求与公众平台官网一致：

图片（image）: 2M，支持bmp/png/jpeg/jpg/gif格式

语音（voice）：2M，播放长度不超过60s，mp3/wma/wav/amr格式

视频（video）：10MB，支持MP4格式

缩略图（thumb）：64KB，支持JPG格式

4、图文消息的具体内容中，微信后台将过滤外部的图片链接，图片url需通过"上传图文消息内的图片获取URL"接口上传图片获取。

5、"上传图文消息内的图片获取URL"接口所上传的图片，不占用公众号的素材库中图片数量的100000个的限制，图片仅支持jpg/png格式，大小必须在1MB以下。

6、图文消息支持正文中插入自己帐号和其他公众号已群发文章链接的能力。

### 新增永久图文素材

**接口地址**：/material/addNews

**请求方式**：POST

**输入**：无

**输出**：json新增结果（成功则包含图文素材id）

**介绍：**

调用此接口上传图文素材，后台暂时使用死数据；

一篇图文素材可包含多条文章信息，文章需指定标题title，封面缩略图thumb\_media\_id，作者author，摘要digest，content内容，原文地址等信息，可指定是否开启评论，是否显示封面等信息。

图文素材的封面缩略图必须是永久素材，文章内容中的图片可通过**上传图文消息内的图片获取URL**接口进行上传。

具体请参考

<https://developers.weixin.qq.com/doc/offiaccount/Asset_Management/Adding_Permanent_Assets.html>

### 上传图文消息内的图片获取URL（未编写）

**接口地址**：/material/upoadImg

**请求方式**：POST

**输入**：file图片文件

**输出**：json上传结果（成功则返回图片url）

**介绍：**

调用此接口用于上传图文消息中的图片，此图片不占用公众号素材库中图片数量100000个的限制。

图片仅支持jpg/png格式，大小必须在1MB以下。

### 新增永久素材

**接口地址**：/material/ uploadMaterial

**请求方式**：POST

**输入**：type素材类型（暂时根据素材类型上传）

**输出**：json上传结果（成功则返回新增的素材id）

**介绍：**

调用此接口用于新增永久素材（图片/缩略图、语音、视频）

暂时根据传来的素材类型进行上传，MaterialController中指定固定文件，固定文件信息。

注：上传视频时，需额外添加表单：视频标题、视频描述。

### 获取永久素材

**接口地址**：/material/getMaterial

**请求方式**：POST

**输入**：mediaId 素材id，type素材类型

**输出**：指定位置生成文件、或返回视频json信息

**介绍：**

调用此接口用于获取永久素材（图片/缩略图、语音、视频、图文）

获取类型为图片、语音时。会进行下载文件，后台暂时指定了下载路径

获取类型为视频时，返回视频信息json，包含视频下载url地址

获取类型为图文时，返回图文json信息

### 删除永久素材

**接口地址**：/material/delMaterial

**请求方式**：POST

**输入**：mediaId 素材id

**输出**：删除结果json

**介绍：**

调用此接口用于删除永久素材（图片/缩略图、语音、视频、图文）

指定素材id进行删除

### 修改永久图文素材（未编写）

**接口地址**：/material/updateNews

**请求方式**：POST

**输入**：mediaId 要修改的图文素材id

**输出**：删除结果json

**介绍：**

调用此接口用于删除永久素材（图片/缩略图、语音、视频、图文）

指定素材id进行删除

### 获取素材总数

**接口地址**：/material/getMaterialCount

**请求方式**：POST

**输入**：无

**输出**：结果json（包含图文、图片、视频、语音素材的数量）

**介绍：**

调用此接口用于获取素材总数（图片/缩略图、语音、视频、图文）

json： "voice\_count":语音素材数量,

"video\_count":视频素材数量

"image\_count":图片素材数量

"news\_count":图文素材数量

### 获取素材列表

**接口地址**：/material/getBatchgetMaterial

**请求方式**：POST

**输入**：type素材类型

**输出**：素材列表json

**介绍：**

调用此接口用于获取素材列表（图片/缩略图、语音、视频、图文）

根据素材类型，获取当前永久素材的list信息

## 用户管理（/ user）

### 用户标签管理

#### 创建标签

**接口地址**：user/createTag

**请求方式**：POST

**输入**：tagName要创建的标签名称

**输出**：新增的标签json信息

**介绍：**

调用此接口用于创建标签，输入标签名进行请求，返回新建的标签信息（包含标签id，名称）

#### 获取已创建标签

**接口地址**：user/ getTag

**请求方式**：POST

**输入**：无

**输出**：json格式，已创建的标签列表信息

**介绍：**

调用此接口用于获取已创建标签列表信息

#### 编辑标签

**接口地址**：user/ updateTag

**请求方式**：POST

**输入**：tagId标签id。tagName新的标签名

**输出**：json格式，修改结果

**介绍：**

调用此接口用于修改指定标签

#### 删除标签

**接口地址**：user/ deleteTag

**请求方式**：POST

**输入**：tagId要删除的标签id

**输出**：json格式，删除结果

**介绍：**

调用此接口用于删除指定标签

#### 获取标签下的用户列表

**接口地址**：user/ getUserByTag

**请求方式**：POST

**输入**：tagId标签id ，nextOpenid开始获取的用户id

**输出**：json格式，用户列表结果

**介绍：**

调用此接口用于获取指定标签下的用户列表

nextOpenid第一个拉取的OPENID，不填默认从头开始拉取（当用户过多时，可使用此进行分页查询）

#### 批量为用户打标签

**接口地址**：user/ batchTagging

**请求方式**：POST

**输入**：tagId标签id ，openidList用户列表（暂为string[]数组类型）

**输出**：json格式，打标签结果

**介绍：**

调用此接口用于批量为指定用户打指定标签

#### 批量为用户取消标签

**接口地址**：user/ batchUnTagging

**请求方式**：POST

**输入**：tagId标签id ，openidList用户列表（暂为string[]数组类型）

**输出**：json格式，取消标签结果

**介绍：**

调用此接口用于批量为指定用户取消指定标签

#### 获取用户身上的标签列表

**接口地址**：user/ getIdList

**请求方式**：POST

**输入**：openId用户id

**输出**：json格式，标签列表信息

**介绍：**

调用此接口用于获取指定用户下的标签列表

### 设置用户备注名（未编写）

**注：此接口仅可在认证服务号使用**

**接口地址**：user/ updateUserRemark

**请求方式**：POST

**输入**：openId用户id ，remark备注名

**输出**：json格式，设置结果

**介绍：**

调用此接口用于获取指定用户设置备注名称

### 获取用户基本信息（unionid机制）

**接口地址**：user/ getUserInfo

**请求方式**：POST

**输入**：openId用户id

**输出**：json格式，用户信息

**介绍：**

调用此接口用于获取指定用户的用户信息

获取的用户信息包含：

是否订阅公众号、openid、昵称、性别、所在国家/省份/城市、使用语言、头像、关注公众号时间、unionid、用户备注、所在分组、标签列表、关注渠道来源等

### 批量获取用户基本信息

**接口地址**：user/ batchGetUserInfo

**请求方式**：POST

**输入**：openIds用户id列表（string[]格式）

**输出**：json格式，用户信息列表

**介绍：**

调用此接口用于批量获取指定用户的用户信息

### 获取（关注）用户列表

**接口地址**：user/ getUserList

**请求方式**：POST

**输入**：nextOpenid起始获取id（第一个拉取的OPENID，不填默认从头开始拉取）

**输出**：json格式，用户列表

**介绍：**

调用此接口用于获取公众号关注者的列表信息

包含数量、用户openid列表

### 黑名单管理

#### **获取黑名单列表**

**接口地址**：user/ getBlacklist

**请求方式**：POST

**输入**：beginOpenId起始获取id（第一个拉取的OPENID，不填默认从头开始拉取）

**输出**：json格式，黑名单用户列表

**介绍：**

调用此接口用于获取公众号的黑名单列表

返回格式和**获取用户列表**相同

#### **拉黑用户**

**接口地址**：user/ batchBlacklist

**请求方式**：POST

**输入**：openIdList用户openid数组（可拉黑多个）

**输出**：json格式，拉黑结果

**介绍：**

调用此接口用于批量拉黑用户

#### 取消**拉黑用户**

**接口地址**：user/ batchUnBlacklist

**请求方式**：POST

**输入**：openIdList用户openid数组（可取消拉黑多个）

**输出**：json格式，取消拉黑结果

**介绍：**

调用此接口用于批量取消拉黑用户

## 工具相关（/tool）

### 二维码

#### 生成带参数二维码

**接口地址**：/tool/createQRCode

**请求方式**：POST

**输入**：无

**输出**：json格式，生成结果，包含ticket（用于获取二维码）

**介绍：**

调用此接口用于生成带参数的二维码

后台暂时定死参数信息

需配置信息包括：

有效时间（若设置，指定该二维码为临时二维码，最大不超过2592000（即30天），若不填，则默认有效期为30秒）

二维码类型：临时/永久、整型参数/字符串参数

二维码详细信息

场景值（整形/字符串两种）

具体参考微信开发文档：

<https://developers.weixin.qq.com/doc/offiaccount/Account_Management/Generating_a_Parametric_QR_Code.html>

#### 获取二维码

**注：此请求不需要access\_token**

**接口地址**：/tool/ showQRCode

**请求方式**：POST

**输入**：ticket二维码的ticket

fileName指定的文件名

**输出**：无（直接生成文件）

**介绍：**

调用此接口用于获取二维码。用过ticket获取二维码信息，设定指定二维码图片名称（若不设置，下载会生成null文件（名称+后缀））

后台默认设定了生成文件的地址

### 长链接转短链接

**接口地址**：/tool/ shortUrl

**请求方式**：POST

**输入**：longUrl长链接url

**输出**：json格式，转换结果，包含短链接url

**介绍：**

调用此接口用于将外部长链接转为微信短链接

# 遇到问题

## 微信接收/返回信息问题

微信接收/返回xml中首字母为大写，



因此实体类中不可使用驼峰原则，应全为大写

