

# Qiwu-OpenAPI-Demo

- [Qiwu-OpenAPI-Demo](#)
  - [一、接口综合说明](#)
    - [系统接口设计](#)
    - [接口公共参数](#)
      - [公共请求头参数](#)
      - [公共响应参数](#)
      - [公共响应示例](#)
      - [公共 Retcode 与对应 Message](#)
    - [请求参数加密方法](#)
      - [QueryString 加密方法](#)
      - [QueryString 加密示例](#)
      - [Body 加密方法](#)
      - [Body 加密示例](#)
    - [接口响应 Payload 解密方法](#)
  - [二、调用api接口的 Api Access Token](#)
    - [Api Token 获取接口](#)
      - [获取 Api Token 接口请求参数](#)
        - [获取 Api Token 接口请求头参数](#)
        - [接口签名生成方法\(sign\)](#)
      - [获取 Api Token 接口请求示例](#)
      - [获取 Api Token 接口响应参数](#)
      - [获取 Api Token 接口响应示例](#)
    - [刷新 Api Token 接口](#)
      - [刷新 Api Token 接口请求参数](#)
        - [刷新 Api Token 接口请求头参数](#)
      - [刷新 Api Token 接口请求示例](#)
      - [刷新 Api Token 接口响应参数](#)
      - [刷新 Api Token 接口响应示例](#)
    - [注销 Api Token 接口](#)
      - [注销 Api Token 接口请求参数](#)
        - [注销 Api Token 接口请求头参数](#)
      - [注销 Api Token 接口请求示例](#)
      - [注销 Api Token 接口响应参数](#)
      - [注销 Api Token 接口响应示例](#)

- 三、账号模块
  - 获取验证码接口
    - 获取验证码接口请求参数
    - 获取验证码接口请求示例
    - 获取验证码接口响应参数
    - 获取验证码接口响应示例
  - 登录/注册接口
    - 登录/注册接口请求参数
    - 登录/注册接口请求示例
    - 登录/注册接口响应参数
    - 登录/注册接口响应示例
  - 刷新 Token 接口
    - 刷新 Token 接口请求参数
    - 刷新 Token 接口请求示例
    - 刷新 Token 接口响应参数
    - 刷新 Token 接口响应示例
  - 登出接口
    - 登出接口请求参数
    - 登出接口请求示例
    - 登出接口响应参数
    - 登出接口响应示例
  - 获取账户信息接口
    - 获取账户信息接口请求参数
    - 获取账户信息接口请求示例
    - 获取账户信息接口响应参数
    - 获取账户信息接口响应示例
  - 新增子账号接口（不需要对接齐悟账号的客户）
    - 新增子账号接口请求参数
    - 新增子账号接口请求示例
    - 新增子账号接口响应参数
    - 新增子账号接口响应示例

## 一、接口综合说明

### 系统接口设计

1. 各应用子系统对外接口统一遵循RESTful的设计风格，遵循HTTP请求的标准。
2. 数据传输格式统一遵循JSON标准，并进行AES加密。

3. **POST请求** 请设置 Content-Type: application/json

## 接口公共参数

### 公共请求头参数

参数名	类型	描述	必须
Timestamp	Long	时间戳	Y
Access-Token	String	商户Api accessToken (通过 二-1 api授权接口获得)	Y
Authorization	String	用户accessToken (通过 三-2 用户登录接口获得) 或子账号ID	Y

注:以上公共参数，均放于请求头，除token获取、刷新接口外

### 公共响应参数

参数名	类型	描述	必须
status	Integer	响应状态	Y
retcode	Integer	请求成功标志	Y
message	String	响应信息	Y
timestamp	Long	时间戳	Y
payload	String	已加密返回数据，可能没有或为空	N

### 公共响应示例

```
{
  "status":200,
  "retcode":0,
  "message":"SUCCESS",
  "timestamp":12345,
  "payload":"CRU0t1IaRvYeYr4qMT8Td1TABGH3B/aguRJyRdec0i0BJZFm5sx4gAURU5xnbA7wBGZAxHhvBiD7B3h7GVFUI9z
}
```

### 公共 Retcode 与对应 Message

retcode	message
---------	---------

retcode	message
0	SUCCESS
160001	账号不存在，请重新确认或联系客服人员。
160002	token错误，请重新确认或联系客服人员。
160003	refreshToken有误，请重新确认或联系客服人员。
160004	您的token已过期，请尝试重新获取。
160005	您的账号已被锁定，如有疑问，请联系客服。
160101	必要参数缺失，请重新确认。
160102	参数无效，请重新确认。
160103	时间戳无效，请重新确认。
160104	签名无效，请重新确认。

## 请求参数加密方法

### QueryString 加密方法

1. 将参数组合成“参数=参数值”的格式，并且把这些参数用&字符连接起来，生成待加密字符串。
2. 商户使用分配的 aesKey 和 aesIv，通过AES算法对1得到的字符串加密，得到已加密字符串。
3. 将2中结果作为value，“q”作为key，生成QueryString

注：加密得到的参数需要进行 url encode操作，处理可能出现的特殊字符

### QueryString 加密示例

1. hello=1&world=2
2. aesKey="7psGzvtQh4OooXtmRK7G36oYwYobHGyDDQ81DTfV1KE=",  
aesIv="5a2wShLw7EWa8Fiw+cWYcQ==",  
得到加密字符串xGeToqBGYADr8/KQomlNNg==
3. targetUrl?q=xGeToqBGYADr8/KQomlNNg==

### Body 加密方法

1. 将接口需要的json数据变成json字符串，生成待加密字符串。
2. 商户使用分配的 aesKey 和 aesIv，通过AES算法对1得到的字符串加密，得到已加密字符串。
3. 将2中结果作为body进行请求（相应的Content-Type为text/plain）。

## Body 加密示例

1. {"hello":1,"world":2}
2. aesKey="7psGzvtQh4OooXtmRK7G36oYwYobHGyDDQ81DTfV1KE=",  
aeslv="5a2wShLw7EWa8Fiw+cWYcQ==",  
得到加密字符串oGeoLf5hzd3WduSN51I9yYlCxcgPxWfanCTFj870YHY=
3. 请求body为oGeoLf5hzd3WduSN51I9yYlCxcgPxWfanCTFj870YHY=

## 接口响应 Payload 解密方法

商户使用分配的 aesKey 和 aeslv，通过对应的解密算法就可以得到明文数据。

# 二、调用api接口的 Api Access Token

## Api Token 获取接口

- path: /api/open/v1/channel/token
- method: GET

## 获取 Api Token 接口请求参数

无

## 获取 Api Token 接口请求头参数

参数名	类型	描述	必须
App-Channel	String	渠道号	Y
Timestamp	Long	时间戳	Y
Sign	String	参数签名	Y

## 接口签名生成方法(sign)

1. 筛选。获取App-Channel参数与Timestamp参数，将筛选的参数按照ASCII码顺序递增排序（字母升序排序）。
2. 拼接。将排序后的参数与其对应值，组合成“参数=参数值”的格式，并且把这些参数用&字符连接起来，此时生成的字符串为待签名字符串。
3. 生成。商户将待签名字符串与**商户私钥**连接带入SHA1算法中得出sign。

```

import org.apache.commons.codec.digest.DigestUtils;

/**
 * 获取签名
 */
public static String getSign(String appChannel, String privateKey){
    Map<String, String> requestParams = new HashMap<>();
    requestParams.put("App-Channel", appChannel);
    requestParams.put("Timestamp", timestamp.toString());

    String sortedParams=getSortedParams(requestMap);
    String paramsStrToEncrypt=sortedParams+privateKey;
    return DigestUtils.sha1Hex(paramsStrToEncrypt);
}

/**
 * 排序请求参数并转换成字符串
 */
private static String getSortedParams(Map<String,String> queryParams) {
    if(CollectionUtils.isEmpty(queryParams)){
        return "";
    }
    return queryParams.keySet().stream()
        .sorted()
        .map(key-> key + "=" + queryParams.get(key))
        .collect(Collectors.joining("&"));
}

```

## 获取 Api Token 接口请求示例

eg: `/api/open/v1/channel/token`

## 获取 Api Token 接口响应参数

与公共响应参数相同，payload解密后参数如下：

参数名	类型	描述	必须
createTime	Date	创建时间	Y
expire	Integer	token有效时长，单位秒	Y
accessToken	String	请求令牌	Y
tokenType	String	令牌类型	Y
refreshToken	String	刷新令牌	Y

# 获取 Api Token 接口响应示例

注： 以下json为payload解密后示例

```
{
  "createTime": "Wed Jul 17 18:22:08 CST 2019",
  "expire": 604800,
  "accessToken": "232c814d-13a8-4276-af41-eb2278f5dd78",
  "tokenType": "Bearer",
  "refreshToken": "24a1c804-1449-4855-b8d2-99954e3087d2"
}
```

## 刷新 Api Token 接口

- path: /api/open/v1/channel/token
- method: POST
- contentType: text/plain

## 刷新 Api Token 接口请求参数

无

### 刷新 Api Token 接口请求头参数

参数名	类型	描述	必须
Refresh-Token	String	刷新令牌	Y
Timestamp	Long	时间戳	Y

## 刷新 Api Token 接口请求示例

无

## 刷新 Api Token 接口响应参数

与公共响应参数相同， payload解密后参数如下：

参数名	类型	描述	必须
createTime	Date	创建时间	Y
expire	Integer	token有效时长， 单位秒	Y
accessToken	String	请求令牌	Y

参数名	类型	描述	必须
tokenType	String	令牌类型	Y
refreshToken	String	刷新令牌	Y

## 刷新 Api Token 接口响应示例

注：以下json为payload解密后示例

```
{
  "createTime": "Wed Jul 17 18:22:08 CST 2019",
  "expire": 604800,
  "accessToken": "232c814d-13a8-4276-af41-eb2278f5dd78",
  "tokenType": "Bearer",
  "refreshToken": "24a1c804-1449-4855-b8d2-99954e3087d2"
}
```

## 注销 Api Token 接口

- path: /api/open/v1/channel/token
- method: DELETE

## 注销 Api Token 接口请求参数

无

### 注销 Api Token 接口请求头参数

与公共请求头参数一致

## 注销 Api Token 接口请求示例

无

## 注销 Api Token 接口响应参数

与公共响应参数相同，payload为空

## 注销 Api Token 接口响应示例

与公共响应示例相同，payload为空

# 三、账号模块



# 获取验证码接口

Path: /api/open/v1/captcha/sms  
Request Method: GET

说明： 通过手机号获取验证码，用户账号登录/注册流程需要调用此接口。

## 获取验证码接口请求参数

参数名	类型	描述	必须
Authorization	String	Auth 类型为 Basic Auth，即 Username:Password 的形式，Username 传需要发送验证码的手机号，Password 不传，最后组成的字符串需要经过 Base64 编码，如 “Authorization: Basic MTM2NjY2NjU1NTU6”	Y

## 获取验证码接口请求示例

```
GET /api/open/v1/captcha/sms HTTP/1.1
Host: {your-host}:{your-port}
Content-Type: application/json
Access-Token: {your-token}
Authorization: Basic MTM2NjY2NjU1NTU6
Accept: */*
Cache-Control: no-cache
Accept-Encoding: gzip, deflate
Content-Length: 68
Connection: keep-alive
```

## 获取验证码接口响应参数

无

## 获取验证码接口响应示例

无

# 登录/注册接口

Path: /api/open/v1/user/token  
Request Method: GET

\*\*说明：\*\*通过手机号获取到验证码以后，即可通过该接口进行注册/登录操作，若该手机号之前没有在齐悟端注册过用户，则第一次调用该接口会进行注册，注册/登录成功后会返回用户 token 信息。

登录/注册接口请求参数

参数名	类型	描述	必须
Authorization	String	Auth 类型为 Basic Auth ，即 Username:Password 的形式，Username 传需要发送验证码的手机号，Password 传验证码，最后组成的字符串需要经过 Base64 编码，如 “Authorization: Basic MTM2NjY2NjU1NTU6MTIzNA==”	Y

登录/注册接口请求示例

```
GET /api/open/v1/user/token HTTP/1.1
Host: {your-host}:{your-port}
Content-Type: application/json
Access-Token: {your-token}
Authorization: Basic MTM2NjY2NjU1NTU6MTIzNA==
Accept: */*
Cache-Control: no-cache
Accept-Encoding: gzip, deflate
Content-Length: 68
Connection: keep-alive
```

登录/注册接口响应参数

参数名	类型	描述	必须
accessToken	String	请求 token，请求需要登录验证的接口传入	Y
refreshToken	String	刷新 token，当 accessToken 过期，刷新 token 时需要传入	Y
tokenType	String	token 类型	Y
expire	String	token 寿命，单位：秒	Y

登录/注册接口响应示例

```
{
  "status": 200,
  "retcode": 0,
  "message": "SUCCESS",
  "payload": "{\"accessToken\":\"4a8bc62c-e44e-46dd-b8d0-b359f3163f48\",\"refreshToken\":\"331b5\",
  "timestamp": 1563864181434
}
```

# 刷新 Token 接口

**Path:** /api/open/v1/user/token  
**Request Method:** PUT

**\*\*说明:** \*\*当 access token 过期后，用户需要调用此接口刷新 access token 方能继续访问其他需要验证登录鉴权的接口。

## 刷新 Token 接口请求参数

参数名	类型	描述	必须
Authorization	String	Auth 类型为 Bearer Token，直接传入登录接口获取到的 refreshToken 即可	Y

## 刷新 Token 接口请求示例

```
PUT /api/open/v1/user/token HTTP/1.1
Host: {your-host}:{your-port}
Content-Type: application/json
Access-Token: {your-token}
Authorization: Bearer 331b5837-3de0-423d-824f-9032e0319bbd
Accept: */*
Cache-Control: no-cache
Accept-Encoding: gzip, deflate
Content-Length: 68
Connection: keep-alive
```

## 刷新 Token 接口响应参数

参数名	类型	描述	必须
accessToken	String	请求 token，请求需要登录验证的接口传入	Y
refreshToken	String	刷新 token，当 accessToken 过期，刷新 token 时需要传入	Y
tokenType	String	token 类型	Y
expire	String	token 寿命，单位：秒	Y

## 刷新 Token 接口响应示例

```
{
  "status": 200,
  "retcode": 0,
  "message": "SUCCESS",
  "payload": "{\"accessToken\":\"e7488b0c-a7ff-4054-b17b-6ebe1b4229ff\",\"refreshToken\":\"d487c\", \"timestamp\": 1563864631251}"}
}
```

## 登出接口

**Path:** /api/open/v1/user/token  
**Request Method:** DELETE

**\*\*说明:** \*\*调用该接口退出登录。

### 登出接口请求参数

参数名	类型	描述	必须
Authorization	String	Auth 类型为 Bearer Token ，直接传入登录接口获取到的 accessToken 即可	Y

### 登出接口请求示例

```
DELETE /api/open/v1/user/token HTTP/1.1
Host: {your-host}:{your-port}
Content-Type: application/json
Access-Token: {your-token}
Authorization: Bearer e7488b0c-a7ff-4054-b17b-6ebe1b4229ff
Accept: */*
Cache-Control: no-cache
Accept-Encoding: gzip, deflate
Content-Length: 68
Connection: keep-alive
```

### 登出接口响应参数

无

### 登出接口响应示例

无

# 获取账户信息接口

Path: /api/open/v1/user  
Request Method: GET

\*\*说明：\*\*登录后，通过 access token 获取用户基本信息。

## 获取账户信息接口请求参数

参数名	类型	描述	必须
Authorization	String	Auth 类型为 Bearer Token ，直接传入登录接口获取到的 accessToken 即可	Y

## 获取账户信息接口请求示例

```
GET /api/open/v1/user HTTP/1.1
Host: {your-host}:{your-port}
Content-Type: application/json
Access-Token: {your-token}
Authorization: Bearer 8bcee37a-953b-41bd-b96d-9b1447d2ad59
Accept: */*
Cache-Control: no-cache
Accept-Encoding: gzip, deflate
Content-Length: 68
Connection: keep-alive
```

## 获取账户信息接口响应参数

参数名	类型	描述	必须
infol	String	用户 ID	Y
nickName	String	用户昵称	Y
linkMansId	List	联系人信息	Y
userPhone	String	用户手机号	Y

## 获取账户信息接口响应示例

```
{
  "status": 200,
  "retcode": 0,
  "message": "SUCCESS",
  "payload": "{\"infoId\":\"1\", \"nickName\":\"450601\", \"linkMansId\": [], \"userPhone\":\"185555\", \"timestamp\": 1563865639947}"
}
```

## 新增子账号接口（不需要对接齐悟账号的客户）

**Path:** /api/open/v1/user/sub  
**Request Method:** POST

**\*\*说明：** \*\*可以通过该接口在渠道商户下增加一个子账号，进行下单操作时是使用子账号下单。

### 新增子账号接口请求参数

参数名	类型	描述	必须
businessId	String	接入方对于此账号定义的唯一标识，比如可以传设备唯一标识，每个设备绑定一个账号	N

### 新增子账号接口请求示例

```
{
  "businessId": "testBusinessId"
}
```

### 新增子账号接口响应参数

参数名	类型	描述	必须
id	String	子账号 ID	Y
appChannel	String	商户渠道号	Y
phone	String	客户对外电话号，用于下单时给该手机号发短信通知等场景	Y
businessId	String	接入方对于此账号定义的唯一标识，比如可以传设备唯一标识，每个设备绑定一个账号	Y
createTime	Date	子账号创建时间	Y

## 新增子账号接口响应示例

```
{
  "status": 200,
  "retcode": 0,
  "message": "SUCCESS",
  "payload": "{\"phone\":\"10086\",\"createTime\":\"2019-07-18T17:51:57.079\",\"businessId\":\"t\", \"timestamp\": 1563443517094",
}
```