

福建高速电商系统

翼码开放平台

(open.wangcaio2o.com)

目 录

文档说明

1.文档说明

2.签名计算

3.接口说明及报文

3.1.会员信息

3.1.1.用户随机码获取

3.1.2.用户信息查询

文档说明

文档概述

版本说明

本文档《福建高速电商系统》由 lidong 编写，最后编辑于 2026-03-05 23:09:42。

发布日期

2026-03-06

在线文档

<https://open.wangcaio2o.com/docs-fjgsgl-member.html>

注意事项

如果当前文档生成时间太久，请到 [翼码开放平台\(open.wangcaio2o.com\)](https://open.wangcaio2o.com) 获取最新的文档，以免信息不对称！

如果您发现文档内容有不恰当的地方，请向我们 [反馈意见](#)。

感谢您使用 [翼码开放平台\(open.wangcaio2o.com\)](https://open.wangcaio2o.com)。

1.文档说明

引言

本接口规范文档定义了福建高速电商系统会员的接口，各应用与该接口服务交互需遵循此规范。本接口规范主要供架构设计师、开发人员和测试人员阅读使用，也可供产品验收时参考使用。

HTTP接口规范:

- 报文格式: JSON
- 请求方式: POST
- 字符编码: UTF-8
- URL地址: <https://fjgs-gateway.imageco.cn>

接入准备

- 获取应用的appid、appSecret。appSecret用于接口签名和验签。

返回码

返回码	描述
0	成功
500	服务内部异常
102	验签失败
106	appid错误

2. 签名计算

接口签名规范

签名目的

防止请求篡改与重放。服务端基于约定的请求要素与双方共享的 appSecret 计算签名，通过校验签名一致性、timestamp 时间窗及 nonce 去重机制，保障请求的安全性。

签名要素（参与签名的数据）

需对以下三类数据按约定规则规范化后拼接，作为签名计算的基础：

1. URL 的查询参数 (query string)
2. 需要签名的请求头 (自定义头部)，例如：appId、nonce、timestamp
3. 请求体 (raw body 字符串)，通常为 JSON 原始字符串 (空体则视为空字符串)

注：编码/解码方式必须与后端一致。本文示例规则为：对 query 和 header 的键值先进行 URL 解码，再按键名称字典序排序并使用 `key=value` 串联。

规范化与拼接规则（详细步骤）

1. Query 参数处理

- 从请求 URL 中提取 `?` 之后的 queryString；若无 query，则视为空字符串。
 - 按 key 升序 (字典序) 排序，以 `k=v` 形式拼接，不同参数间用 `&` 连接，得到 `sortedQueryString`。
- 示例：`a=1&b=2`

2. Header 处理

- 提取约定的参与签名的 header 列表 (示例：`['appId','nonce','timestamp']`)，header 名称大小写以约定为准 (示例使用小驼峰)。
- 对每个存在的 header 值取原始字符串 (不额外 encode)，不存在的 header 直接跳过。
- 按 key 升序排序，以 `k=v` 形式拼接，不同 header 间用 `&` 连接，得到 `sortedHeaderString`。

3. Body 处理

- 若 `Content-Type` 为 `application/json` 且 body 有内容，取 body 的原始 raw 字符串 (不修改空格/缩进，与服务端序列化方式一致)。
- 若无 body，视为空字符串 `''`。

4. 最终拼接

示例约定：

```
signatureBaseString = sortedQueryString + '&' + sortedHeaderString + '&' + requestBodyString
```

说明：即使某一部分为空，仍保留 `&` 作为分隔 (需与后端达成一致，本文示例采用固定三段拼接并保留分隔符)。

签名算法

1. 使用 HMAC-SHA256 (或后端指定算法)，以 `appSecret` 作为密钥，对 `signatureBaseString` 计算 HMAC-SHA256 摘要。
2. 将计算结果按约定编码。
3. 将最终签名写入请求头：`sign: <签名值>`。

防重放与校验

- **timestamp**：仅在服务端允许的时间窗 (例如 ± 5 分钟) 内有效。
- **nonce**：服务端记录最近 N 分钟内的 nonce，重复的 nonce 将直接拒绝请求。

示例（完整举例）

已知条件

- appId = test
- appSecret = 123456
- 请求 URL: `https://fjgs-gateway.imageco.cn/open-api/member/verification?userId=286&price=2&bizType=11&bizId=2865&mode=1¬e=11`
- nonce = e7eb4265-885d-40eb-ace3-2ecfc34bd635
- timestamp = 1717494535932
- 请求 body = 空字符串 (raw body = " ")

计算步骤

1. 处理 Query 参数

解析并 decode 后得到对象:

```
{ bizId: "2865", bizType: "11", mode: "1", note: "11", price: "2", userId: "286" }
```

按 key 字典序排序后拼接:

```
sortedQueryString = "bizId=2865&bizType=11&mode=1&note=11&price=2&userId=286"
```

2. 处理 Header 参数

参与签名的 header 列表: `['appId', 'nonce', 'timestamp']`

有效 header 及值:

- appId=test
- nonce=e7eb4265-885d-40eb-ace3-2ecfc34bd635
- timestamp=1717494535932

排序后拼接:

```
sortedHeaderString = "appId=test&nonce=e7eb4265-885d-40eb-ace3-2ecfc34bd635&timestamp=1717494535932"
```

3. 拼接签名基础串

```
signatureBaseString = "bizId=2865&bizType=11&mode=1&note=11&price=2&userId=286&appId=test&nonce=e7eb4265-885d-40eb-ace3-2ecfc34bd635&timestamp=1717494535932&"
```

注意: 末尾的 `&` 因 body 为空保留, 需与后端规则一致。

4. 计算签名

使用 HMAC-SHA256 算法, 密钥为 `123456`, 对 `signatureBaseString` 计算摘要并转 hex 大写:

```
signature = hmac_sha256_hex("123456", signatureBaseString)
```

(示例占位符: 9A1B2C3D4E5F..., 实际值需通过代码计算)

5. 设置请求头

将签名写入请求头: `sign: 9A1B2C3D4E5F...`

总结

1. 签名计算需包含 **query 参数**、**指定 header (appId/nonce/timestamp)**、**请求体** 三类数据, 且需与后端统一编码/排序/拼接规则。
2. 签名核心是基于 `appSecret` 对规范化后的 `signatureBaseString` 执行 HMAC-SHA256 计算, 并按约定编码。
3. 防重放依赖 `timestamp` 时间窗和 `nonce` 唯一性校验, 需确保每次请求的 nonce 唯一、timestamp 有效。

3.接口说明及报文

- [3.1.会员信息](#)

3.1.会员信息

- 用户随机码获取
- 用户信息查询

3.1.1.用户随机码获取

根据用户token获取随机码

接口地址: /open-api/member/user/getRandomCode

请求方式: POST

接口描述: 通过用户token获取随机码

公共请求参数:

参数名称	参数说明	请求类型	是否必须	数据类型	示例值
appld	应用ID	header	true	string	test
nonce	随机字符串	header	true	string	75ba4a58-8db0-4ce0-b403-2ccc8dbaea72
timestamp	时间戳	header	true	string	1772763315016
sign	签名串	header	true	string	89f4ad32ea7151cc33477215c24e8020bfb801a9...

业务请求参数:

参数名称	参数说明	请求类型	是否必须	数据类型	示例值
token	用户token	query	true	string	eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9eyJ1c2VySWQiOiJlY200ASIm5hbWUiOiLkuK3lm70iLCJleHAiOjE3MzU2MzIwMDB9

请求示例

https://fjgs-gateway.imageco.cn/open-api/member/user/getRandomCode?

token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9eyJ1c2VySWQiOiJlY200ASIm5hbWUiOiLkuK3lm70iLCJleHAiOjE3MzU2MzIwMDB9

响应参数:

参数名称	参数说明	类型	示例值
code	返回码	integer	0
msg	返回信息	string	
data	业务数据	string	eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9eyJ1c2VySWQiOiJlY200ASIm5hbWUiOiLkuK3lm70iLCJleHAiOjE3MzU2MzIwMDB9

响应示例:

```

1. {
2.   "code": 0,
3.   "msg": "",
4.   "data": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9eyJ1c2VySWQiOiJlY200ASIm5hbWUiOiLkuK3lm70iLCJleHAiOjE3MzU2MzIwMDB9"
5. }
```

3.1.2.用户信息查询

根据随机码获取用户信息

接口地址: `/open-api/member/user/getUserInfoByRandomCode`

请求方式: `POST`

接口描述: 通过随机码获取用户信息

公共请求参数:

参数名称	参数说明	请求类型	是否必须	数据类型	示例值
appId	应用ID	header	true	string	test
nonce	随机字符串	header	true	string	75ba4a58-8db0-4ce0-b403-2ccc8dbaea72
timestamp	时间戳	header	true	string	1772763315016
sign	签名串	header	true	string	89f4ad32ea7151cc33477215c24e8020bfb801a9...

业务请求参数:

参数名称	参数说明	请求类型	是否必须	数据类型	示例值
token	用户token	query	true	string	eyJhbGciOiJIUzI1Ni...

请求示例

`https://fjgs-gateway.imageco.cn/open-api/member/user/getUserInfoByRandomCode?`

`token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9eyJ1c2VySWQiOiJlZ200DAIm5hbWUiOiJlLkkuK3lm70iLCJleHAiOjE3MzU2MzIwMDB9`

响应参数:

参数名称	参数说明	类型	示例值
code	返回码	integer	0
msg	返回信息	string	
data	业务数据	JSONObject	参考data业务数据

data 业务数据 (JSONObject)

参数名称	参数说明	类型	示例值
mobile	手机号	string	19999999999
status	状态	string(byte)	2
nickname	用户昵称	string	李四
avatar	头像	string	https://www.imageco.cn/x.png
name	用户姓名	string	李四
sex	用户性别	integer	1
areald	所在地编号	integer	4371
areaName	所在地全程	string	上海上海市普陀区
birthday	出生日期	string(date-time)	2023-03-12
mark	会员备注	string	我是备注
tagIds	会员标签	array	[1,2]

参数名称	参数说明	类型	示例值
levelId	会员等级编号	integer	1
groupId	用户分组编号	integer	1
id	编号	integer	23788
registerIp	注册 IP	string	127.0.0.1
loginIp	最后登录IP	string	127.0.0.1
loginDate	最后登录时间	string	
createTime	创建时间	string	
point	积分	integer	10
coin	金币	integer	10
experience	经验值	integer	1024
level	用户等级	Level	Level
id	等级编号	integer	1
name	等级名称	string	翼码科技
level	等级	integer	1
icon	等级图标	string	https://www.imageco.cn/fjgsgl.jpg
brokerageEnabled	是否成为推广员	boolean	true

响应示例:

```

1. {
2.   "code": 0,
3.   "msg": "",
4.   "data": {
5.     "mobile": 19999999999,
6.     "status": 2,
7.     "nickname": "李四",
8.     "avatar": "https://www.imageco.cn/x.png",
9.     "name": "李四",
10.    "sex": 1,
11.    "areaId": 4371,
12.    "areaName": "上海上海市普陀区",
13.    "birthday": "2023-03-12",
14.    "mark": "我是小备注",
15.    "tagIds": "[1,2]",
16.    "levelId": 1,
17.    "groupId": 1,
18.    "id": 23788,
19.    "registerIp": "127.0.0.1",
20.    "loginIp": "127.0.0.1",
21.    "loginDate": "",
22.    "createTime": "",
23.    "point": 10,
24.    "coin": 10,
25.    "experience": 1024,
26.    "level": {
27.      "id": 1,
28.      "name": "翼码科技",
29.      "level": 1,
30.      "icon": "https://www.imageco.cn/fjgsgl.jpg"
31.    },
32.    "brokerageEnabled": true
33.  }
34. }
```