

佳诺虚充业务 API 接口文档

(V3.0)

更新日期：2020-03-05

目录

1. 请求和响应报文.....	3
2. API 请求地址与账户信息.....	3
3. 提单与查询接口定义.....	4
4. 订单回调通知.....	6
5. 账户接口定义.....	7
6. 业务定义.....	7
7. 错误定义.....	8
8. 签名算法.....	8

1.请求和响应报文

接口协议说明	
网络协议	HTTP
请求方法	POST
请求格式	ContentType: application/json
应答格式	JSON
字符集	UTF-8

接口采用 http 进行交互，所有提交参数均 UTF-8 编码，采用 POST 方式提交

(ContentType: application/json)。

返回报文统一使用 JSON，编码为 UTF-8，报文中某个可选参数的值为空，则该标签可以不出现。

标注 (签名) 的参数值在不为空时，需要对其进行签名校验，签名算法参见文档尾

2.API 请求地址与账户信息

地址与账户信息	
请求 URL (网关)	http://{DomainName}/ApiAgent/GatewayV3
	其中 DomainName 为实际的虚充平台域名或者 IP
UserId	帐号 ID (注意不是登陆名)
ApiKey	API 接口密钥

注意：

- 1、请求接口的参数都需要区分大小写的！！！！
- 2、请求接口的选填参数在不为空时，需要参与签名
- 3、请求接口的网关地址都是一样的，通过 Service 命令参数区分接口的请求
- 4、以上 3 个关键的信息，请联系平台商务获取

3.提单接口定义

3.1. 下单接口

请求

Service	下单命令(签名)	SubmitOrder 下单接口固定值
UserId	帐号 ID(签名)	注意不是登陆名
BizType	业务类型(签名)	见业务定义，如：ECARD 、OIL
OrderNo	订单号(签名)	不大于 32 位的唯一订单号
ProductId	标准产品 ID(签名)	直充与卡密的产品 ID 是相互独立的
AccountVal	充值目标账号(签名)	
Phone	[选填]手机号(签名)	用于接收通知消息
BuyNum	[选填]购买数量(签名)	为空时默认为 1
CustomerIP	[选填]终端 IP(签名)	此 IP 需填写下单人的 IP，而不是接口调用方 IP 充值 Q 币、区域产品时必填，其他产品可以为空
ExtraData	[选填]附加数据(签名)	特定产品充值时需要的参数，默认直接为空 格式为 JSON 字符串，注意是序列化后的字符串
Time	时间戳(签名)	1970-1-1 (UTC) 到现在的总秒数
Sign	签名值	参见签名算法

响应

code	消息代码	成功返回 0，出错返回对应的错误码
msg	消息文本	
BizType	业务类型	原样返回请求中的业务类型
OrderNo	订单号	原样返回请求中的订单号
OrderStatus	订单状态	如果 code==0 时，才会返回： UNDERWAY：下单成功，异步充值中

注意：

- 如果接口超时，请不要直接对订单状态进行处理，可以在稍后通过查询接口进行订单状态获取，成功或者失败平台也会回调。
- 如果发起相同订单号，平台会对订单进行对比，如果传入参数业务类型、产品、目标账号、数量都相同的订单，平台会对接口幂等操作（返回该订单当前的实时状态），否则对接口返回订单号重复的错误。

3.2. 查询接口

请求

Service	查询接口命令(签名)	QueryOrder 查询接口固定值
UserId	帐号(签名)	注意不是登陆名
BizType	业务类型(签名)	见业务定义, 如: ECARD 、 OIL
OrderNo	订单号(签名)	
Time	时间戳(签名)	1970-1-1 (UTC) 到现在的总秒数
Sign	签名值	参见签名算法

响应

code	消息代码	成功返回 0, 出错返回对应的错误码
msg	消息文本	
BizType	业务类型	原样返回请求中的业务类型
OrderNo	订单号	原样返回请求中的订单号
OrderStatus	订单状态	如果 code==0 时, 才会返回: SUCCESS: 充值成功 FAILED: 充值失败 UNDERWAY: 下单成功, 充值中 NOTEXIST: 订单不存在, 下单失败
ProductData	订单产品数据	返回类型为 JSON 字符串,读取时需要 2 次解析, 特定产品订单才会返回, 直充都返回空, 卡密产品 会返回卡密信息。 卡密产品格式: (JSON 数组列表) [[type: "0" ,// 0 卡密, 1 激活 URL, 2: 卡密+URL code: "E1327419237491",// 卡号 key: "345352",// 卡密钥 url: "",// 卡激活 URL 地址 effstart: "2020-02-20",// 有效开始时间 effend: "2020-03-19",// 有效结束时间 extinfo: null,// 其它信息]]

4. 订单回调通知

4.1. 推送状态

接口用户提供接收 http post 推送的 url 地址。接收到推送数据解析成功后返回响应文本，即为推送成功，订单状态成功或者失败时才会推送，其他状态不会推送。

请求 http post **ContentType:** application/json

UserId	账号(签名)	注意不是登陆名
BizType	业务类型(签名)	见业务定义，如：ECARD 、OIL
OrderNo	订单号(签名)	
AccountVal	充值目标账号(签名)	
OrderStatus	订单状态(签名)	SUCCESS：充值成功 FAILED：充值失败 UNDERWAY：下单成功，充值中
ProductData	订单产品数据(签名)	返回类型为 JSON 字符串,读取时需要 2 次解析，特定产品订单才会返回，直充都返回空，卡密产品会返回卡密信息。 卡密产品格式：(JSON 数组列表) [[type: "0", // 0 卡密, 1 激活 URL, 2: 卡密+URL code: "E1327419237491", // 卡号 key: "345352", // 卡密钥 url: "", // 卡激活 URL 地址 effstart: "2020-02-20", // 有效开始时间 effend: "2020-03-19", // 有效结束时间 extinfo: null, // 其它信息]]
Time	时间戳(签名)	1970-1-1 (UTC) 到现在的总秒数
Sign	签名值	

回调响应

code	消息代码	成功返回 0，出错返回对应的错误码
msg	消息文本	

成功返回 JSON 文本：{"code": 0}

失败返回 JSON 文本：{"code": -1, "msg": "错误描述"}

注意：

接口用户给我们的回调响应不是订单状态，是服务器收到了我们的通知，无论订单是否成功都需要返回成功接收到的 JSON 文本，避免双方服务器资源的浪费。

5.账户接口定义

5.1. 查询账户余额

请求

Service	查询余额命令(签名)	QueryBalance 固定值
UserId	帐号(签名)	注意不是登陆名
BizType	业务类型(签名)	见业务定义，如：ECARD 、OIL
Time	时间戳(签名)	1970-1-1 (UTC) 到现在的总秒数
Sign	签名值	参见签名算法

响应

code	消息代码	成功返回 0，出错返回对应的错误码
msg	消息文本	
Balance	当前账户余额	单位 分，类型允许负数

6.业务定义

OIL	加油卡业务
ECARD	点卡会员业务 (Q 币、会员、视频、游戏等)
MOBILE	手机话费业务

7.错误定义

0	成功
101	参数错误 或者 缺少必须的业务参数
102	签名错误
103	用户状态不正常 【联系商务确定 ID、密钥是否正确，产品、接口是否都已经配置】
104	余额不足
105	产品不存在或者状态不正常
106	目标充值账号不合法或者状态不正常
107	时间戳相差过大
108	订单号已经存在
109	协议版本错误 或者 不能识别的或者未开放的接口服务
110	产品提交数量不符合规范值【特定产品会要求】
111	IP 未在白名单中
999	未知错误(不直接作为失败，是否成功请联系平台确认)

8.签名算法

按照 asc 排序键值对参数名排序(需要区分大小写)，如果选填参数同时值为空时不能参与签名计算，拼接键值对字符串，需要把 api key 放到要签名的键值对最后。

例如 sign=MD5(参数 A 名字参数 A 值参数 B 名字参数 B 值参数 C 名字参数 C 值密钥)

用 32 位 MD5 加密，采用 UTF-8 编码后的 MD5，生成的 MD5 小写

如：

BizType= OIL

UserId= Test8888

ApiKey= 0CC2EC0AE5AD4C2DA0FD419D36EBA160

Time= 131653774326942493

得到的的字符串：

BizTypeOILTime131653774326942493UserIdTest88880CC2EC0AE5AD4C2DA0FD419D36EBA160

生成的 Sign (MD5 加密后转小写)：

8d1a1f3fd7f1d0e87ce1a705c971cea9

9.请求示例

9.1. 直充提交订单

请求:

http://{DomainName}/ApiAgent/GatewayV3

采用 POST, ContentType: application/json

POST

http://10.211.55.3:8000/ApiAgent/GatewayV3

Params

Authorization

Headers (9)

Body

Pre-request Script

Tests

Settings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

1

{

2

"Service": "SubmitOrder",

3

"UserId": "ZXC002",

4

"BizType": "ECARD",

5

"OrderNo": "ZXC00260202073749123258395",

6

"ProductId": "XMG003", /*对应直充产品ID*/

7

"AccountVal": "78677168",

8

"Time": "1582790444",

9

"BuyNum": "1",

10

"CustomerIP": "125.64.91.103",

11

"Phone": "18508388866",

12

"Sign": "dad4ab674ffd4a995790713464f743f0"

13

}

签名串:

AccountVal78677168BizTypeECARDBuyNum1CustomerIP125.64.91.103OrderNoZXC00260202073749123258395Phone18508388866ProductIdXMG003ServiceSubmitOrderTime1582790444UserIdZXC002CD97B664C0A54152BF947C521ED1BB79

签名值: dad4ab674ffd4a995790713464f743f0

返回:

数据格式: JSON

1

{

2

"code": 0,

3

"msg": "操作成功",

4

"BizType": "ECARD",

5

"OrderStatus": "UNDERWAY",

6

"OrderNo": "ZXC00260202073749123258395"

7

}

9.2. 直充订单查询

请求: <http://{DomainName}/ApiAgent/GatewayV3>

```
1 {  
2   "Service": "QueryOrder",  
3   "UserId": "ZXC002",  
4   "BizType": "ECARD",  
5   "OrderNo": "ZXC00260202073749123258395", /*直充订单号*/  
6   "Time": "1582791378",  
7   "Sign": "6a030889eb21c6947fc4f707374eaffe"  
8 }
```

返回:

```
1 {  
2   "code": 0,  
3   "msg": "操作成功",  
4   "BizType": "ECARD",  
5   "OrderStatus": "UNDERWAY",  
6   "OrderNo": "ZXC00260202073749123258395"  
7 }
```

9.3. 直充订单回调（异步推送）

请求: (由我们发起, 向绑定的回调地址推送)

```
1 {  
2   "UserId": "ZXC003",  
3   "BizType": "ECARD",  
4   "OrderNo": "ZXC00360198242128720095469",  
5   "AccountVal": "78677168",  
6   "OrderStatus": "SUCCESS", /*状态: SUCCESS 或者 FAILED*/  
7   "Time": "1582771928",  
8   "Sign": "22bec14c4e1dff5558b1ffb65bf968e6"  
9 }
```

返回: (接口用户收到后, 无论业务是否成功, 都请返回{"code":0})

```
1 {"code":0}
```

9.4. 卡密提交订单

请求: <http://{DomainName}/ApiAgent/GatewayV3>

```
1 {  
2     "Service": "SubmitOrder",  
3     "UserId": "ZXC002",  
4     "BizType": "ECARD",  
5     "OrderNo": "ZXC00260198163417526499448",  
6     "ProductId": "BDTXSP001", /*卡密 产品ID*/  
7     "AccountVal": "78677168",  
8     "Time": "1582771343",  
9     "CustomerIP": "125.64.91.103",  
10    "Phone": "18508388866",  
11    "Sign": "29f5245b042790ec47c90236e0b26326"  
12 }
```

返回:

```
1 {  
2     "code": 0,  
3     "msg": "操作成功",  
4     "OrderStatus": "UNDERWAY", /*卡密订单当前状态*/  
5     "OrderNo": "ZXC00260198163417526499448",  
6     "BizType": "ECARD"  
7 }
```

9.5. 卡密订单查询

请求: <http://{DomainName}/ApiAgent/GatewayV3>

```
1 {  
2     "Service": "QueryOrder",  
3     "UserId": "ZXC002",  
4     "BizType": "ECARD",  
5     "OrderNo": "ZXC00260097289353654478464", /*卡密 订单ID*/  
6     "Time": "1582790437",  
7     "Sign": "255b8db9fcdb0ad2b0007d732998873a"  
8 }
```

返回:

```
1 {  
2     "code": 0,  
3     "msg": "操作成功",  
4     "OrderStatus": "SUCCESS",  
5     "OrderNo": "ZXC00260097289353654478464",  
6     "ProductData": "[{"type": "\0", "code": "\E62701079097945723151", "key": "\728554", "url": null, "effstart": "\2020-02-21", "effend": "\2020-02-21", "extinfo": null}]", /*卡密信息, 需要2次解析*/  
7     "BizType": "ECARD"  
8 }
```

9.6. 卡密订单回调（异步推送）

请求：（由我们发起，向绑定的回调地址推送）

```
1 {
2   "UserId": "ZXC003",
3   "BizType": "ECARD",
4   "OrderNo": "ZXC00360198235852009653541",
5   "AccountVal": "78677168",
6   "OrderStatus": "SUCCESS",
7   "ProductData": "[{"type": "\0", "code": "\4534352345", "key": "\243523", "url": "\452345", "effstart": "\2020-02-01",
8     "effend": "\2021-02-27", "extinfo": null}], /*注意是字符串，需要2次解析*/
9   "Time": "1582771675",
10  "Sign": "2427c194847e09f8e157ade4386a25a5"
11 }
```

返回：（接口用户收到后，无论业务是否成功，都请返回{"code":0}）

```
1 {"code":0}
```