

编号: CCN-ZJ-4..20-18

CCN-虚拟商品充值服务文档

上海中商网络股份有限公司
2018 年 4 月

文档变更历史				
版本号	日期（年/月/日）	说明	编辑	审核
V 1.0.0.0	2018/04/20	初稿	金志新、陈天云、徐磊	金志新

目录

CCN-虚拟商品充值服务文档.....	1
1.1 术语与缩写解释	3
1.2 通用公共定义	3
1.2.1 系统执行状态	3
1.2.2 API 队列订单状态充值状态	5
1.2.3 统一产品列表	5
1.3 API 接口	7
1.3.1 直冲接口	7
1.3.2 查询订单状态	8
1.3.3 订单回调推送	9
1.3.4 余额查询	10
1.3.5 卡密提取	11

1.1 术语与缩写解释

缩写、术语	解 释
Server	测试服务器： http://152l8u0817.51mypc.cn:10894/Voucher 正式服务器： http://api.recharge.yesno.com.cn/
appid	系统分配的 appid，线下提供
appsecret	系统分配的 appsecret，线下提供
keyst	系统分配的密钥，线下提供
timestamp	时间戳，精确到毫秒
orderid	订单号码
account	虚拟卡券类： 值为需要充值账号 流量/话费： 值为手机号码 红包/转账： 值为 openid（小程序或者公众号）
quantity	充值数量 视讯类： 值为充值的虚拟卡数量 流量/话费： 只能为 1 红包/转账： 如果是普通红包只能为 1，裂变红包 3~20； 如果是转账只能为 1
money	充值金额 视讯类： 不需要 流量/话费： 不需要 红包/转账： 普通红包：1~200 裂变红包：money/quantity 大于等于 1 转账：1~1000
productid	充值的产品类型代码，见产品列表
mark	红包/转账描述
wishing	红包祝福语
actName	红包活动名称
IsNormalRedPacket	true:普通红包 false:裂变红包 默认普通红包
sign	MD5(appid+appsecret+keyst+v1+v2+v3.....)

	需要把 appid 、 appsecret 、 keyst 放到要签名的列表中，按照 ASCII 排序方法，对参数的值进行排序，拼接字符串，再用 32 位 md5 加密
AES 加密密钥	默认 keyst 长度 16 位字符

1.2 通用公共定义

1.2.1 系统执行状态

系统状态代码	名 称	描 述
0000	系统异常	存在未处理异常
1000	成功	成功
2001	参数不正确	参数不正确
2002	充值签名错误	订单签名错误，无法充值
2003	TOKEN 失效	Token 已经失效，请重新获取
2004	IP 未授权	IP 白名单未配置
2005	IP 格式不正确	IP 格式不正确
2006	手机号码格式不正确	手机号码格式不正确
3002	充值成功订单不允许二次充值	充值成功订单不允许二次充值
3013	队列中已存在	队列中已存在
3014	订单号重复	订单号不允许重复
3016	不支持该商品	不支持该商品
3018	账户余额不足	账户余额不足
3019	本地卡密数不足	本地卡密数不足
3020	预扣失败	预扣失败
3021	卡号卡密提取数量不正确	卡号卡密提取数量不正确
3022	Appid 无效	Appid 无效
3023	时间戳格式不正确	时间戳格式不正确
3024	订单号不能为空	订单号不能为空
3025	帐号不能为空	帐号不能为空
3026	Quantity 不能为空	Quantity 不能为空
3027	订单不存在	订单不存在

3028	时间戳已过期	时间戳已过期
3029	卡号卡密提取数量不正确	卡号卡密提取数量不正确
4001	不支持该商品	不支持该商品
4002	未配置商品源	未配置商品源
4003	该应用未配置余额信息	该应用未配置余额信息
9001	不支持充值	不支持充值
9002	订单不能再次充值	订单不能再次充值
9003	不存在此订单	不存在此订单
9004	本地卡密不足	本地卡密不足
10000	自定义内容错误	自定义内容错误

1.2.2 充值方式

充值方式	名称	描述
1	直充	直接提交充值
2	卡密充	卡密提取

1.2.3 API 队列订单状态充值状态

订单状态	名称	描述
0	初始化	数据初始化成功
1	处理成功	处理成功
2	已提交	已提交，等待充值
3	处理失败	订单完成，但失败

1.2.4 统一产品列表

产品代码	名称	描述
1001	1 元话费	1 元话费
1002	5 元话费	5 元话费
1003	10 元话费	10 元话费

1004	20 元话费	20 元话费
1005	30 元话费	30 元话费
1006	50 元话费	50 元话费
1007	100 元话费	100 元话费
1008	200 元话费	200 元话费
1009	300 元话费	300 元话费
1010	500 元话费	500 元话费
2001	3 元流量	3 元流量 移动
2002	5 元流量	5 元流量 移动
2003	10 元流量	10 元流量 移动
2004	20 元流量	20 元流量 移动
2005	30 元流量	30 元流量 移动
2006	50 元流量	50 元流量 移动
2007	70 元流量	70 元流量 移动
2008	100 元流量	100 元流量 移动
2009	130 元流量	130 元流量 移动
2010	180 元流量	180 元流量 移动
2011	280 元流量	280 元流量 移动
2013	3 元流量	3 元流量 联通
2014	6 元流量	6 元流量 联通
2015	10 元流量	10 元流量 联通
2016	15 元流量	15 元流量 联通
2017	30 元流量	30 元流量 联通
2018	1 元流量	1 元流量 电信
2019	2 元流量	2 元流量 电信
2020	5 元流量	5 元流量 电信
2021	7 元流量	7 元流量 电信
2022	10 元流量	10 元流量 电信
2023	15 元流量	15 元流量 电信
2024	30 元流量	30 元流量 电信

2025	50 元流量	50 元流量 电信
5001	爱奇艺年卡	爱奇艺年卡
5002	爱奇艺季卡	爱奇艺季卡
5003	爱奇艺月卡	爱奇艺月卡
5004	优酷年卡	优酷年卡
5005	优酷季卡	优酷季卡
5006	优酷月卡	优酷月卡
5007	腾讯年卡	腾讯年卡
5008	腾讯季卡	腾讯季卡
5009	腾讯月卡	腾讯月卡
5010	滴滴专车 20 元	20 元面值
5011	滴滴专车 30 元	30 元面值
5012	滴滴专车 50 元	50 元面值
5013	滴滴专车 70 元	70 元面值
5014	滴滴专车 100 元	100 元面值
5015	滴滴顺风车 20 元	20 元面值
5016	滴滴顺风车 30 元	30 元面值
5017	滴滴顺风车 50 元	50 元面值
5018	滴滴专车 5 元	5 元面值
5019	滴滴专车 10 元	10 元面值
5020	饿了么代金券 5 元	5 元代金券
5021	饿了么代金券 10 元	10 元代金券
6001	微信红包	微信红包 默认[1, 200]
7001	微信零钱转账	微信零钱转账 [1, 1000]

1.3 API 接口

1.3.1 直冲接口

名称:	商品直充接口
功能说明:	目前支持流量、话费、微信红包、微信零钱、视讯类直充
调用方法:	/Recharge
方法:	POST 提交 JSON 格式: Server/Recharge
输入参数:	<p>1、红包接口入参，以及参与签名的参数（红色标记） appid orderid timestamp account money mark wishing actName productid quantity sign isNormalRedPacket</p> <p>2、话费/流量接口入参，以及参与签名的参数（红色标记） appid orderid timestamp account quantity productid sign</p> <p>3、卡券接口入参，以及参与签名的参数（红色标记） appid orderid timestamp quantity productid sign</p> <p>4、零钱转账接口入参，以及参与签名的参数（红色标记） appid orderid timestamp account money mark productid quantity sign wishing actName isNormalRedPacket</p>
返回值:	<p>JSON 格式</p> <p>State: 系统状态</p> <p>Message: 状态描述</p> <p>RawJson: JSON 数据</p> <p>PostUrl: 提交的地址</p>
举例	<p>入参:</p> <pre>{ "appid": "89757", "orderid": "{orderid}", "account": "18301956235", "productid": "6001", "money": "", "mark": "", "wishing": "", "actName": "", "quantity": "1", "isNormalRedPacket": " " }</pre>

	<pre>"timestamp": "1528944783005", "sign": "23" } 出参: { "RawJson": null, "PostUrl": null, "State": 1000, "Message": "成功" }</pre>
--	---

1.3.2 查询订单状态

名称:	查询订单状态
功能说明:	根据订单号码查询
调用方法:	/Query
方法:	POST 提交 JSON 格式: Server/Query
输入参数:	<p>红色参数代表需要进行签名加密的参数</p> <p>appid orderid timestamp sign</p>
返回值:	<p>JSON 格式</p> <p>Code: 系统状态 Message: 状态描述 RawJson: JSON 数据 PostUrl: 提交的地址 Data: Orderid: 订单号 State: 订单状态 0 初始化 1 处理成功 2 已提交 3 处理失败 SubmitDate: 提交订单时间 Appid: 平台帐号</p>
举例	入参:

	<pre>{ "appid": "1", "orderid": "7383175436425703", "timestamp": "1528944783005", "sign": "51c711388a63a9e813a31e7fb272c9d0" }</pre> <p>出参:</p> <pre>{ "Data": { "Orderid": "7383175436425703", "State": 0, "SubmitDate": "2018-05-18T19:04:11", "Appid": "1" }, "RawJson": null, "PostUrl": null, "State": 1000, "Message": "成功" }</pre>
--	---

1.3.3 订单回调推送

名称:	客户接收订单回调接口
功能说明:	客户提供 api 地址, CCN 推送给你订单充值信息
调用方法:	POST 提交客户提供 API 地址
方法:	该回调地址需要第三方响应: success, 否则会发起第二次请求 (最多 3 次), 每半小时响应一次
输入参数:	<p>红色参数代表需要进行签名加密的参数</p> <p>orderid code message (url utf-8 编码) account sign</p>
返回值:	<p>string 格式如:</p> <p>Response.ContentType = "text/plain";</p>

	Response.Write("success");
--	----------------------------

1.3.4 余额查询

名称:	客户余额查询
功能说明:	客户查询当前余额
调用方法:	/Balance
方法:	POST 提交 JSON 格式: Server/Balance
输入参数:	<p>红色参数代表需要进行签名加密的参数</p> <p>appid timestamp sign</p>
返回值:	<p>JSON 格式</p> <p>Code: 系统状态</p> <p>Message: 状态描述</p> <p>RawJson: JSON 数据</p> <p>PostUrl: 提交的地址</p> <p>Data 帐号余额</p> <p>balance: 余额</p>
举例	<p>入参:</p> <pre>{ "appid": "2", "timestamp": "1528944783005", "sign": "sign" }</pre> <p>出参:</p> <pre>{ "Data": { "Balance": 78 }, "RawJson": null, "PostUrl": null, "State": 1000, "Message": "成功" }</pre>

	}
--	---

1.3.5 卡密提取

名称:	卡号卡密提取
功能说明:	卡号卡密提取，主要针对虚拟卡券充值如：视讯类充值
调用方法:	/ExtractCard
方法:	POST 提交格式： Server/ExtractCard
输入参数:	红色参数代表需要进行签名加密的参数 appid orderid timestamp quantity productid sign
返回值:	JSON 格式 Code: 系统状态 Message: 状态描述 Data 卡密集合 Card 卡号 Passowrd 卡密 卡号卡密使用 AES 加密，使用密钥需要进行 AES 解密 在线工具 http://tool.oschina.net/encrypt/
举例	入参： { "appid": "1", "quantity": "1", "productid": "5003", "timestamp": "1528944783005", "orderid": "{{orderid}}", "sign": "9a9545c87463b952076ac1906f12d7e1" } 出参： { "Data": [{

	<pre>"Cardno": "oh1xlghVC29MP8RKlycuS9pBCmiuOVtR", "CardPwd": "oh1xlghVC29MP8RKlycuS9pBCmiuOVtR" }], "RawJson": null, "PostUrl": null, "State": 1000, "Message": "成功" }</pre>
--	---