



抽奖工坊

API 接入手册



目 录

封面

API 接入说明

API 简介

API 配置

API 导图 (抽奖机会)

API 导图 (积分抽奖)

通讯令牌 (token)

云端 API

云端 API 说明

获取用户登录标识

获取用户剩余抽奖次数

增减用户剩余抽奖机会

应用端 API

应用端 API 说明

连通性测试

查询用户积分

增减用户积分

中奖记录推送

发放 API 奖品

其他信息

API 接入示例包下载

技术支持

封面



API 接入说明

[API 简介](#)

[API 配置](#)

[API 导图（抽奖机会）](#)

[API 导图（积分抽奖）](#)

[通讯令牌（token）](#)

API 简介

API简介

抽奖工坊可以API用于与PC/手机网站、APP、游戏等应用端进行对接整合，可以实现的功能包括：

1. 使用应用端的用户身份参与抽奖活动，抽奖工坊可以同步头像、昵称和登录的用户身份。
2. 使用应用端的积分进行抽奖，比如100积分抽奖一次。
3. 将应用端的内容作为奖品并自动发放，比如用户积分、优惠券/现金券、游戏道具等。
4. 应用端自由设定条件给用户奖励抽奖机会，比如首次下载APP、消费满额等。
5. 通过API自动设置抽奖白名单，例如将订单号发送到白名单，让用户凭订单号抽奖。

API分类简介

为方便阐述API使用方便，本手册定义了几个名词，说明如下：

- 云端：指抽奖工坊的引擎
- 云端API：指抽奖工坊平台的开放API，比如要给用户增加抽奖机会，则需要应用端访问云端API的接口。
- 应用端：指用户的PC/收集网站、微信平台、APP、游戏等需要接入的应用。
- 应用端API：指由应用端开放的API，比如用户中奖后要自动发放奖品，则需要云端访问应用端的API通知发奖。

云端API是让开发者调用的，开发人员在需要的时候进行对应调用就可以。

应用端则需要开发者写好，在需要的时候，云端会进行调用。应用端API地址要在抽奖工坊后台设置填入，具体请看看手册中“API配置”章节。

API 配置

API配置

抽奖工坊默认使用独立模式进行活动。因此在进行API接入前，需要先进行API模式配置。
入口位置：抽奖工坊管理后台 -> 系统设置 -> 用户模式，如下图：

用户模式

☐ 游客登记模式

☐ 公众号模式

☒ API模式

API配置

使用API模式可与现有平台深度整合，点击查看详细说明。

云端API地址

https://s1.lezhi99.com/api

复制

API密钥(secret)

d3M2VfBL5BPd3cztDxVzhtzg2Evn1yn

复制

重置

接入端API地址

http://draw.lezhi99.net/site/api.php

未登录跳转URL

http://draw.lezhi99.net/site/index.php

若用户直接进入活动页（未登录），将跳转到此URL。

IP白名单

*

多个使用英文逗号间隔，设置后只有列表的IP才能调用抽奖API，填"*"则不限制

中奖记录推送

☐ 推送

☒ 不推送

积分抽奖模式

☒ 开启

☐ 关闭

积分名称

积分

抽奖一次消耗积分

10

以下是配置项的逐一说明：

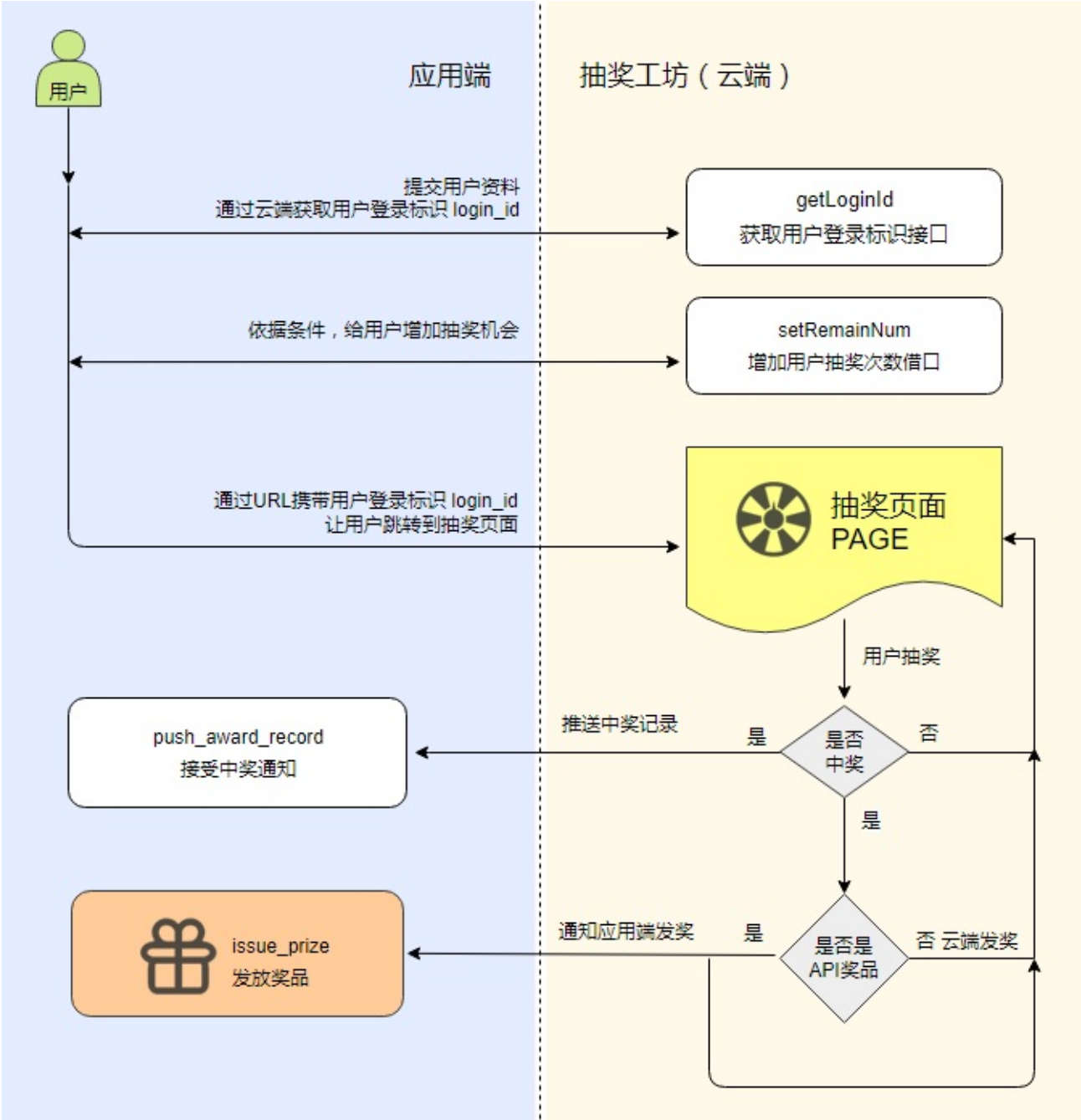
配置项	说明
用户模式	必须选择API模式，以开启API功能。
云端API地址	云端API的调用地址，不可编辑和修改。
API密钥(secret)	API通讯的密钥，用于生成通讯 token，可以点击后面的按钮进行重置。
应用端API地址	应用端的API地址，云端将通过此地址调用应用端AIP

未登录跳转URL	用户进入抽奖页面时候未携带login_id参数或login_id参数无效时，活动页面将跳转到此URL
IP白名单	多个使用英文逗号间隔，设置后只有列表的IP才能调用云端API，填 “*” 则不限制
中奖记录推送	是否将所有中奖信息（不管是否是API奖品）都推送到应用端，默认关闭
积分抽奖模式	是否开启积分模式，选择否则使用默认的抽奖机会模式。开启后可以设置积分名称和抽奖一次需消耗的积分数量

API 导图（抽奖机会）

抽奖机会模式API导图

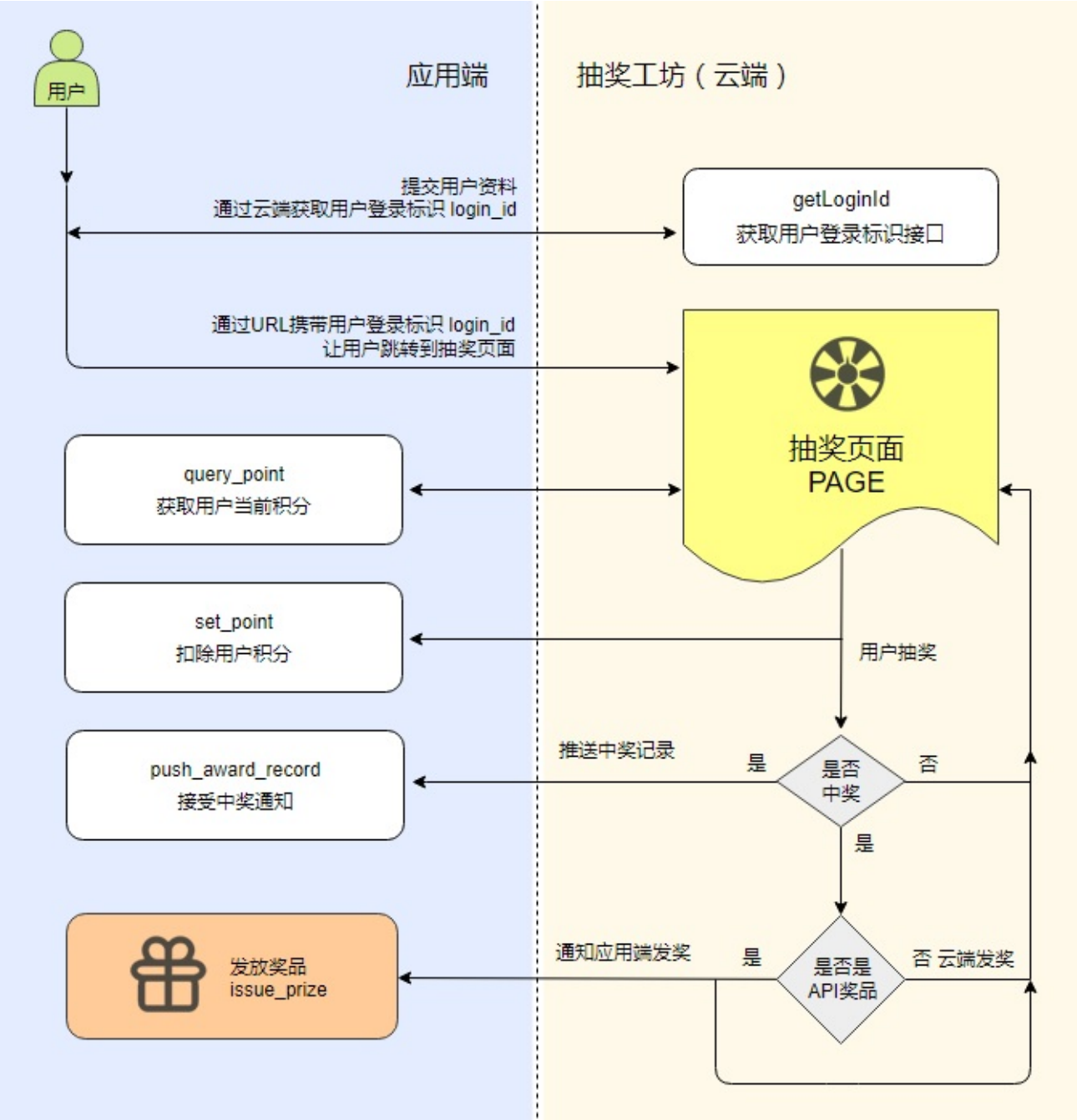
抽奖机会模式是指不使用积分抽奖，而是仅通过奖励抽奖机会的模式让用户参与抽奖。
例如：每日签到赠送抽奖机会、新用户赠送抽奖机会、充值满额赠送抽奖机会等



API 导图（积分抽奖）

积分抽奖模式API导图

积分抽奖模式是指直接消耗应用端用户积分，来参与抽奖。
例如：让用户消耗100金币抽奖一次。



通讯令牌 (token)

通讯令牌

不论是调用云端 API 还是应用端 API，都需要携带 token 参数，以此来检查每一次请求的合法性。避免因为 API 地址泄漏导致 API 被外部调用造成损失。

通讯令牌的生成方法

每一次 API 通讯的 token 参数都是不一样的，请使用以下步骤生成：

- 1、将 API 通讯中除 token 之外所有 post 参数按 键名升序 进行排序然后拼接值。
- 2、在第 1 步中拼接好的新字符串前面加上 API 密钥 (secret) 值，然后对此字符串进行 md5 加密。

提醒：API 密钥 (secret) 请在抽奖工坊管理平台的 用户模式->API 模式 中查看，请务必保护好 secret 值，不要泄漏。

【重要】云端在收到应用端的 api 请求后，首先会对 token 参数进行校验，校验通过才会进行相关操作并返回正确的信息。相同的，在云端调用应用端 API 的时候，也会同时传递 token 参数，应用端 API 请务必进行 token 校验。

生成过程详解/范例

例如客户端需要调用云端的 setRemainNum 接口，给 identifier 为 888 的用户增加 3 次抽奖机会。

一般调用时 POST 参数如下：

```
identifier : 888  
change : 3  
note : "新用户赠送"
```

如果不考虑 token 参数的话，请参数s就这样就可以了。

但是为了安全性，我们还需要加上 token 参数，否则的话，API 地址一旦泄漏，谁都可以调用这个 API 随意增加抽奖次数了。

根据 token 参数生成方法，我们将参数 根据键名升序，然后拼接字符串，接着在前面加上 secret 值。

本例中的拼接方式为：

```
secret + change + identifier + note
```

通讯令牌 (token)

假设 secret 值为 aaaaaaaaaaaaaaaaaa , 那么拼接结果就是 :

```
aaaaaaaaaaaaaaaaaaaa + ( 3 + 888 + 新用户赠送 ) = "aaaaaaaaaaaaaaaaaaaa3888新用户赠送"
```

然后对此字符串进行 md5 加密 , 获得最终 token 值

```
md5("aaaaaaaaaaaaaaaaaaaa3888新用户赠送") = "01087d87d5bbe9f14aacb1f8374ea662"
```

而将 token 带入 POST 请求参数后 , 最终 POST 参数为 :

```
identifier : 888
change : 3
note : "新用户赠送"
token : "01087d87d5bbe9f14aacb1f8374ea662"
```

PHP代码示例

token 生成函数 :

```
function gen_token($params,$secret){
    //让参数根据键名升序进行排序
    ksort($params,SORT_STRING);
    //拼接参数字符串, 并在前面加上 secret 值
    $str = $secret .implode(' ', $params);
    //对拼接后的字符串进行 MD5 加密, 并返回
    return md5($str);
}
```

调用片段 :

```
//.....
$secret= "aaaaaaaaaaaaaaaaaaaa";
//整理 POST 参数
$params = [
    'identifier' => 888,
    'change'     => 3,
    'note'       => "新用户赠送",
    'time'       => time(),
];
//附加 token
$params['token'] = gen_token($params,$secret);
//.....
```

云端 API

[云端 API 说明](#)

[获取用户登录标识](#)

[获取用户剩余抽奖次数](#)

[增减用户剩余抽奖机会](#)

云端 API 说明

云端 API 说明

云端 API 是指由抽奖工坊提供，供应用端调用的接口。

云端 API 基于 https 协议，通过 POST - application/x-www-form-urlencoded 方式传递数据。

获得云端 API 地址以及 secret

云端 API 地址以及 secret 可以在 抽奖工坊管理平台 -> 系统设置 -> 用户模式 -> API模式 中查看。

PS：关于时间戳 (Unix timestamp) time 参数的说明

每个云端 API 都会要求传入一个时间戳 (Unix timestamp) time 参数，每个 API 请求必须在这个时间后的 30 秒内送达，否则云端会判断超时。从而防止同一 API 请求信息被二次利用。同时，包含此参数可以在没有额外参数参与 token 生成的情况下利用 token 来破解 secret 信息。

获取用户登录标识

获取用户登录标识 user/getLoginId

此 API 是用于初始化用户信息并获取登录 ID (login_id)，用户在未进入抽奖页面之前，需要用此 API 将用户的信息注册到抽奖工坊，并且获取一个登录 ID (login_id)，此登录 ID 需要在进入活动页面时通过 URL 参数方式传递过去，实现以终端用户身份自动登录。

通过 login_id 实现用户登录到活动页的方法：

在通过 user/getLoginId 获得 login_id 后，引导用户跳转到活动页面URL。在活动页面的 URL 尾部加上字符串 “&login_id=xxxxxxxxx” 即可，其中 “xxxxxxxxxx” 是指实际获得的 login_id 的值。

例如您的活动入口 URL 是：

```
https://c1.lezhi99.com/?p=BUEj32HQU
```

那么追加 login_id 参数后的 URL 是：

```
https://c1.lezhi99.com/?p=BUEj32HQU&login_id=xxxxxxxxxx
```

应用端只需引导用户进入上面这个URL，在活动页面就会自动录用户账号。

提醒：每次获取的 login_id 都只有短暂的有效期，请不要将 login_id 保存在应用端，每次都应该重新获取。

API 地址

```
https://s1.lezhi99.com/user/getLoginId
```

API 参数列表&测试

```
post:https://s1.lezhi99.com/api/user/getLoginId
*int:project_id=10233#抽奖项目 ID
*string:identifier=23823#用户的唯一标识，例如用户 ID
*string:nickname=樱桃丸子#用户昵称/用户名等
*string:avatar=https://xxx.com/a.png#用户头像URL
*int:time=1576830778#时间戳(Unix timestamp)，精确到秒
*string:token=xxxxxxxxxx#通讯令牌 (token) 值
<<<
success
{
  "status": "success",
```

获取用户登录标识

```
    "data": {
      "login_id": "xxxxxxxxxxxxxxxxxxxxxxxxxxxx"
    }
  }
  <<<
  error
  {
    "status": "error",
    "msg": "Invalid token"
  }
}
```

返回

此API返回内容为JSON格式，包含如下信息：

参数名	类型	必须	说明
status	字符串	是	success=成功；error=失败/错误；
data	数据集	否	执行返回的数据集，当 status = success 的时候存在
- data.login_id	字符串	是	用户的登录 ID (login_id)
msg	字符串	否	提示信息，当 status=error 时存在

获取成功返回示例：

```
{
  "status": "success",
  "data": {
    "login_id": "xxxxxxxxxxxxxxxxxxxxxxxxxxxx"
  }
}
```

失败返回示例：

```
{
  "status": "error",
  "msg": "Invalid token"
}
```

获取用户剩余抽奖次数

获取用户剩余抽奖次数 user/queryRemainNum

此 API 用于获取用户在抽奖工坊中剩余的抽奖次数。

API 地址

<https://s1.lezhi99.com/user/queryRemainNum>

API 参数列表&测试

```
post:https://s1.lezhi99.com/api/user/queryRemainNum
*int:project_id=10233#抽奖项目 ID
*string:identifier=23823#用户的唯一标识, 例如用户 ID
*int:time=1576830778#时间戳 (Unix timestamp), 精确到秒
*string:token=xxxxxxxxxx#通讯令牌 (token) 值
<<<
success
{
  "status": "success",
  "data": {
    "remain_num": "23"
  }
}
<<<
error
{
  "status": "error",
  "msg": "Invalid token"
}
```

返回

此API返回内容为JSON格式，包含如下信息：

参数名	类型	必须	说明
status	字符串	是	success=成功； error=失败/错误；
data	数据集	否	执行返回的数据集， 当 status = success 的时候存在
			用户剩余的次数

获取用户剩余抽奖次数

- data.remain_num	整数	是	若为-1则表示获取失败（对应的用户未注册到抽奖工坊）
msg	字符串	否	提示信息，当status=error 时存在。

获取成功返回示例：

```
{
  "status": "success",
  "data": {
    "remain_num": "23"
  }
}
```

失败返回示例：

```
{
  "status": "error",
  "msg": "Invalid token"
}
```

增减用户剩余抽奖机会

增减用户剩余抽奖机会 user/setRemainNum

此 API 用于增加或减少用户在抽奖工坊中剩余的抽奖次数。

因为在给用户增加抽奖机会的时，有些时候抽奖工坊中并未有此用户的信息。因此此接口要求同事要传入用户的昵称和头像等信息。

API 地址

<https://s1.lezhi99.com/user/setRemainNum>

API 参数列表&测试

```
post:https://s1.lezhi99.com/api/user/setRemainNum
*int:project_id=10233#抽奖项目 ID
*string:identifier=23823#用户的唯一标识，例如用户 ID
*string:nickname=樱桃丸子#用户昵称/用户名等
string:avatar=https://xxx.com/a.png#用户头像URL，可留空
*int:change=1#要操作的次数。正数为增加，负数为减少
*string:note=系统赠送#备注信息
*int:time=1576830778#时间戳(Unix timestamp)，精确到秒
*string:token=xxxxxxxxxx#通讯令牌 (token) 值
<<<
success
{
  "status": "success",
  "data": {
    "remain_num": "23"
  }
}
<<<
error
{
  "status": "error",
  "msg": "Invalid token"
}
```

返回

此API返回内容为JSON格式，包含如下信息：

参数名	类型	必须	说明
			success=成功；

status	字符串	是	error=失败/错误；
data	数据集	否	执行返回的数据集， 当 status = success 的时候存在
- data.remain_num	整数	是	用户新的剩余次数
msg	字符串	否	提示信息，当 status=error 时存在

获取成功返回示例：

```
{
  "status": "success",
  "data": {
    "remain_num": "23"
  }
}
```

失败返回示例：

```
{
  "status": "error",
  "msg": "Invalid token"
}
```

应用端 API

[应用端 API 说明](#)

[连通性测试](#)

[查询用户积分](#)

[增减用户积分](#)

[中奖记录推送](#)

[发放 API 奖品](#)

应用端 API 说明

应用端 API 说明

应用端 API 是指由应用端提供，供抽奖工坊调用的接口。

抽奖工坊通过 http POST - application/x-www-form-urlencoded 方式，将相关数据推送给应用端。

设置应用端 API 地址，以及获得生成 token 的 secret

开发者必须将应用端的 API 地址设置到抽奖工坊中，这样在需要调用应用端的 API 的时候，抽奖工坊才知道要通过哪个网址。请在 抽奖工坊管理平台 -> 系统设置 -> 用户模式 -> API模式 中设置应用端的API的地址。

同时应用端也需要校验 token 的有效性，使用的 secret 也从上述设置页面中获取。

PS：抽奖工坊后台设置中只能设置一个应用端 API 地址，但是 API 是有多个的，怎么处理？

抽奖工坊调用应用端 API 的时候，使用统一的地址。并通过 POST 参数 api 的值，来区分不同的请求。

例如在 联通测试 中，POST 参数的 api 将传入 ping，而在发放奖品的时候 api 将传入 issue_prize。

开发者必须根据传入的 api 参数来进行不同的操作和进行响应。

PS：关于时间戳 (Unix timestamp) time 参数的说明

云端往应用端的每个 API 请求都会包含一个时间戳 (Unix timestamp) time 参数，建议应用端对这个时间戳进行检查。比如请求到达时间与这个时间戳相差超过 30 秒就拒绝（可返回请求超时提示），从而防止已用过的 API 请求信息被二次利用。同时包含此参数，还可以防止在没有额外参数参与 token 生成的情况下 token 用来破解 secret 信息。

连通性测试

连通性测试 / ping

在抽奖工坊管理平台后台首页，若开启了 API 模式，则会有一个 API 状态的面板会显示 API 可用性以及速度。此 API 即用于测试应用端的连通性。

API 地址

此 API 由云端往您的应用端 API 地址发送请求

API 将传入参数

参数名	类型	必须	示例	说明
api	字符串	是	ping	API名称，此处固定传入 ping
time	整数	是	1576830778	时间戳 (Unix timestamp)，精确到秒
token	字符串	是	xxxxxxxxxxx	访问令牌 token

返回

请返 JSON 格式的内容，包含如下信息：

通讯成功请返回：

```
{ "status": "success" }
```

失败返回：

```
{ "status": "error", "msg": "这里写失败说明" }
```

PHP 代码片段示例

```
<?php
.....
//连通性测试
case 'ping':
```

```
exit('{"state":"success"}');  
.....  
?>
```

查询用户积分

查询用户积分 / query_point

若开启了 API 积分模式，那么当用户进入活动页面的时候，需要在页面上显示用户的当前积分。抽奖工坊即用此 API 查询用户的剩余积分。

API 地址

此 API 由云端往您的应用端 API 地址发送请求

API 将传入参数

参数名	类型	必须	示例	说明
api	字符串	是	query_point	API名称，此处固定传入 query_point
user_identifier	字符串	是	10001	用户唯一标识
time	整数	是	1576830778	时间戳 (Unix timestamp)，精确到秒
token	字符串	是	xxxxxxxxxxx	访问令牌 token

返回

通讯成功请返回以下 JSON ，其中 point 指用户剩余积分数额。

```
{"status": "success", "point": 2000}
```

失败返回以下 JSON :

```
{"status":"error","msg":"这里写失败说明"}
```

PHP 代码片段示例

```
<?php
.....
//查询积分
case 'query_point':
    //获得UID
    $uid = (int)$_POST['user_identifier'];
    //获得用户积分
    $point= DB("user")->where("uid",$uid)->getValue("point");
    //获取成功, 返回积分信息
    if($point!==false){
        exit({'status':"success","point":'.'.$point.''});
    }else{
        //如果获取失败, 返回错误信息
        exit('{"state":"error","msg":"不存在此用户"}');
    }
.....
?>
```

增减用户积分

查询用户积分 / set_point

若开启了 API 积分模式，在用户进行抽奖的时候，抽奖工坊将通过此 API 通知用户扣除积分，若抽奖失败（例如奖品不足了），则抽奖工坊会通过此 API 通知应用端返还用户积分。

API 地址

此 API 由云端往您的应用端 API 地址发送请求

API 将传入参数

参数名	类型	必须	示例	说明
api	字符串	是	set_point	API名称，此处固定传入 set_point
user_identifier	字符串	是	10001	用户唯一标识
change	整数	是	100 / -100	需要操作的分数，正数表示增加积分，负数标识扣除积分
time	整数	是	1576830778	时间戳 (Unix timestamp)，精确到秒
token	字符串	是	xxxxxxxxxxx	访问令牌 token

返回

通讯成功请返回以下 JSON：

```
{"status": "success"}
```

失败返回以下 JSON：

```
{"status": "error", "msg": "这里写失败说明"}
```

PHP 代码片段示例

```
<?php
.....
//增减积分
case 'set_point':
    //获得参数
    $uid = (int)$_POST['user_identifier'];           //获得UID
    $change = (int)$_POST['change'];                 //操作的分数，负数为扣除，
正数为增加
    //增减用户积分
    $setPoint=DB('user')->where("uid",$uid)->setInc('point',$change);
    //如果增减成功
    if($setPoint===true){
        exit({'status':"success"}');
    }else{
        //获取成功，返回积分信息
        exit({'state':"error","msg":"积分操作失败"}');
    }
    .....
?>
```

中奖记录推送

中奖记录推送 / push_award_record

若在 API 配置中开启了中奖记录推送，则用户的每次中奖，不管是否是 API 类型的奖品，都会进行推送。主要用于获取中奖记录，例如要在应用端展示中奖信息等。

注意事项：此 API 仅作为中奖记录推送，不能作为 API 奖品发放使用。抽奖工坊将中奖信息进推送到此 API 后，不会理会应用端是否有返回成功信息。处理 API 奖品发放，见“发放 API 奖品”接口说明。

API 地址

此 API 由云端往您的应用端 API 地址发送请求

API 将传入参数

参数名	类型	必须	示例	说明
api	字符串	是	push_award_record	API名称，此处固定传入 push_awa rd_record
user_identifier	字符串	是	10001	用户唯一标识
prize_name	字符串	是	100 积分	奖品名称
award_date	字符串	是	2019-12-20 22:06:03	中奖时间
time	整数	是	1576830778	时间戳 (Unix timestam p)，精确到秒
token	字符串	是	xxxxxxxxxx	访问令牌 token

返回

此 API 不用返回信息。云端只负责推送过去，对应用端是否接受和处理不做反应。

发放 API 奖品

发放 API 奖品 / issue_prize

抽奖系统支持多种奖品类型，对于实物奖品/卡密等奖品可以由平台内部发放，但是接入 API 后，将支持 API 奖品。API 奖品是指由应用端自动发放的，比如抵用券、金币、游戏道具等。在用户抽中 API 奖品后，抽奖工坊将通过此 API 通知应用端进行奖品发放。

API 地址

此 API 由云端往您的应用端 API 地址发送请求

API 将传入参数

参数名	类型	必须	示例	说明
api	字符串	是	issue_prize	API 名称，此处固定传入 issue_prize
record_id	整数	是	20301	中奖纪录的唯一ID
prize_identifier	字符串	是	gold_88	奖品标识，详解表格下方说明
user_identifier	字符串	是	10001	用户唯一标识
				用户

exchange_info	JSON字符串	否	{"手机":"18012345678"}	的兑奖信息，详见下方表格说明
time	整数	是	1576830778	时间戳 (Unix timestamp)，精确到秒
token	字符串	是	xxxxxxxxxx	访问令牌 token

参数详细说明

- record_id：是中奖记录的唯一 ID。有时候因为网络或程序错误导致 API 没有正确返回成功信息。抽奖工坊可能会二次通知 API 发放奖品。这种情况，就有可能造成重复发放奖品。因此在发放奖品的时候，可以记录这个唯一 ID，在发放奖品的时候检查一下对于的 ID 是否已经发放过。若发放过了，就直接返回 success 标识，避免重复发放奖品。
- prize_identifier：这个参数是奖品指的标识。在抽奖工坊添加 API 奖品的时候，除了奖品名称外，还需要填写一个奖品标识，即用于此处传送给应用端 API。这个标识可以让我们更灵活的设置奖项。比如一个金币的奖品，可以设置这样一个标识：gold_xx，后面的 xx 表示金币的数量。例如 gold_88 表示 88 个金币，gold_120 表示 120 个金币。这样就可以在抽奖平台设置任意数额的金币奖品了。
- exchange_info：对于一些特殊的 API 奖品，有可能需要让中奖用户填一些兑奖的表达。此参数将回传用户填写的信息。比如电话号码、游戏账号等。

返回

通讯成功请返如以下 JSON，其中 msg 中的信息会原样显示给用户：

```
{"status": "success", "msg": "优惠券已发放至您的账户，请注意查收。"}
```

失败返回以下 JSON：

```
{"status":"error","msg":"系统错误，请稍后再试！"}
```

PHP 代码片段示例

```
<?php
.....
//发放奖品
case 'issue_prize':
    //获得参数
    $uid                      = (int)$_POST["user_identifier"];           //获得中
    奖用户UID
    $prize_identifier         = $_POST["prize_identifier"];               //奖品表示

    //根据奖品标识获得奖品信息
    $tempArr = explode("_",$prizeIdentifier);
    $type     = $tempArr[0];                                             //奖品类型

    $num       = (int)$tempArr[1];                                       //数量
    //在实际应用中，可以根据奖品的类型不同分别执行不同的奖品发放操作，例如发放道具、开通
    VIP权限等
    switch($type){
        case "gold":           //金币
            DB("user")->where("uid",$uid)->setInc("gold",$num);
            //返回提示
            exit('{"state":"success","msg":"金币已发放至您的账户，请注意查收。"}');
        );
            break;
        case "coupon":         //抵用券
            DB("user")->where("uid",$uid)->setInc("coupon",$num);
            //返回提示
            exit('{"state":"cuccess","msg":"{$num}抵用券以发放至您的账户，请注意
            查收。"}');
            break;
        default:
            $this->echo_json('error','参数错误，所兑换的奖品无效。');
            break;
    }
    .....
?>
```

其他信息

[API 接入示例包下载](#)

[技术支持](#)

API 接入示例包下载

API 接入示例包下载

正在整理中.....

技术支持

技术支持

如果您在使用 API 接入应用端时碰到不明白的地方，请不要着急。马上和我们联系，我们将为您提供协助。

- 联系QQ：1060443360
- E-mail：lezhi99@qq.com
- 用户社区：<https://bbs.lezhi99.com/>