

PPk Open API

版本 **0.5**

修订历史

日期	版本	描述	作者
2012-03-10	0.1		陈晖
2012-06-20	0.2		陈晖
2012-11-27	0.3	增加活动类接口定义	陈晖
2013-01-24	0.4	修订活动相关接口，按 Activity, Meet, Connect 三个要素重新定义	陈晖
2013-01-25	0.5	增加“赞”接口	陈晖

目录

1 介绍.....	6
2 接口定义.....	6
2.1 基础接口.....	6
2.1.1 前端向 PPK 平台申请唯一标识.....	6
2.1.2 前端以 AJAX 回调模式向 PPK 平台请求访问.....	7
2.2 用户注册/登录/资料管理相关接口.....	8
2.2.1 前端快速注册新用户.....	8
2.2.2 前端登录 PPK 平台.....	9
2.2.3 前端退出 PPK 平台.....	10
2.2.4 前端获取用户注册信息.....	11
2.2.5 前端更新用户注册信息.....	12
2.2.6 前端更新用户 OAUTH 关联登录信息.....	13
2.3 PIN 类对象处理接口.....	13
2.3.1 批量提交多个新的 PIN 类对象数据.....	14
2.3.2 获取指定 PIN 类对象数据.....	15
2.3.3 更新指定 PIN 类对象数据.....	16
2.3.4 删除指定 PIN 类对象数据.....	17
2.3.5 前端获取随机返回的两个比拼对象数据.....	18
2.3.6 前端上报比拼胜出者.....	19
2.4 第三方帐户关联接口.....	20
2.4.1 获取平台当前支持关联的第三方拼站列表.....	20
2.4.2 前端获取用户关联第三方拼站帐号.....	21
2.4.3 前端新增用户关联的第三方拼站帐号.....	22
2.4.4 前端更新用户关联的第三方拼站帐号信息.....	23
2.4.5 前端删除用户关联的第三方拼站帐号信息（尚未实现）.....	24
2.5 信息聚合工具接口.....	25
2.5.1 获取用户拥有的画板列表.....	25
2.5.2 提交新的聚合记录数据.....	26
2.5.3 获取单条聚合记录的详细信息.....	28
2.5.4 更新单条聚合记录数据.....	32
2.5.5 添加图片等来源数据到指定聚合记录.....	34
2.5.6 删除单条聚合记录数据.....	36
2.5.7 提交查询聚合记录数据请求.....	36
2.5.8 获取指定查询聚合记录结果集数据.....	39
2.6 活动(Activity)组织接口.....	41
2.6.1 创建新的活动记录.....	41
2.6.2 获得指定活动记录详细信息.....	42
2.6.3 修改指定活动记录.....	44
2.6.4 删除指定活动记录.....	45
2.7 活动碰(Meet)接口.....	46
2.7.1 提交新的“碰”.....	46
2.7.2 修改指定“碰”.....	47
2.7.3 删除指定“碰”.....	49

2.7.4 获得指定“碰”详细信息.....	49
2.7.5 获得指定活动的相关"碰"列表.....	51
2.8 活动关联(Connect)内容接口.....	53
2.8.1 关联新的内容.....	53
2.8.2 删除指定关联.....	54
2.9 赞(Like)接口.....	55
2.9.1 “赞”指定活动的关联内容.....	55
2.9.2 删除对指定关联内容的“赞”.....	56
2.9.3 “赞”指定 PIN 类对象.....	56
2.9.4 删除对指定 PIN 类对象的“赞”.....	57
2.10 活动动态聚合(Activity Feed)接口.....	58
2.10.1 获取当前用户所参与指定或全部活动的最新动态聚合.....	58
2.11 查询(Search)接口.....	59
2.11.1 提交查询 PIN 类内容数据请求.....	59
2.11.2 提交查询活动记录数据请求.....	61
2.11.3 提交查询关联关系的请求.....	63
2.11.4 获取指定查询结果集数据.....	65
2.11.5 刷新指定查询结果集数据（尚未实现）.....	66
3 接口调用开发示例.....	67
3.1 JQuery AJAX.....	67
3.2 PHP.....	68

1 介绍

本文描述了 PPK 基于 WEB 平台向用户使用的客户端提供的接口协议和功能。

2 接口定义

2.1 基础接口

2.1.1 前端向 PPK 平台申请唯一标识

前端从 PPK 平台获得唯一标识 ID，此标识长期有效，前端可以自行保存重复使用。

URL: `Http://PPK_API_HOST/api/v1/tool/gen_front_id/`

请求方式：API POST

输入：

HTTP URL Fields：

无

HTTP Header Cookie Fields：

无

HTTP Post Form Fields：

`front_type`：前端环境类型，取值定义见《PPK_API_Common_Define》

`product_modal`：设备型号，取值定义见《PPK_API_Common_Define》

`product_sn`：产品本身的标识/序列号（如苹果设备的唯一硬件 ID），取值可为空，但建议提供，有助于更有效地识别前端

`screen_width`：设备屏幕宽度，可以为空

`screen_height`：设备屏幕高度，可以为空

`app_id`：应用的唯一标识

`app_token`：应用的认证码

正常输出：

HTTP Header：无

HTTP Content :

resp_status : OK

resp_content:

front_id: 前端标识 字符串

2.1.2 前端以 **AJAX** 回调模式向 **PPK** 平台请求访问

前端可采用 AJAX 的 JSONP 回调协议直接向 PPK 平台发出访问请求，以规避 AJAX 存在的域安全限制。

URL:

Http://PPK_API_HOST/api/v1/cross/?req=***ENCODED_REQUEST_JSON_STRING***&callback=?

请求方式：API GET-JSONP

输入：

HTTP URL Fields :

ENCODED_REQUEST_JSON_STRING: 经 escape 编码后的实际请求参数数组对应 JSON 字符串，以调用“关联新的第三方网站帐户”接口为例，该字符串取值示例如下：

```
{
  "method":"POST",
  "resource":"front/ADKKKDEE/user/123456/ext_account/",
  "cookie":{
    "uuid":"123456",
    "access_token":"FDSFWWWd#WWW"
  },
  "post":{
    "pinhost_id":"18",
    "login_id":"tester@test.com",
    "login_pwd":"123456789"
  }
}
```

HTTP Header Cookie Fields :

无

正常输出：

HTTP Header ：无

HTTP Content：

对应接口应答的结果（JSON 格式）

2.2 用户注册/登录/资料管理相关接口

2.2.1 前端快速注册新用户

前端向 PPK 平台快速注册新用户。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/register/

请求方式：API POST

输入：

HTTP URL Fields：

FRONT_ID ：前端环境唯一标识

HTTP Header Cookie Fields：

无

HTTP Post Form Fields：

nick_name：用户昵称 字符串

对于使用 EMAIL 注册新用户时：

email：有效的电子邮箱，可以在后续作为登录名使用 字符串

mphone：有效的移动通讯号码，可以在后续作为登录名使用 字符串

encoded_pwd：经过编码的用户密码 字符串

对于使用第三方平台帐号进行关联绑定注册新用户时：

oauth_site: 第三方平台域名

oauth_uid : 第三方平台帐号用户标识

oauth_token: 关联绑定产生的唯一识别令牌, 由提交注册请求的应用根据关联绑定信息自行生成, 需满足确保用户下一次使用第三方帐户登录时能生成一致的 token 用于登录鉴权。

正常输出 :

HTTP Header : 无

HTTP Content :

resp_status : OK

resp_content:

uuid : 用户 ID 数字

access_token: 前端访问 PPK 平台的令牌 16 位字符串

nick_name: 用户昵称 字符串

2.2.2 前端登录 PPK 平台

前端登陆 PPK 平台, 获得由后台生成的访问令牌。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/login/

请求方式 : API POST

输入 :

HTTP URL Fields :

FRONT_ID : 前端环境唯一标识

HTTP Header Cookie Fields :

无

HTTP Post Form Fields :

对于使用用户 ID, 注册时登记的联系电话或电子邮箱登录时:

uuid_mphone_or_email : PPK 上的用户 ID, 注册时登记的联系电话或电子邮箱
数字

encoded_pwd：经过编码的用户密码 字符串

对于使用第三方平台帐号进行关联登录时：

oauth_site: 第三方平台域名

oauth_uid：第三方平台帐号用户标识

oauth_token: 关联绑定产生的唯一识别令牌，由提交注册请求的应用根据关联绑定信息自行生成，需与用户第一次使用第三方帐户关联注册登录时生成的 token 一致。

正常输出：

HTTP Header：无

HTTP Content：

resp_status：OK

resp_content:

uuid: 用户 ID

access_token: 前端访问 PPK 平台的令牌 16 位字符串

nick_name: 用户昵称 字符串

2.2.3 前端退出 PPK 平台

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/login/

请求方式：API DELETE

输入：

HTTP URL Fields：

FRONT_ID：前端环境唯一标识

HTTP Header Cookie Fields：

uuid: 当前登录用户 ID

access_token: 前端访问 PPK 平台的令牌

HTTP Post Form Fields：无

正常输出：

HTTP Header：无

HTTP Content :

resp_status : OK
resp_content: 无

2.2.4 前端获取用户注册信息

前端向 PPK 平台查询指定用户（当前登录用户或该用户的好友）的注册信息。

URL: Http://PPK_API_HOST/api/v1/front/***FRONT_ID***/user/***DEST_UUID***

请求方式：API GET

输入：

HTTP URL Fields :

FRONT_ID : 前端环境唯一标识

DEST_UUID: 对象用户 ID

HTTP Header Cookie Fields :

uuid: 当前登录用户 ID

access_token: 已授权令牌

HTTP Post Form Fields :

无

正常输出：

HTTP Header : 无

HTTP Content :

resp_status : OK
resp_content:
 uuid : 用户 ID 数字
 nick_name : 用户昵称 字符串
 email : 注册的电子邮箱 字符串
 mphone : 注册的移动通讯号码，可能为空 字符串
 user_level: 用户级别

2.2.5 前端更新用户注册信息

前端向 PPK 平台更新指定用户（需与当前登录用户一致）的注册信息。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/user/**DEST_UUID**

请求方式：API POST

输入：

HTTP URL Fields：

FRONT_ID：前端环境唯一标识

DEST_UUID：对象用户 ID

HTTP Header Cookie Fields：

uuid：当前登录用户 ID

access_token：已授权令牌

HTTP Post Form Fields：

nick_name：用户昵称 字符串

email：有效的电子邮箱，可以在后续作为登录名使用 字符串

mphone：有效的移动通讯号码，可以在后续作为登录名使用 字符串

encoded_pwd：经过编码的用户原密码，如果为空，则表示不修改密码 字符串

new_encoded_pwd：经过编码的用户新密码 字符串

正常输出：

HTTP Header：无

HTTP Content：

resp_status：OK

resp_content:

uuid：用户 ID 数字

2.2.6 前端更新用户 OAUTH 关联登录信息

前端向 PPK 平台更新指定用户（需为当前登录用户）的 OAUTH 关联登录信息。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/user/**DEST_UUID**/oauth_bind/

请求方式：API POST

输入：

HTTP URL Fields：

FRONT_ID：前端环境唯一标识

DEST_UUID：对象用户 ID

HTTP Header Cookie Fields：

uuid：当前登录用户 ID

access_token：已授权令牌

HTTP Post Form Fields：

oauth_site：第三方平台域名

oauth_uid：第三方平台帐号用户标识

oauth_token：关联绑定产生的唯一识别令牌，由提交注册请求的应用根据关联绑定信息自行生成，需满足确保用户下一次使用第三方帐户登录时能生成一致的 token 用于登录鉴权。

正常输出：

HTTP Header：无

HTTP Content：

resp_status：OK

2.3 PIN 类对象处理接口

2.3.1 批量提交多个新的 PIN 类对象数据

PPK 平台批量提交多个新的 PIN 对象数据，申请被系统自动预处理通过后获得标识编号。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/pinobj/

请求方式：API POST

HTTP URL Fields：

FRONT_ID：前端环境唯一标识

输入：

HTTP Header Cookie Fields：

uuid: 当前登录用户 ID

access_token: 已授权令牌

HTTP Content Post Fields：

new_record_num：此次提交对象数据记录数，整形

new_record_list：对象数据数组，采用 JSON 格式，字符串

数组元素下标按 0、1、2 依次编号，每个元素的数据块定义如下：

pinobj_type：对象类型，取值定义见《PPK_API_Common_Define》

pinobj_url：PIN 类对象资源网址，字符串

pinobj_title：内容标题，字符串，可以“#标签 1#标签 2#”的形式指定
关键字标签

content_type：资源的内容格式，字符串，可以为空，取值定义见
《PPK_API_Common_Define》

snap_url_default：缺省建议使用的快照预览图像 URL

from_page_url: 来源网页网址，如该资源不是从网页中采集中的（比如从
本地上传），则该字段不用填写

正常输出：

HTTP Header：无

HTTP Content :

resp_status : OK

resp_data:

accepted_record_num : 被系统接受的数组记录数, 整形

accepted_record_list : 被系统接受的数组记录数组

数组元素下标按 0、1、2 依次编号, 每个元素的数据块定义如下:

pinobj_type : 系统预处理通过的对象类型, 取值定义见
《PPK_API_Common_Define》

pinobj_id : 对象唯一标识, 字符串, 长度为 32 个 ASCII 字符

pinobj_status : 数据记录的状态, 取值定义见
《PPK_API_Common_Define》

pinobj_url : PIN 类对象所在网址, 字符串

failed_record_num : 被系统拒绝的数组记录数, 整形

failed_record_list : 被系统拒绝的数组记录数组

数组元素下标按 0、1、2 依次编号, 每个元素的数据块定义如下:

failed_reason : 出错原因描述, 字符串

pinobj_url : PIN 类对象所在网址, 字符串

2.3.2 获取指定 PIN 类对象数据

向 PPK 平台查询指定 PIN 类对象数据。

URL:

Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/pinobj/**PPK_OBJECT_TYPE**/**PPK_OBJ_ID**/

请求方式: API GET

HTTP URL Fields :

FRONT_ID : 前端环境唯一标识

PPK_OBJECT_TYPE : 指定对象类型, 取值定义见《PPK_API_Common_Define》

PPK_OBJ_ID : 对象标识

输入：

HTTP Header Cookie Fields：

uuid: 当前登录用户 ID

access_token: 已授权令牌

正常输出：

HTTP Header：无

HTTP Content：

resp_status: OK

resp_data:

pinobj_type：系统认可的对象类型，取值定义见
《PPK_API_Common_Define》

pinobj_status：数据记录的状态，取值定义见
《PPK_API_Common_Define》

pinobj_id：PIN 类对象标识，字符串

pinobj_url：所在网址，字符串

pinobj_title：名称，字符串

content_type: 内容格式，字符串，取值定义见

《PPK_API_Common_Define》

snap_url_default：缺省建议使用的快照图像 URL

record_utc：数据录入时间戳，取值定义见《PPK_API_Common_Define》

2.3.3 更新指定 PIN 类对象数据

向 PPK 平台提交保存发生修改的对象数据。

URL: Http://PPK_API_HOST/api/v1/front/***FRONT_ID***/pinobj/***PINOBJ_TYPE***/***PINOBJ_ID***/

请求方式：API POST

HTTP URL Fields：

FRONT_ID：前端环境唯一标识

PINOBJ_TYPE：指定对象类型，取值定义见《PPK_API_Common_Define》

PINOBJ_ID：对象标识

输入：

HTTP Header Cookie Fields :

uuid: 当前登录用户 ID

access_token: 已授权令牌

HTTP Content Post Fields :

pinobj_url : PIN 类对象所在网址, 字符串, 可以为空, 表示不修改

pinobj_title : 名称, 字符串, 可以为空, 表示不修改。可以 “#标签 1#标签 2#” 的形式指定关键字标签

content_type: 内容格式, 字符串, 可以为空, 表示不修改。取值定义见《PPK_API_Common_Define》

snap_url_default : 缺省使用的快照图像 URL, 可以为空, 表示不修改

from_page_url: 来源网页网址, 可以为空, 表示不修改

正常输出:

HTTP Header : 无

HTTP Content :

resp_status : OK

resp_data:

pinobj_type : 系统预处理通过的对象类型, 取值定义见《PPK_API_Common_Define》

pinobj_id : 对象唯一标识, 字符串

pinobj_status : 数据记录的状态, 取值定义见《PPK_API_Common_Define》

2.3.4 删除指定 PIN 类对象数据

向 PPK 平台提交删除指定对象数据。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/pinobj/**PINOBJ_TYPE**/**PINOBJ_ID**/

请求方式: API DELETE

HTTP URL Fields :

FRONT_ID : 前端环境唯一标识

PINOBJ_TYPE : 指定对象类型, 取值定义见《PPK_API_Common_Define》

PINOBJ_ID : 对象标识

输入：

HTTP Header Cookie Fields：

uuid: 当前登录用户 ID

access_token: 已授权令牌

正常输出：

HTTP Header：无

HTTP Content：

resp_status: OK

resp_data:
无

2.3.5 前端获取随机返回的两个比拼对象数据

前端访问 PPK 平台，获取随机返回的两个指定类型的比拼对象数据。

URL: Http://PPK_API_HOST/api/v1/front/***FRONT_ID***/pk/***PINOBJ_TYPE***

请求方式：API POST

HTTP URL Fields：

FRONT_ID：前端环境唯一标识

PINOBJ_TYPE：指定对象类型，取值定义见《PPK_API_Common_Define》。目前只支持 PINBOARD 取值。

输入：

HTTP Header Cookie Fields：

uuid: 用户 ID

access_token: 前端访问 PPK 平台的已登录验证令牌

正常输出：

HTTP Header：无

HTTP Content：

resp_status: OK

resp_data:

- pk_task_id : 返回新生成的比拼任务号
- pinobj_type : 返回参与比拼的两个对象的相同类型，取值定义见《PPK_API_Common_Define》

pinobj_list: 对象数据列表数组，两个元素按 0, 1 排序

- pinobj_id : 对象标识，字符串
- pinobj_url : 所在网址，字符串
- pinobj_title : 名称，字符串
- snap_url_default : 缺省建议使用的快照图像 URL
- snap_url_pref : 更多尺寸快照图像 URL 前缀，保留字段
- tags_str : 相关的主题关键字字符串，多个关键字间以英文逗号分隔
- record_utc : 数据录入时间戳，取值定义见《PPK_API_Common_Define》
- win_counter : PK 获胜次数，数字
- lost_counter : PK 失败次数，数字
- total_counter : PK 总次数，数字
- rank_val : 等级分，数字
- stat_update_utc : 数据更新时间戳，取值定义见《PPK_API_Common_Define》

2.3.6 前端上报比拼胜出者

前端访问 PPK 平台，上报已生成和展示的比拼记录中用户选择喜欢的一个。

URL:

Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/pk/**PINOBJ_TYPE**/**PK_TASK_ID**/**PINOBJ_ID**
/

请求方式：API POST

HTTP URL Fields :

FRONT_ID : 前端环境唯一标识
PINOBJ_TYPE : 指定对象类型，取值定义见《PPK_API_Common_Define》
PK_TASK_ID : 比拼任务编号
PINOBJ_ID : 比拼胜出的对象标识

输入：

HTTP Header Cookie Fields :

uuid: 用户 ID

access_token: 前端访问 PPK 平台的已登录验证令牌

正常输出：

HTTP Header ：无

HTTP Content：

resp_status : OK

resp_data:
无

2.4 第三方帐户关联接口

2.4.1 获取平台当前支持关联的第三方拼站列表

向 PPK 平台查询当前支持关联的第三方拼站列表。

URL: Http://PPK_API_HOST/api/v1/common/pinhost/linkable/

请求方式：API GET

HTTP URL Fields：

无

输入：

HTTP Header Cookie Fields：

uuid: 用户 ID，取值为空时表示匿名访问

access_token: 已授权令牌，取值为空时表示匿名访问

正常输出：

HTTP Header ：无

HTTP Content：

```

resp_status : OK
resp_data:
    result_num : 查询结果中的记录总条数
    result_list: 拼站列表数组，元素按 0，1，……，N 排序
        pinobj_id : 拼站编号，字符串
        pinobj_title : 名称，字符串
        pinobj_url : 网址，字符串

```

2.4.2 前端获取用户关联第三方拼站帐号

前端向 PPK 平台获取当前登录用户所关联的第三方拼站用户帐号列表信息。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/user/**UUID**/ext_account/

请求方式：API GET

输入：

HTTP URL Fields：

FRONT_ID : 前端环境唯一标识
UUID: 用户 ID

HTTP Header Cookie Fields：

access_token: 已授权令牌

HTTP Post Form Fields：

无

正常输出：

HTTP Header : 无

HTTP Content：

```

resp_status : OK
resp_content:
    result_num : 关联第三方帐号数量      数字
    ext_account_list : 数组，每个元素的格式定义如下：
        ext_accound_id: 关联帐号记录编号

```

status: 帐号关联状态,取值定义见《PPK_API_Common_Define》
last_error: 如果帐号关联状态为出错异常,则该字段会有相应的错误描述
pinhost_id: 第三方拼站的编号
pinhost_title: 第三方拼站的名称
label: 名称
url 用户所在第三方拼站的用户帐号公开首页
login_id: 用户在第三方拼站的登录使用的用户名
nick_name: 用户在第三方拼站使用的昵称

2.4.3 前端新增用户关联的第三方拼站帐号

前端向 PPK 平台登记增加当前登录用户所关联的第三方用户帐号记录。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/user/**UUID**/ext_account/

请求方式: API POST

输入:

HTTP URL Fields:

FRONT_ID: 前端环境唯一标识

UUID: 用户 ID

HTTP Header Cookie Fields:

access_token: 已授权令牌

HTTP Post Form Fields:

label: 名称

pinhost_id: 第三方拼站的编号:

login_id: 用户在第三方拼站的登录使用的用户名

login_pwd: 用户在第三方拼站的登录使用的用户密码

正常输出:

HTTP Header : 无

HTTP Content:

resp_status : OK
 resp_content:
 ext_accound_id : 新关联的帐号记录编号 数字
 status : 新 关 联 的 帐 号 记 录 状 态 , 取 值 定 义 见
 《PPK_API_Common_Define》

2.4.4 前端更新用户关联的第三方拼站帐号信息

前端向 PPK 平台请求修改当前登录用户所关联的指定第三方用户帐号记录。

URL:

Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/user/**UUID**/ext_account/**EXT_ACCOUNT_ID**

请求方式：API POST

输入：

HTTP URL Fields：

FRONT_ID : 前端环境唯一标识

UUID: 用户 ID

EXT_ACCOUNT_ID : 关联记录标识

HTTP Header Cookie Fields：

access_token: 已授权令牌

HTTP Post Form Fields：

label: 名称

pinhost_id: 第三方拼站的编号:

login_id : 用户在第三方拼站的登录使用的用户名

login_pwd : 用户在第三方拼站的登录使用的用户密码，可以为空，表示不修

改

status: 目前前端只能取值为-4,-9 或为空，定义见《PPK_API_Common_Define》

正常输出：

HTTP Header : 无

HTTP Content :

resp_status : OK

resp_content:

ext_accound_id: 关联帐号记录编号

status : 帐号关联状态 , 取值定义见
《PPK_API_Common_Define》

2.4.5 前端删除用户关联的第三方拼站帐号信息 (尚未实现)

前端向 PPK 平台请求删除当前登录用户所关联的指定第三方用户帐号记录。

URL:

Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/user/**UUID**/ext_account/**EXT_ACCOUNT_ID**/

请求方式: API DELETE

输入:

HTTP URL Fields :

FRONT_ID : 前端环境唯一标识

UUID: 用户 ID

EXT_ACCOUNT_ID : 关联记录标识

HTTP Header Cookie Fields :

access_token: 已授权令牌

HTTP Post Form Fields :

无

正常输出:

HTTP Header : 无

HTTP Content :

resp_status : OK

resp_content:
无

2.5 信息聚合工具接口

2.5.1 获取用户拥有的画板列表

前端向 PPK 平台查询当前登录用户所关联拥有的第三方拼站画板列表信息，用于被选择作为转发目标。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/user/**UUID**/pinboard/

请求方式：API GET

输入：

HTTP URL Fields :

FRONT_ID : 前端环境唯一标识

UUID: 用户 ID

HTTP Header Cookie Fields :

access_token: 已授权令牌

HTTP Post Form Fields :

无

正常输出：

HTTP Header : 无

HTTP Content :

resp_status : OK

resp_content:

数组，取值示例如下：(蓝色字体为注释)

"18": //以网站记录标识为索引键值

{

"pinhost_title": "花瓣网", //所在网站名称

```

"pinboards": //用户在该网站的全部画板列表
{
  "1171223": //画板记录标识为键值
  {
    "pinhost_id": "18", //所在网站记录标识
    "pinhost_titl": "花瓣网", //所在网站名称
    "pingallery_id": "112671", //所在图册对应记录标识
    "pinboard_id": "1171223", //画板对应记录标识
    "pinboard_title": "创意", //画板标题
    "ppk_uuid": "125671", //用户标识
    "ph_board_id": "90915", //画板在所属网站的内部标识
    "ppk_user_ext_account_id": "1296", //对应关联帐号记录标识
    "ph_user_id": "g3381534" //画板拥有者在所属网站的内部标识

    "pb_pins": "21" //画板包含的 PIN 数
    "pb_goods": "15" //画板包含的属于商品的 PIN 数
    "pb_goods": "15" //画板包含的属于商品的 PIN 数
    "pb_follows": "231" //画板被关注数
    "pb_likes": "112" //画板被喜欢次数

  }
}

```

2.5.2 提交新的聚合记录数据

向 PPK 平台提交新的聚合记录数据，申请被系统自动预处理通过后获得标识编号，进入待处理状态。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/user/**UUID**/agent/

请求方式：API POST

HTTP URL Fields：

FRONT_ID：前端环境唯一标识

UUID：用户 ID

输入：

HTTP Header Cookie Fields：

access_token: 已授权令牌

HTTP Content Post Fields：

agent_title：聚合记录标，字符串

tags_str：相关的标签关键字字符串，多个关键字间以英文逗号分隔，可以为空

agent_remark：转发内容简介

original_url：来源网址，字符串

good_price_fen：商品参考价格，数字，单位：人民币分。可选。

agent_srcs_json：来源内容数组，JSON 格式编码，字符串。

取值举例：(蓝色字体为注释)

```
{
  "pics"=>[ //来源图片数组
    {
      "src_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524214F8.jpg", //图片 1
      "thumb_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524214F8_thumb.jpg", //对应预览图，可为空，表示没有生成预览图
    },
    {
      "src_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524220015.jpg", //图片 2，依次类推
      "thumb_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524220015_thumb.jpg", //对应预览图，可为空，表示没有生成预览图
    }
  ]
}
```

agent_dests_json：转发目标画板集合设置，JSON 格式编码，字符串。

取值举例：(蓝色字体为注释)

```
{
  "1165422":{ //画板对应记录标识作为数组元素的键值
    "pinhost_id":"18", //画板所在网站对应记录标识
    "pingallery_id":"116255", //画板所在图册对应记录标识
  }
}
```

```

    "pinboard_id": "1165422", //画板对应记录标识
    "pinboard_title": "\u6444\u5f71", //画板标题
    "ppk_uid": "123721", //所属用户标识
    "ppk_user_ext_accound_id": "2679", //对应关联帐号记录标识
    "ph_board_id": "850240", //画板在所属网站的内部标识
    "ph_user_id": "250450" //画板拥有者在所属网站的内部标识
  },
  "1168916": { //第 2 条设置，依次类推
    "pinhost_id": "72",
    "pingallery_id": "112561",
    "pinboard_id": "1168916",
    "pinboard_title": "\u7231\u8f66\u96c6\u5408",
    "ppk_uid": "123721",
    "ppk_user_ext_accound_id": "3567",
    "ph_board_id": "14027633982369",
    "ph_user_id": "10822198"
  }
}

```

正常输出：

HTTP Header ：无

HTTP Content：

resp_status : OK

resp_data:

agent_id ： 聚合记录唯一标识，字符串

agent_status ： 数据记录的状态，取值定义见
《PPK_API_Common_Define》

2.5.3 获取单条聚合记录的详细信息

向 PPK 平台提交获取指定编号的聚合记录对应的详细信息，包括转发处理结果。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/user/**UUID**/agent/**AGENT_ID**

请求方式：API GET

HTTP URL Fields：

FRONT_ID：前端环境唯一标识

UUID：用户 ID

AGENT_ID：聚合记录标识

输入：

HTTP Header Cookie Fields：

access_token: 已授权令牌

HTTP Content Post Fields：

无

正常输出：

HTTP Header：无

HTTP Content：

resp_status：OK

resp_data:

agent_id：聚合记录唯一标识，字符串

agent_status：数据记录的状态，取值定义见《PPK_API_Common_Define》

agent_user_zhid：所属用户 ID，字符串

agent_title: 标题，字符串

tags_str：相关的标签关键字字符串，多个关键字间以英文逗号分隔

original_url：内容原始来源网址，字符串

good_price_fen: 商品参考价格，整数，单位：人民币分

agent_remark: 备注说明信息，字符串

record_utc：数据录入时间戳，取值定义见《PPK_API_Common_Define》

agent_srcs：转发图片链接数组，数组。

取值举例：[\(蓝色字体为注释\)](#)

"pics"=>[//来源图片数组

{

"src_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524214F8.jpg", [//图片 1](#)

"thumb_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524214F8_thumb.jpg", [//对应预览图，可能为空，表](#)

示没有生成预览图

```
    },  
    {  
        "src_url"=>"http://cms.xitek.com/uploads/allimg/120524/  
84-120524220015.jpg", //图片 2，依次类推  
        "thumb_url"=>"http://cms.xitek.com/uploads/allimg/1205  
24/84-120524220015_thumb.jpg", //对应预览图，可能为空，表  
示没有生成预览图
```

```
    }  
]  
]  
agent_dests:
```

转发目标画板参数设置，数组。

取值举例：（蓝色字体为注释）

```
{  
    "1165422":{ //画板对应记录标识作为数组元素的键值  
        "pinhost_id":"18", //画板所在网站对应记录标识  
        "pinhost_title":"花瓣网", //画板所在网站名称  
        "pingallery_id":"116255", //画板所在图册对应记录标识  
        "pinboard_id":"1165422", //画板对应记录标识  
        "pinboard_title":"测试", //画板标题  
        "pinboard_url":"http://www.duitang.com/album/1165422/", //  
画板网址  
        "ppk_uuid":"123721", //所属用户标识  
        "ppk_user_ext_account_id":"2679", //对应关联帐号记录标识  
        "ph_board_id":"850240", //画板在所属网站的内部标识  
        "ph_user_id":"250450" //画板拥有者在所属网站的内部标识  
    },  
    "1168916":{ //第 2 条设置，依次类推  
        "pinhost_id":"72",  
        "pinhost_title":"开心集品",  
        "pingallery_id":"112561",  
        "pinboard_id":"1168916",  
        "pinboard_title":"测试",  
        "pinboard_url":"http://www.duitang.com/album/1168916/", //  
画板网址  
        "ppk_uuid":"123721",  
        "ppk_user_ext_account_id":"3567",  
        "ph_board_id":"14027633982369",  
        "ph_user_id":"10822198"  
    }  
}
```

relate_pins：转发处理生成的 PIN 记录，数组。

取值举例：（蓝色字体为注释）

```
{
  "17":{//所发网站标识
    "1165422":{//所发画板标识
      "11787123":{ //PIN 对应记录标识作为数组元素的键值
        "pinobj_id" : "11787123" //PIN 对应记录标识
        "belong_pinboard_id" : "1165422" //PIN 所属画板标识
        "belong_pingallery_id" : "1178511" //PIN 所属画册标识
        "belong_pinhost_id" : "17" //所在网站对应记录标识
        "belong_user_zhid" : "1000001" //所属用户标识
        "src_img_url":"http://img03.taobaocdn.com/bao/uploaded/i3/T1Z5W5XgRjXXbgeWIU_013603.jpg_310x310.jpg"
        //来源图片网址
        "pinobj_url":"http://www.duitang.com/people/mblog/31828072/detail/" //生成的 PIN 记录网址
        "pinobj_status" : "1" //PIN 记录状态
        "pin_clicks" : "2" //PIN 记录被点击数
        "pin_repins" : "3" //PIN 记录被转发数
        "pin_likes" : "-1" //PIN 记录被喜欢数,-1 表示未知
        "pin_comments" : "-1" //PIN 记录被评论数,-1 表示未知
      },
      "11787256":{ //同画板下的更多记录依次列出
        "pinobj_id" : "11787256"
        "belong_pinboard_id" : "1165422"
        "belong_pingallery_id" : "1178511"
        "belong_pinhost_id" : "17"
        "belong_user_zhid" : "1000001"
        "src_img_url":"http://img03.taobaocdn.com/bao/uploaded/i5/T1Z5W5XgRjXXbgeWIU_013603.jpg_310x310.jpg"
        "pinobj_url":"http://www.duitang.com/people/mblog/31828022/detail/"
        "pinobj_status" : "1"
        "pin_clicks" : "0"
        "pin_repins" : "0"
        "pin_likes" : "-1"
        "pin_comments" : "-1"
      }
    }
  }
  "1165422":{//发到该网站其它画板的更多记录依次列出
```

```

        //省略
    }
},
    "72" { //发到其它网站的更多记录依次列出
        //省略
    }
}

```

2.5.4 更新单条聚合记录数据

向 PPK 平台提交保存发生修改的聚合记录数据，系统将自动处理对应的修改。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/user/**UUID**/agent/**AGENT_ID**/

请求方式：API POST

HTTP URL Fields：

FRONT_ID：前端环境唯一标识

UUID：用户 ID

AGENT_ID：聚合记录标识

输入：

HTTP Header Cookie Fields：

access_token: 已授权令牌

HTTP Content Post Fields：

tags_str：相关的标签关键字字符串，多个关键字间以英文逗号分隔，可以为空

agent_remak：转发内容简介

original_url：来源网址，字符串

good_price_fen：商品参考价格，数字，单位：人民币分。可选。

agent_srcs_json：转发图片链接数组，JSON 格式编码，字符串。

取值举例：**(蓝色字体为注释)**

"pics"=>[//来源图片数组

{

"src_url"=>"http://cms.xitek.com/uploads/allimg/120524/"


```
84-120524214F8.jpg", //图片 1
    "thumb_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524214F8_thumb.jpg", //对应预览图，可为空，表示没有生成预览图
},
{
    "src_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524220015.jpg", //图片 2，依次类推
    "thumb_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524220015_thumb.jpg", //对应预览图，可为空，表示没有生成预览图
}
```

append_agent_srcs_json : 追加转发图片链接数组，JSON 格式编码，字符串。

取值举例：(蓝色字体为注释)

```
"pics"=>[ //来源图片数组
{
    "src_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524214F8.jpg", //图片 1
    "thumb_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524214F8_thumb.jpg", //对应预览图，可为空，表示没有生成预览图
},
{
    "src_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524220015.jpg", //图片 2，依次类推
    "thumb_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524220015_thumb.jpg", //对应预览图，可为空，表示没有生成预览图
}
]
```

注: 当 append_agent_srcs_json 和 agent_srcs_json 都没有有效内容时, 表示清空原有内容; 当 append_agent_srcs_json 有内容, agent_srcs_json 无内容时表示只是追加新内容到原有内容上。

agent_dests_json : 转发目标画板集合设置，JSON 格式编码，字符串。

取值举例：(蓝色字体为注释)

```
{
    "1165422":{ //画板对应记录标识作为数组元素的键值
        "pinhost_id":"18", //画板所在网站对应记录标识
        "pingallery_id":"116255", //画板所在图册对应记录标识
    }
}
```

```

    "pinboard_id":"1165422", //画板对应记录标识
    "pinboard_title":"\u6444\u5f71", //画板标题
    "ppk_uid":"123721", //所属用户标识
    "ppk_user_ext_accound_id":"2679", //对应关联帐号记录标识
    "ph_board_id":"850240", //画板在所属网站的内部标识
    "ph_user_id":"250450" //画板拥有者在所属网站的内部标识
  },
  "1168916":{ //第 2 条设置，依次类推
    "pinhost_id":"72",
    "pingallery_id":"112561",
    "pinboard_id":"1168916",
    "pinboard_title":"\u7231\u8f66\u96c6\u5408",
    "ppk_uid":"123721",
    "ppk_user_ext_accound_id":"3567",
    "ph_board_id":"14027633982369",
    "ph_user_id":"10822198"
  }
}

```

正常输出：

HTTP Header ：无

HTTP Content：

resp_status : OK

resp_data:

agent_id ： 聚合记录唯一标识，字符串

agent_status ： 数据记录的状态，取值定义见
《PPK_API_Common_Define》

2.5.5 添加图片等来源数据到指定聚合记录

向 PPK 平台的指定聚合记录数据追加图片等来源数据，系统将自动处理对应的修改。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/user/**UUID**/agent/**AGENT_ID**/source/

请求方式：API POST

HTTP URL Fields：

FRONT_ID：前端环境唯一标识

UUID：用户 ID

AGENT_ID：聚合记录标识

输入：

HTTP Header Cookie Fields：

access_token：已授权令牌

HTTP Content Post Fields：

append_srcs_json：来源数据数组，JSON 格式编码，字符串。

取值举例：（蓝色字体为注释）

```
"pics"=>[ //来源图片数组
{
  "src_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524214F8.jpg", //图片 1
  "thumb_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524214F8_thumb.jpg", //对应预览图，可为空，表示没有生成预览图
},
{
  "src_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524220015.jpg", //图片 2，依次类推
  "thumb_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524220015_thumb.jpg", //对应预览图，可为空，表示没有生成预览图
}
]
```

正常输出：

HTTP Header：无

HTTP Content：

resp_status：OK

resp_data:

agent_id：聚合记录唯一标识，字符串

agent_status : 数据记录的状态，取值定义见
《PPK_API_Common_Define》

2.5.6 删除单条聚合记录数据

向 PPK 平台提交删除指定聚合记录数据。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/user/**UUID**/agent/**AGENT_ID**

请求方式：API DELETE

HTTP URL Fields：

FRONT_ID : 前端环境唯一标识

UUID: 用户 ID

AGENT_ID : 聚合记录标识

输入：

HTTP Header Cookie Fields：

uuid : 已登录用户标识

access_token: 已授权令牌

正常输出：

HTTP Header : 无

HTTP Content：

resp_status : OK

resp_data:
无

2.5.7 提交查询聚合记录数据请求

向 PPK 平台提交查询指定聚合记录数据的请求。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/user/**UUID**/agent/search/

请求方式：API POST

HTTP URL Fields：

FRONT_ID：前端环境唯一标识

UUID：用户 ID

输入：

HTTP Header Cookie Fields：

uuid：已登录用户标识

access_token：已授权令牌

HTTP Content Post Fields：

agent_id：指定聚合记录编号

agent_status：指定聚合记录状态，取值定义见

《PPK_API_Common_Define》。可以为空，表示不限定

agent_url：指定转发短地址，可以为空，表示不限。

original_url：指定原始内容地址，可以为空，表示不限。

keywords：相关关键字，会自动于名称、标签等文字字段中进行匹配。
可以为空，表示不限定

record_utc_start：指定记录时间起始时间戳，UTC 时间，可以为空，表示不限。时间戳取值定义见

《PPK_API_Common_Define》

record_utc_end：指定记录时间结束时间戳，UTC 时间，可以为空，表示不限。时间戳取值定义见

《PPK_API_Common_Define》

prefered_result_num：优先返回的查询结果记录条数

order_by_col_name：指定结果集中排序使用的字段对应名称，可以从返回结果集的字段如 agent_id，

good_price_fen,agent_pins,agent_clicks 等中间选取一个。可以为空，表示不指定

order_by_direction：指定排序方向，可以为空，表示按数据缺省记录顺序。

取值范围：ASC 从小到大； DESC 从大到小

正常输出：

HTTP Header ：无

HTTP Content：

resp_status : OK

resp_data:

ppk_search_result_id : 查询结果对应临时记录集编号

ppk_search_result_num : 查询结果中的记录总条数

ppk_this_list_start_no : 此次返回列表对应起始记录序号

ppk_this_list_end_no : 此次返回列表对应结束记录序号

result_list: 查询结果数据列表数组，按 0，1 形式排序，每个元素的取值定义如下：

agent_id : 聚合记录唯一标识，字符串

agent_status : 数据记录的状态，取值定义见
《PPK_API_Common_Define》

agent_user_zhid: 所属用户 ID，字符串

agent_title: 标题，字符串

tags_str : 相关的标签关键字字符串，多个关键字间以英文逗号分隔

original_url: 内容原始来源网址，字符串

good_price_fen: 商品参考价格，整数，单位：人民币分

agent_remark: 备注说明信息，字符串

record_utc : 数据录入时间戳，取值定义见
《PPK_API_Common_Define》

agent_srcs : 转发图片链接数组，数组。

取值举例：(蓝色字体为注释)

"pics"=>[//来源图片数组

{

"src_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524214F8.jpg", //图片 1

"thumb_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524214F8_thumb.jpg", //对应预览图

},

{

"src_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524220015.jpg", //图片 2，依次类推

"thumb_url"=>"http://cms.xitek.com/uploads/allimg/120524/84-120524220015_thumb.jpg", //对应预览图

}

]

```

"agent_pins" : "0" //转发处理产生的 PIN 记录数
"agent_clicks" : "0" //转发处理产生的 PIN 记录合计被点击数
"agent_repins" : "0" //转发处理产生的 PIN 记录合计被转发数
"agent_likes" : "-1" //转发处理产生的 PIN 记录合计被喜欢数，-1 表示未知
"agent_comments" : "-1" //转发处理产生的 PIN 记录合计被评价数，-1 表示未知
]

```

2.5.8 获取指定查询聚合记录结果集数据

向 PPK 平台查询聚合记录结果集数据。

URL:

Http://PPK_API_HOST/api/v1/front/***FRONT_ID***/user/***UUID***/agent/search/***PPK_SEARCH_RESULT_ID/START_NO/END_NO***

请求方式：API GET

HTTP URL Fields :

FRONT_ID : 前端环境唯一标识

UUID: 用户 ID

PPK_SEARCH_RESULT_ID: 查询结果对应临时记录集编号

START_NO: 起始记录序号，可以为空，缺省表示从 1 开始

END_NO: 结束记录序号，可以为空，缺省表示取 ***START_NO*** 值，即只返回 1 条记录

录

输入：

HTTP Header Cookie Fields :

uuid : 已登录用户标识

access_token: 已授权令牌

正常输出：

HTTP Header : 无

HTTP Content :

resp_status : OK

resp_data:

ppk_search_result_id : 查询结果对应临时记录集编号
ppk_search_result_num : 查询结果中的记录总条数
ppk_this_list_start_no : 此次返回列表对应起始记录序号
ppk_this_list_end_no : 此次返回列表对应结束记录序号

result_list: 查询结果数据列表数组，按 0，1 形式排序，每个元素的取值定义如下：

agent_id : 聚合记录唯一标识，字符串

agent_status : 数据记录的状态，取值定义见
《PPK_API_Common_Define》

agent_user_zhid: 所属用户 ID，字符串

agent_title: 标题，字符串

tags_str : 相关的标签关键字字符串，多个关键字间以英文逗号分隔

original_url: 内容原始来源网址，字符串

good_price_fen: 商品参考价格，整数，单位：人民币分

agent_remark: 备注说明信息，字符串

record_utc : 数据录入时间戳，取值定义见
《PPK_API_Common_Define》

agent_srcs : 转发图片链接数组，数组。

取值举例：(蓝色字体为注释)

```
"pics"=>[ //来源图片数组
{
    "src_url"=>"http:\\cms.xitek.com\\uploads\\allimg\\120524\\84-120524214F8.jpg", //图片 1
    "thumb_url"=>"http:\\cms.xitek.com\\uploads\\allimg\\120524\\84-120524214F8_thumb.jpg", //对应预览图
},
{
    "src_url"=>"http:\\cms.xitek.com\\uploads\\allimg\\120524\\84-120524220015.jpg", //图片 2，依次类推
    "thumb_url"=>"http:\\cms.xitek.com\\uploads\\allimg\\120524\\84-120524220015_thumb.jpg", //对应预览图
}
]
```

"agent_pins" : "0" //转发处理产生的 PIN 记录数

"agent_clicks" : "0" //转发处理产生的 PIN 记录合计被点击数

"agent_repins" : "0" //转发处理产生的 PIN 记录合计被转发数

"agent_likes" : "-1" //转发处理产生的 PIN 记录合计被喜欢数，-1 表示未知

"agent_comments" : "-1" //转发处理产生的 PIN 记录合计被评价数，-1 表示未知

2.6 活动(Activity)组织接口

2.6.1 创建新的活动记录

向 PPK 平台提交新的活动记录数据，申请被系统自动预处理通过后获得标识编号。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/activity/

请求方式：API POST

HTTP URL Fields：

FRONT_ID ： 前端环境唯一标识

输入：

HTTP Header Cookie Fields：

uuid: 当前登录用户 ID

access_token: 已授权令牌

HTTP Content Post Fields：

activity_title ： 活动名称，字符串，可以“#标签 1#标签 2#”的形式指定关键字标签

activity_status ： 活动记录状态，取值定义见《PPK_API_Common_Define》

activity_privacy: 活动隐私设置，取值定义见《PPK_API_Common_Define》

activity_meetable: 活动可报名状态。取值定义见《PPK_API_Common_Define》

logo_url: 活动 LOGO 图片 URL，字符串

space_times_json : 活动时间和位置设置, JSON 格式的文本字符串, 取值定义见
《PPK_API_Common_Define》

meet_options_json : 活动报名参数设置, JSON 格式的文本字符串, 取值定义见
《PPK_API_Common_Define》

pay_sets_json : 活动支付参数设置, JSON 格式的文本字符串, 取值定义见
《PPK_API_Common_Define》

正常输出:

HTTP Header : 无

HTTP Content :

resp_status : OK

resp_data:

activity_id : 活动记录唯一标识, 字符串, 长度为 32 个 ASCII
字符

activity_status : 活动记录的状态, 取值定义见
《PPK_API_Common_Define》

2.6.2 获得指定活动记录详细信息

通过 PPK 平台获得指定活动记录详细信息。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/activity/**ACTIVITY_ID**

请求方式: API GET

HTTP URL Fields :

FRONT_ID : 前端环境唯一标识

ACTIVITY_ID: 活动记录唯一标识

输入:

HTTP Header Cookie Fields :

uuid: 当前登录用户 ID

access_token: 已授权令牌

正常输出 :

HTTP Header : 无

HTTP Content :

resp_status : OK

resp_data:

activity_id : 活动记录标识, 字符串

activity_title : 活动名称, 字符串

activity_status : 活动状态, 取值定义见《PPK_API_Common_Define》

activity_privacy: 活动可访问设置, 取值定义见

《PPK_API_Common_Define》

activity_meetable: 活动可报名状态, 可以为空, 表示不限。取值定义见

《PPK_API_Common_Define》

creator_uuid : 创建者的用户标识

creator_nickname : 创建者的用户昵称

waiting_num: 新报名待确认人数

meteed_num: 已确认的有效报名人数

checked_num: 线下实际签到人数

logo_url: 活动 LOGO 图片 URL, 字符串

record_utc : 数据录入时间戳, 取值定义见《PPK_API_Common_Define》

space_times_json : 活动时间和地点设置, JSON 格式的文本字符串, 取

值定义见《PPK_API_Common_Define》

meet_options_json : 活动报名参数设置, JSON 格式的文本字符串, 取值

定义见《PPK_API_Common_Define》

pay_sets_json : 活动支付参数设置, JSON 格式的文本字符串, 取值定义见《PPK_API_Common_Define》

2.6.3 修改指定活动记录

向 PPK 平台提交更新活动记录数据。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/activity/**ACTIVITY_ID**

请求方式: API POST

HTTP URL Fields :

FRONT_ID : 前端环境唯一标识
ACTIVITY_ID: 活动记录唯一标识

输入:

HTTP Header Cookie Fields :

uuid: 当前登录用户 ID
access_token: 已授权令牌

HTTP Content Post Fields :

activity_title : 活动名称, 字符串, 可以“#标签 1#标签 2#”的形式指定关键字标签
activity_status : 活动记录状态, 取值定义见《PPK_API_Common_Define》
activity_meetable: 活动可报名状态, 可以为空, 表示不限。取值定义见
《PPK_API_Common_Define》

logo_url : 活动 LOGO 图片 URL, 字符串

activity_privacy: 活动可访问设置, 取值定义见《PPK_API_Common_Define》

space_times_json : 活动时间和地点设置, JSON 格式的文本字符串, 取值定义见
《PPK_API_Common_Define》

meet_options_json : 活动参数设置, JSON 格式的文本字符串, 取值定义见
《PPK_API_Common_Define》

pay_sets_json : 活动支付参数设置, JSON 格式的文本字符串, 取值定义见
《PPK_API_Common_Define》

正常输出:

HTTP Header : 无

HTTP Content :

resp_status : OK

resp_data:

activity_id : 活动记录唯一标识, 字符串

activity_status : 活动记录的状态, 取值定义见
《PPK_API_Common_Define》

2.6.4 删除指定活动记录

向 PPK 平台提交删除指定活动记录数据。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/activity/**ACTIVITY_ID**

请求方式: API DELETE

HTTP URL Fields :

FRONT_ID : 前端环境唯一标识

ACTIVITY_ID: 活动记录唯一标识

输入：

HTTP Header Cookie Fields：

uuid: 当前登录用户 ID

access_token: 已授权令牌

正常输出：

HTTP Header：无

HTTP Content：

resp_status: OK

resp_data:

activity_id：活动记录唯一标识，字符串

activity_status：活动记录的状态，取值定义见

《PPK_API_Common_Define》

2.7 活动碰(Meet)接口

2.7.1 提交新的“碰”

向 PPK 平台提交新的报名记录。

URL: Http://PPK_API_HOST/api/v1/front/***FRONT_ID***/meet/***ACTIVITY_ID***/

请求方式：API POST

HTTP URL Fields：

FRONT_ID：前端环境唯一标识

ACTIVITY_ID: 活动记录唯一标识

输入：

HTTP Header Cookie Fields：

uuid: 用户 ID

access_token: 已授权令牌

HTTP Content Post Fields：

guster_id: 活动组织者代未注册用户报名时该报名者的标识，可以是手机号或者电子邮件，取值格式：手机“mobile:xxxxxxxxxxx” 电子邮件“email:xxxx@xxx.xx”

meeter_nickname: 报名提交者的用户昵称

meeter_info_json: 按活动要求填写的报名提交者的相关详细信息，JSON 格式的文本字符串，取值定义见《PPK_API_Common_Define》的“4.3 个人信息结构化数据”

meet_remark: 报名备注说明，字符串

ex_meet_datas_json：该活动自定义的报名输入项，JSON 格式的文本字符串，取值定义见《PPK_API_Common_Define》

正常输出：

HTTP Header：无

HTTP Content：

resp_status: OK

resp_data:

activity_id: 活动标识

meeter_id: “碰”者的标识，对已注册用户是其 UUID，对于未注册用户，可以是手机号或者电子邮件，取值格式：手机“mobile:xxxxxxxxxxx” 电子邮件“email:xxxx@xxx.xx”

meet_process：报名记录的处理状态，取值定义见

《PPK_API_Common_Define》

2.7.2 修改指定“碰”

向 PPK 平台提交更新已存在的报名记录。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/meet/**ACTIVITY_ID**/**MEETER_ID**

请求方式：API POST

HTTP URL Fields：

FRONT_ID：前端环境唯一标识

MEETER_ID：“碰”者标识，对已注册用户是其 UUID，对于未注册用户，可以是手机号或者电子邮件，取值格式：手机

“mobile:xxxxxxxxxxx” 电子邮件 “email:xxxx@xxx.xx”

ACTIVITY_ID：活动记录唯一标识

输入：

HTTP Header Cookie Fields：

uuid: 当前登录用户 ID

access_token: 已授权令牌（对应报名提交者或活动组织者，两者修改权限不一样）

HTTP Content Post Fields：

meeter_nickname: 报名提交者的用户昵称

meeter_info_json: 按活动要求填写的报名提交者的相关详细信息，JSON 格式的文本字符串，取值定义见《PPK_API_Common_Define》的“4.3 个人信息结构化数据”

meet_remark: 报名备注说明，字符串

ex_meet_datas_json：该活动自定义的报名输入项，JSON 格式的文本字符串，取值定义见《PPK_API_Common_Define》

meet_process：作为活动组织者修改报名记录的处理状态，取值定义见

《PPK_API_Common_Define》

pay_process：作为活动组织者修改支付处理状态，取值定义见

《PPK_API_Common_Define》

refund_process：作为活动组织者修改退款处理状态，取值定义见

《PPK_API_Common_Define》

正常输出：

HTTP Header ：无

HTTP Content：

resp_status：OK

resp_data:

meet_id ： 报名记录唯一标识，字符串

meet_process ： 报名记录的处理状态，取值定义见
《PPK_API_Common_Define》

2.7.3 删除指定“碰”

向 PPK 平台请求删除指定报名记录。

URL: Http://PPK_API_HOST/api/v1/front/***FRONT_ID***/meet/***ACTIVITY_ID***/***MEETER_ID***

请求方式：API DELETE

HTTP URL Fields：

FRONT_ID ： 前端环境唯一标识

MEETER_ID: “碰”者的标识，对已注册用户是其 UUID，对于未注册用户，
可以是手机号或者电子邮件，取值格式： 手机 “mobile:xxxxxxxxxxx” 电子邮件
“email:xxxx@xxx.xx”

ACTIVITY_ID: 活动记录唯一标识

输入：

HTTP Header Cookie Fields：

uuid: 当前登录用户 ID

access_token: 已授权令牌

正常输出：

HTTP Header ：无

HTTP Content :

resp_status : OK

2.7.4 获得指定“碰”详细信息

向 PPK 平台提交新的报名记录。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/meet/**ACTIVITY_ID**/**MEETER_ID**

请求方式：API GET

HTTP URL Fields :

FRONT_ID : 前端环境唯一标识

MEETER_ID: “碰”者的标识，对已注册用户是其 UUID，对于未注册用户，可以是手机号或者电子邮件，取值格式：手机
“mobile:xxxxxxxxxxx” 电子邮件 “email:xxxx@xxx.xx”

ACTIVITY_ID: 活动记录唯一标识

输入：

HTTP Header Cookie Fields :

uuid: 当前登录用户 ID（活动组织者和碰者自己有权查看）

access_token: 已授权令牌

正常输出：

HTTP Header : 无

HTTP Content :

resp_status : OK

resp_data:

activity_id : 报名记录标识，字符串

meeter_id : 碰者的标识，对已注册用户是其 UUID，对于未注册用户，可以是手机号或者电子邮件，取值格式：手机
“mobile:xxxxxxxxxxx” 电子邮件 “email:xxxx@xxx.xx”

meeter_nickname : 报名提交者的用户昵称

meeter_info_json: 按活动要求填写的报名提交者的相关详细信息, JSON 格式的文本字符串, 取值定义见

《PPK_API_Common_Define》的“4.3 个人信息结构化数据”

meet_remark: 报名备注说明

ex_meet_datas_json: 自定义输入项, JSON 格式的文本字符串, 取值定义见《PPK_API_Common_Define》

meet_utc : 报名生成时间戳, 取值定义见《PPK_API_Common_Define》

edit_utc : 报名最后一次修改时间戳, 取值定义见《PPK_API_Common_Define》

meet_process : 报名记录处理状态, 取值定义见《PPK_API_Common_Define》

pay_process : 支付处理状态, 取值定义见《PPK_API_Common_Define》

refund_process : 退款处理状态, 取值定义见《PPK_API_Common_Define》

2.7.5 meet 获得指定活动的相关"碰"列表

获取指定活动的相关的报名记录。

URL:

Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/meets/**ACTIVITY_ID**/**MEET_PROCESS**[**START_NO**/**END_NO**]

请求方式：API GET

HTTP URL Fields：

FRONT_ID：前端环境唯一标识

ACTIVITY_ID：活动记录唯一标识

MEET_PROCESS：报名处理状态，取值为 ALL 时表示不限，其他取值定义见
《PPK_API_Common_Define》

START_NO：起始记录序号，可以为空，缺省表示从 1 开始

END_NO：结束记录序号，可以为空，缺省表示取到默认最大可返回记录条数

输入：

HTTP Header Cookie Fields：

uuid：当前登录用户 ID（活动组织者有权查看）

access_token：已授权令牌

正常输出：

HTTP Header：无

HTTP Content：

resp_status：OK

resp_data:

ppk_search_result_id：查询结果对应临时记录集编号

ppk_search_result_num：查询结果中的记录总条数

ppk_this_list_start_no：此次返回列表对应起始记录序号

ppk_this_list_end_no：此次返回列表对应结束记录序号

result_list：报名数据列表数组，按 0，1...N 形式排序，每个元素的取值定义

如下：

meeter_id：“碰”者的标识，对已注册用户是其 UUID，对于未注册用户，可以是手机号或者电子邮件，取值格式：手机

“mobile:xxxxxxxxxxx” 电子邮件“email:xxxx@xxx.xx”

meeter_nickname：报名提交者的用户昵称

meet_utc：报名生成时间戳，取值定义见《PPK_API_Common_Define》

edit_utc：报名最后一次修改时间戳，取值定义见

《PPK_API_Common_Define》

meet_process : 报名记录处理状态, 取值定义见

《PPK_API_Common_Define》

2.8 活动关联(Connect)内容接口

2.8.1 关联新的内容

向 PPK 平台提交将新的内容关联到指定活动。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/conn/**ACTIVITY_ID**

请求方式: API POST

HTTP URL Fields :

FRONT_ID : 前端环境唯一标识

ACTIVITY_ID: 活动记录唯一标识

输入:

HTTP Header Cookie Fields :

uuid: 当前登录用户 ID

access_token: 已授权令牌

HTTP Content Post Fields :

pinobj_num: 内容对象数量

pinobj_list : 内容对象数据数组, 采用 JSON 格式, 字符串

数组元素下标按 0、1、2 依次编号, 每个元素的数据块定义如下:

pinobj_type : 对象类型, 取值定义见《PPK_API_Common_Define》

pinobj_id : PIN 对象标识, 字符串

conn_title: 关联标题, 可以“#标签 1#标签 2#”的形式加上关键字标签。

from_conn_id: 来源关联关系标识, 可以为空。该字段取值有效时表示转

发

正常输出:

HTTP Header : 无

HTTP Content :

resp_status : OK

resp_data:

result_num: 生成关联记录数

result_list: 关联记录列表数组, 按 0, 1...N 形式排序, 每个元素的取值定

义如下:

conn_id: 关联记录唯一标识, 字符串, 长度为 32 个 ASCII 字符

pinobj_type: 对象类型, 取值定义见《PPK_API_Common_Define》

pinobj_id : PIN 对象标识, 字符串

2.8.2 删除指定关联

向 PPK 平台请求删除指定关联记录。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/conn/**ACTIVITY_ID/CONN_ID/**

请求方式: API DELETE

HTTP URL Fields :

FRONT_ID : 前端环境唯一标识

ACTIVITY_ID: 活动记录唯一标识

CONN_ID: 关联记录 ID

输入:

HTTP Header Cookie Fields :

uuid: 当前登录用户 ID

access_token: 已授权令牌 (活动组织者和提交关联者都有权删除)

正常输出：

HTTP Header : 无

HTTP Content :

resp_status : OK

2.9 赞(Like)接口

2.9.1 “赞”指定活动的关联内容

向 PPK 平台提交赞指定关联内容。

URL: Http://PPK_API_HOST/api/v1/front/***FRONT_ID***/like/conn/***ACTIVITY_ID***/

请求方式：API POST

HTTP URL Fields :

FRONT_ID : 前端环境唯一标识

ACTIVITY_ID: 活动记录唯一标识

输入：

HTTP Header Cookie Fields :

uuid: 当前登录用户 ID

access_token: 已授权令牌

HTTP Content Post Fields :

conn_list : 关联关系记录数组，采用 JSON 格式，字符串

数组元素下标按 0、1、2 依次编号，每个元素的数据块定义如下：

conn_id : 关联关系标识, 字符串

正常输出:

HTTP Header : 无

HTTP Content :

resp_status : OK

2.9.2 删除对指定关联内容的“赞”

向 PPK 平台请求删除对指定关联记录的“赞”。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/like/conn/**ACTIVITY_ID**/**CONN_ID**/

请求方式: API DELETE

HTTP URL Fields :

FRONT_ID : 前端环境唯一标识

ACTIVITY_ID: 活动记录唯一标识

CONN_ID: 关联记录 ID

输入:

HTTP Header Cookie Fields :

uuid: 当前登录用户 ID

access_token: 已授权令牌

正常输出:

HTTP Header : 无

HTTP Content :

resp_status : OK

2.9.3 “赞”指定 PIN 类对象

向 PPK 平台提交赞指定 PIN 类对象。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/like/pinobj/

请求方式：API POST

HTTP URL Fields：

FRONT_ID：前端环境唯一标识

输入：

HTTP Header Cookie Fields：

uuid: 当前登录用户 ID

access_token: 已授权令牌

HTTP Content Post Fields：

pinobj_list：关联关系记录数组，采用 JSON 格式，字符串

数组元素下标按 0、1、2 依次编号，每个元素的数据块定义如下：

pinobj_type：对象类型，取值定义见《PPK_API_Common_Define》

pinobj_id：PIN 对象标识，字符串

正常输出：

HTTP Header：无

HTTP Content：

resp_status：OK

2.9.4 删除对指定 PIN 类对象的“赞”

向 PPK 平台请求删除对指定 PIN 类对象的“赞”。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/like/pinobj/**PINOBJ_ID**/

请求方式：API DELETE

HTTP URL Fields：

FRONT_ID：前端环境唯一标识
PINOBJ_ID：PIN 类对象唯一标识

输入：

HTTP Header Cookie Fields：

uuid: 当前登录用户 ID

access_token: 已授权令牌

正常输出：

HTTP Header：无

HTTP Content：

resp_status：OK

2.10 活动动态聚合(Activity Feed)接口

2.10.1 获取当前用户所参与指定或全部活动的最新动态聚合

通过 PPK 平台获取当前登录用户的指定或全部活动的最新动态聚合信息信息流。

URL:

Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/feed/activity/**LAST_UPDATE_UTC**[**ACTIVITY_ID**]

请求方式：API GET

HTTP URL Fields：

FRONT_ID : 前端环境唯一标识

LAST_UPDATE_UTC: 指定最后一次更新时间点, 系统将返回该时间点后的动态记录。第一次或者需全部获取时, 该值取为 0

ACTIVITY_ID: 可以指定活动记录标识, 不指定时表示该用户相关的全部活动

输入:

HTTP Header Cookie Fields :

uuid: 当前登录用户 ID

access_token: 已授权令牌

正常输出:

HTTP Header : 无

HTTP Content :

resp_status : OK

resp_data:

ppk_search_result_id : 查询结果对应临时记录集编号

ppk_search_result_num : 查询结果中的记录总条数

ppk_this_list_start_no : 此次返回列表对应起始记录序号

ppk_this_list_end_no : 此次返回列表对应结束记录序号

result_list: feed 列表数组, 按 0, 1...N 形式排序, 每个元素的取值定义如下:

activity_id: 关联活动记录唯一标识

feed_json : 动态内容, JSON 结构字符串, 取值定义见

《PPK_API_Common_Define》

feed_utc : 动态发生时间, 取值定义见《PPK_API_Common_Define》

2.11 查询(Search)接口

2.11.1 提交查询 PIN 类内容数据请求

向 PPK 平台提交查询指定 PIN 类对象数据的请求，结果将按随机方式排序。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/search/

请求方式：API POST

HTTP URL Fields：

FRONT_ID：前端环境唯一标识

输入：

HTTP Header Cookie Fields：

uuid: 当前登录用户 ID

access_token: 已授权令牌

HTTP Content Post Fields：

search_class：取值 pinobj

pinobj_type：对象类型，可以为空，表示不限定，取值定义见

《PPK_API_Common_Define》

keywords：相关关键字，会自动于名称、标签等文字字段中进行匹配。
可以为空，表示不限定

record_utc_start：指定记录时间起始时间戳，UTC 时间，可以为空，表示不限。时间戳取值定义见

《PPK_API_Common_Define》

record_utc_end：指定记录时间结束时间戳，UTC 时间，可以为空，表示不限。时间戳取值定义见

《PPK_API_Common_Define》

prefered_result_num：优先返回的查询结果记录条数

正常输出：

HTTP Header：无

HTTP Content：

resp_status：OK

resp_data:

ppk_search_result_id : 查询结果对应临时记录集编号
ppk_search_result_num : 查询结果中的记录总条数
ppk_this_list_start_no : 此次返回列表对应起始记录序号
ppk_this_list_end_no : 此次返回列表对应结束记录序号

result_list: 对象数据列表数组，按 0，1 形式排序，每个元素的取值定义如下：

pinobj_type : 系统认可的对象类型，取值定义见《PPK_API_Common_Define》
pinobj_id : PIN 类对象标识，字符串
pinobj_url : PIN 类对象所在网址，字符串
pinobj_title : PIN 类对象名称，字符串
content_type: 内容格式，取值定义见《PPK_API_Common_Define》
snap_url_default : 缺省建议使用的快照图像 URL
record_utc : 数据录入时间戳，取值定义见《PPK_API_Common_Define》
is_liker: 当前登录用户是否“赞”过该关联关系。大于 0 的数字表示“是”；取值为空时表示“否”

异步输出：

当不能马上应答查询结果时，则将采用异步格式，具体定义见

《PPK_API_Common_Define》。前端收到异步应答后，需要提取出其中返回的 ppk_search_result_id 字段取值，然后定时（建议 10 秒查询 1 次）调用”获取指定查询结果集数据”接口查询直至获得最终结果。

2.11.2 提交查询活动记录数据请求

前端向 PPK 平台提交查询符合条件的活动信息列表的请求。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/search/

请求方式：API POST

HTTP URL Fields：

FRONT_ID : 前端环境唯一标识

输入：

HTTP Header Cookie Fields :

uuid: 当前登录用户 ID

access_token: 已授权令牌

HTTP Content Post Fields :

search_class : 取值 activity

creator_uuid : 作为活动创建者的用户标识, 可以为空, 表示不限。

meeter_id : “碰”者的标识, 对已注册用户是其 UUID, 对于未注册用户, 可以是手机号或者电子邮件, 取值格式: 手机
“mobile:xxxxxxxxxxx” 电子邮件
“email:xxxx@xxx.xx”

activity_status : 活动记录状态, 可以为空, 表示不限。取值定义见

《PPK_API_Common_Define》

activity_privacy: 指定活动隐私设置, 可以为空, 表示不限。取值定义见《PPK_API_Common_Define》

activity_meetable: 活动可报名状态, 可以为空, 表示不限。取值定义见《PPK_API_Common_Define》

keywords : 相关关键字, 会自动于活动名称、标签等文字字段中进行匹配。可以为空, 表示不限定。多个关键字间以英文空格字符分隔。如需指定标签, 请使用“#标签名#”格式

record_utc_start : 指定记录时间起始时间戳, UTC 时间, 可以为空, 表示不限。时间戳取值定义见《PPK_API_Common_Define》

record_utc_end : 指定记录时间结束时间戳, UTC 时间, 可以为空, 表示不限。时间戳取值定义见《PPK_API_Common_Define》

prefered_result_num : 优先返回的查询结果记录条数

order_by_col_name : 指定结果集中排序使用的字段对应名称, 可以从返回结果集的字段如 activity_id, record_utc 等中间选取一个。可以为空, 表示不指定

order_by_direction : 指定排序方向, 可以为空, 表示按数据缺省记录顺序。取值范围: ASC 从小到大; DESC 从大到小

正常输出：

HTTP Header ：无

HTTP Content：

resp_status : OK

resp_data:

ppk_search_result_id ： 查询结果对应临时记录集编号

ppk_search_result_num ： 查询结果中的记录总条数

ppk_this_list_start_no ： 此次返回列表对应起始记录序号

ppk_this_list_end_no ： 此次返回列表对应结束记录序号

result_list： 活动数据列表数组，按 0，1...N 形式排序，每个元素的取值定义如下：

activity_id ： 活动记录标识，字符串

activity_title ： 活动名称，字符串

activity_status ：活动记录状态，取值定义见《PPK_API_Common_Define》

activity_privacy: 活动可访问设置，取值定义见

《PPK_API_Common_Define》

activity_meetable: 活动可报名状态，可以为空，表示不限。取值定义见

《PPK_API_Common_Define》

creator_uuid ： 创建者的用户标识

creator_nickname ： 创建者的用户昵称

logo_url： 活动 LOGO 图片 URL，字符串

record_utc ： 数据录入时间戳，取值定义见《PPK_API_Common_Define》

space_times_json ： 活动时间和地点设置，JSON 格式的文本字符串，取值定义见《PPK_API_Common_Define》

is_liker： 当前登录用户是否“赞”过该活动。大于 0 的数字表示“是”；

取值为空时表示“否”

异步输出：

当不能马上应答查询结果时，则将采用异步格式，具体定义见

《PPK_API_Common_Define》。前端收到异步应答后，需要提取出其中返回的 ppk_search_result_id 字段取值，然后定时（建议 10 秒查询 1 次）调用”获取指定查询结果集数据”接口查询直至获得最终结果。

2.11.3 提交查询关联关系的请求

向 PPK 平台提交查询关联内容的请求，结果将按随机方式排序。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/search/

请求方式：API POST

HTTP URL Fields：

FRONT_ID：前端环境唯一标识

输入：

HTTP Header Cookie Fields：

uuid: 当前登录用户 ID

access_token: 已授权令牌

HTTP Content Post Fields：

search_class: 取值 conn

activity_id: 指定活动标识，可以为空，表示不限定

maker_uuid: 指定内容关联者用户标识，可以为空，表示不限定

keywords: 相关关键字，会自动于名称、标签等文字字段中进行匹配。
可以为空，表示不限定

record_utc_start: 指定记录时间起始时间戳，UTC 时间，可以为空，表示不限。时间戳取值定义见

《PPK_API_Common_Define》

record_utc_end : 指定记录时间结束时间戳, UTC 时间, 可以为空, 表示不限。时间戳取值定义见

《PPK_API_Common_Define》

prefered_result_num : 优先返回的查询结果记录条数

order_by_col_name : 指定结果集中排序使用的字段对应名称, 可以从返回结果集的字段如 conn_id, record_utc 等中间选取一个。可以为空, 表示不指定

order_by_direction : 指定排序方向, 可以为空, 表示按数据缺省记录顺序。
取值范围: ASC 从小到大; DESC 从大到小

正常输出:

HTTP Header : 无

HTTP Content :

resp_status : OK

resp_data:

ppk_search_result_id : 查询结果对应临时记录集编号

ppk_search_result_num : 查询结果中的记录总条数

ppk_this_list_start_no : 此次返回列表对应起始记录序号

ppk_this_list_end_no : 此次返回列表对应结束记录序号

result_list : 对象数据列表数组, 按 0, 1 形式排序, 每个元素的取值定义如下:

conn_id : 关联关系标识

from_conn_id : 作为转发来源的关联关系标识, 该字段为空时表示原发。

conn_title : 关联描述标题

maker_uuid : 提交关联者用户标识

maker_nickname : 提交关联者用户昵称

activity_id : 关联的活动标识

is_liker : 当前登录用户是否“赞”过该关联关系。大于 0 的数字表示“是”; 取值为空时表示“否”

pinobj_type : 关联的内容对象类型, 取值定义见

《PPK_API_Common_Define》

pinobj_id : 关联的 PIN 类对象标识, 字符串

pinobj_url : PIN 类对象所在网址, 字符串
pinobj_title : PIN 类对象名称, 字符串
content_type : 内容格式, 取值定义见《PPK_API_Common_Define》
snap_url_default : 缺省建议使用的快照图像 URL
record_utc : 关联时间戳, 取值定义见《PPK_API_Common_Define》

异步输出：

当不能马上应答查询结果时，则将采用异步格式，具体定义见

《PPK_API_Common_Define》。前端收到异步应答后，需要提取出其中返回的 ppk_search_result_id 字段取值，然后定时（建议 10 秒查询 1 次）调用”获取指定查询结果集数据”接口查询直至获得最终结果。

2.11.4 获取指定查询结果集数据

向 PPK 平台查询指定对象数据。

URL:

Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/search/**PPK_SEARCH_RESULT_ID**/**START_NO**/**END_NO**

请求方式：API GET

HTTP URL Fields：

FRONT_ID : 前端环境唯一标识

PPK_SEARCH_RESULT_ID : 查询结果对应临时记录集编号

START_NO : 起始记录序号，可以为空，缺省表示从 1 开始

END_NO : 结束记录序号，可以为空，缺省表示取 **START_NO** 值，即只返回 1 条记录

输入：

HTTP Header Cookie Fields：

uuid: 当前登录用户 ID

access_token: 已授权令牌，取值为空时表示匿名访问

正常输出：

HTTP Header : 无

HTTP Content :

resp_status : OK

resp_data:

ppk_search_result_id : 查询结果对应临时记录集编号

ppk_search_result_num : 查询结果中的记录总条数

ppk_this_list_start_no : 此次返回列表对应起始记录序号

ppk_this_list_end_no : 此次返回列表对应结束记录序号

result_list: 结果数据列表数组, 多个元素按 0, 1, ...N 排序

结果项的内容定义参考发起查询请求时对应的应答定义

2.11.5 刷新指定查询结果集数据（尚未实现）

向 PPK 平台提交重新查询指定结果集。

URL: Http://PPK_API_HOST/api/v1/front/**FRONT_ID**/search/**PPK_SEARCH_RESULT_ID**

请求方式: API POST

HTTP URL Fields :

FRONT_ID : 前端环境唯一标识

PPK_SEARCH_RESULT_ID : 查询结果对应临时记录集编号

输入:

HTTP Header Cookie Fields :

uuid: 当前登录用户 ID

access_token: 已授权令牌, 取值为空时表示匿名访问

正常输出:

HTTP Header : 无

HTTP Content :

resp_status : OK

resp_data:

ppk_search_result_id : 查询结果对应临时记录集编号
ppk_search_result_num : 刷新后的查询结果中的记录总条数

3 接口调用开发示例

3.1 JQuery AJAX

(1) 建议定义一个公用的全局变量，用于设置平台 API 访问地址前缀。示例如下：
var PPK_API_URL="http://ppcore.sinaapp.com/api/v1/";

(2) 通过使用 AJAX 的 JSONP 回调模式来规避 JAVASCRIPT 的跨域限制问题。以通过 JSONP 来调用“获取前端唯一编号”接口为例：

```
//按平台 AJAX 支持接口定义组织 API 访问 URL
var cross_api_url=PPK_API_URL+"cross/?req=" +
'{"method":"POST","resource":"tool/gen_front_id/"}'+'&callback=?';

//调用 JQuery AJAX 方法
$.ajax({
    url: cross_api_url,
    type: "get",
    dataType: "jsonp",
    jsonp: "callback",
    success: function(data) {
        processFrondIdRespData(data); //注册处理应答结果的回调方法
    }
});

function processFrondIdRespData(resp_result)
{
    var g_strFrontId=resp_result['resp_data']['front_id']; //从结果数组中提取所需字段
    alert('New FrontId:'+g_strFrontId); //测试打印结果
}
```

3.2 PHP

待加