HTTP API Design Guide

This guide describes the design of Puluo APIs: common practices among all APIs and each API's input/output format.
The primary purpose of this set of APIs is to support front-end mobile applications.

Overview

This section describes how Puluo API works in general and practices we use among all APIs.

Dummy Result

All APIs have a dummy counterpart which returns hard-coded json response for testing and development. For example, if the API's URL is '/users/login', it's dummy API's path is '/dummy/users/login'. In addition, responses returned by dummy interfaces are not consistent nor meaningful. It's primarily to illustrate data format. For example, almost for all types uuids, dummy APIs use the same string.

Authentication

We use session based authentication for all APIs.
Each request must be through an authenticated, non-expired session,
and the server is always able to match a session to a user in the system.
A session is authenticated through successful login or registration.

Caching

ETag is not supported in this version.

Clients

Clients **MUST** address requests using HTTPS.

cURL Examples

cURL Examples are not available in this version.

Errors

Failing responses will have an appropriate status and a JSON body containing more details about a particular error.

See error responses for more example ids.

Name	Type Description	Example
id	int	unique numeric id
error_type	string	id of error raised
message	string	end user message of error raised
url	string	reference url with more information about the error

Note that the url is included only when relevant and may not be present in the response.

Error Response

```
{
  "id":       "rate_limit",
  "message":  "Your account reached the API rate limit.",
  "url":      "https://183.131.76.93/articles/rate_limit"
}
```

Methods

Method	Usage
GET	used for retrieving lists and individual objects
POST	used for updating existing objects
PUT	used for creating new objects

- We don't support 'DELETE' method. Data can only be invalidated (through POST).
- We don't support 'PATCH' method. Incremental updates are done by POST.

Parameters

Values that can be provided for an action are divided between optional and required values. Parameters should be JSON encoded and passed in the request body.

Ranges

We don't support Ranges in this version.

List requests will return a Content-Range header indicating the range of values returned.
Large lists may require additional requests to retrieve.
If a list response has been truncated you will receive a 206 Partial Content status and the
Next-Range header set.
To retrieve the next range, repeat the request with the Range header set
to the value of the previous request's Next-Range header.

Request-ID

Each API response contains a unique request id in the Request-Id header to facilitate
tracking.
When reporting issues, providing this value makes it easier to pinpoint problems
and provide solutions more quickly.

Status

The result of responses can be determined by returned status.

Successful Responses

Status	Description
200 OK	request succeeded
201 Created	resource created, for example a new app was created or an add-on was provisioned
202 Accepted	request accepted, but the processing has not been completed
206 Partial Content	request succeeded, but this is only a partial response, see ranges
###### Error Responses	
Error responses can be divided in to two classes.	
Client errors result from malformed requests and should be addressed by the	

client.	
Puluo errors result from problems on the server side and must be addressed internally.	

Client Error Responses

Status	Error ID	Description
400 Bad Request	bad_request	request invalid, validate usage and try again
401 Unauthorized	unauthorized	request not authenticated, session token is missing, invalid or expired
403 Forbidden	forbidden	request not authorized, provided credentials do not provide access to specified resource
403 Forbidden	suspended	request not authorized, account or application was suspended.
404 Not Found	not_found	request failed, the specified resource does not exist
406 Not Acceptable	not_acceptable	request failed, set Accept: application/vnd.heroku+json; version=3 header and try again
416 Requested Range Not Satisfiable	requested_range_not_satisfiable	request failed, validate Content-Range header and try again
422 Unprocessable Entity	invalid_params	request failed, validate parameters try again

Puluo Error Responses

Status	Description
500 Internal Server Error	error occurred, we are notified, but contact support if the issue persists
503 Service Unavailable	API is unavailable, check response body or Puluo status for details

Authentication

Register

Create a new user and returns error if the user already exists

```
PUT /users/register
```

Required Parameters

Name	Type	Description	Example
mobile	Long	user's mobile number	12346789000
password	String	SHA-256 of user's plain text password	cd8460a5e0f2c2af596f170009bffc02df06b54d
auth_code	String	authentication code user received from SMS	

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X PUT https://183
-H "Content-Type: application/json" \
-d '{
    "mobile": "12346789000",
    "password": "cd8460a5e0f2c2af596f170009bffc02df06b54d"
}'
```

Response Example

```
HTTP/1.1 201 Created
Last-Modified: Sun, 01 Jan 2012 12:00:00 GMT
```

```
{
    "uuid": "de305d54-75b4-431b-adb2-eb6b9e546013",
    "mobile": "12346789000",
    "password": "cd8460a5e0f2c2af596f170009bffc02df06b54d",
    "created_at": 12345677889900, // unix timestamp
    "updated_at": 12345677889900 // unix timestamp
}
```

Login

Login a user to the system by creating a session him.
Successful login (or the user has already logged in) returns the user's uuid and session's created_at

```
POST /users/login
```

Required Parameters

Name	Type	Description	Example
mobile	Long	user's login id	12346789000
password	String	SHA-256 of user's plain text password	cd8460a5e0f2c2af596f170009bffc02df06b54d

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X POST https://18
-H "Content-Type: application/json" \
-d '{
  "mobile": "12346789000",
  "password": "cd8460a5e0f2c2af596f170009bffc02df06b54d"
}'
```

Response Example

```
HTTP/1.1 201 Created
Last-Modified: Sun, 01 Jan 2012 12:00:00 GMT
```

```
{
  "uuid": "de305d54-75b4-431b-adb2-eb6b9e546013", //user's uuid
  "created_at": 12345677889900, // unix timestamp, session created time
  "last_login": 12345677889900 // unix timestamp, last login time, empty for
}
```

Logout

Destroy current session and all states saved in the session.

```
 POST /users/logout
```

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X POST https://18
-H "Content-Type: application/json"
```

Response Example

```
HTTP/1.1 200 Created
Last-Modified: Sun, 01 Jan 2012 12:00:00 GMT
```

```
{
  "uuid": "de305d54-75b4-431b-adb2-eb6b9e546013", //user's uuid
  "duration_seconds": "12345" //session created time
}
```

Update Password

Update a user's password

```
POST /users/credential/update
```

Required Parameters

Name	Type	Description	Example
password	String	SHA-256 of user's current password	cd8460a5e0f2c2af596f170009bffc02df06b54d
new_password	String	SHA-256 of user's new password	cd8460a5e0f2c2af596f170009bffc02df06b54d

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X POST https://18
-H "Content-Type: application/json" \
-d '{
  "password": "cd8460a5e0f2c2af596f170009bffc02df06b54d",
  "new_password": "cd8460a5e0f2c2af596f170009bffc02df06b54d"
}'
```

Response Example

```
HTTP/1.1 200 Created
Last-Modified: Sun, 01 Jan 2012 12:00:00 GMT
```

```
{
  "uuid": "de305d54-75b4-431b-adb2-eb6b9e546013",
  "created_at": 12345677889900, // unix timestamp
  "updated_at": 12345677889900 // unix timestamp
}
```

User

User Login Status

Get a user's login status

```
 GET /users/status
```

cURL Example

```
 $ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X GET https://183
```

Response Example

```
 HTTP/1.1 200 Created
 Last-Modified: Sun, 01 Jan 2012 12:00:00 GMT
```

```
 {"login":true}
```

User Profile

Get a user's complete profile

```
 GET /users/{user_mobile_or_uuid}
```

cURL Example

```
 $ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X GET https://183
```

Response Example

```
 HTTP/1.1 200 Created
 Last-Modified: Sun, 01 Jan 2012 12:00:00 GMT
```

```
 {
   "uuid": "de305d54-75b4-431b-adb2-eb6b9e546013",
   "public_info":{
     "first_name": "Tracey",
     "last_name":  "Boyd",
     "thumbnail": "http://upyun/puluo/userThumb.jpg!200",
     "large_image": "http://upyun/puluo/userThumb.jpg",
     "saying": "I've got an app for that."
     "likes": 2,
     "banned": false, // has the viewing user banned this users.
     "following": 1, // Is the user following this user.
     "is_coach":false
   },
   "private_info": { // this info is only visible for the logged in user
     "email": "tracey.boyd@kotebo.com",
     "sex": "m",
     "birthday": "1984-09-12",
     "occupation": "Internet Plumber",
     "country": "USA",
     "state": "Washington",
     "city": "Seattle",
     "zip": "234234"
   },
   "created_at": 12345677889900, // unix timestamp
   "updated_at": 12345677889900 // unix timestamp
 }
```

User Profile Update

Update user profile

```
 POST /users/update
```

Optional Parameters

Name	Type	Description	Example
first_name	String	user's first name	"Tracy"
last_name	String	user's last name	"Boyd"
thumbnail	String	link to thumbnail of user's profile image	"http://upyun/puluo/userThumb.jpg!200"

large_image	String	link to user's profile image	"http://upyun/puluo/userThumb.jpg!200"
saying	String	some words user put on his profile	"i've got an app for that"
email	String	user's email	"tracey.boyd@kotebo.com (mailto:tracey.boyd@kotebo.com)"
sex	String	user's sex	"M"
birthday	Date	user's birthday	1984-09-12
country	String	user's country	"USA"
state	String	user's state	"Washington"
city	String	user's city	"Seattle"
zip	String	user's zip code	"234234"

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X POST /users/upd
-H "Content-Type: application/json" \
-d '{
  "first_name": "baolin",
  "country": "China",
  "state": "Shanghai",
  "city": "Shanghai"
}'
```

Response Example

```
HTTP/1.1 200 Created
Last-Modified: Sun, 01 Jan 2012 12:00:00 GMT
```

```
{
  "uuid": "de305d54-75b4-431b-adb2-eb6b9e546013",
  "public_info":{
    "first_name": "Tracey",
    "last_name":  "Boyd",
    "thumbnail": "http://upyun/puluo/userThumb.jpg!200",
    "large_image": "http://upyun/puluo/userThumb.jpg",
    "saying": "I've got an app for that."
  },
  "private_info": { // this info is only visible for the logged in user
    "email": "tracey.boyd@kotebo.com",
    "sex": "m",
    "birthday": "1984-09-12",
    "occupation": "Internet Plumber",
    "country": "USA",
    "state": "Washington",
    "city": "Seattle",
    "zip": "234234"
  },
  "created_at": 12345677889900, // unix timestamp
  "updated_at": 12345677889900 // unix timestamp
}
```

User Search

Search user by parameters. If more parameters are specified, search conditions are 'AND' together.

```
POST /users/search
```

Optional Parameters

Name	Type	Description	Example
first_name	String	user's first name	"Tracy"
last_name	String	user's last name	"Boyd"
email	String	user's email	"tracey.boyd@kotebo.com (mailto:tracey.boyd@kotebo.com)"

mobile	String	user's mobile	"1234567780"

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X POST /users/upd
-H "Content-Type: application/json" \
-d '{
  "first_name": "baolin"
}'
```

Response Example

```
HTTP/1.1 200 Created
Last-Modified: Sun, 01 Jan 2012 12:00:00 GMT
```

```
{
    "details":[
    {
      "uuid": "de305d54-75b4-431b-adb2-eb6b9e546013",
      "public_info":{
        "first_name": baolins,
        "last_name":  Boyd,
        "email": "tracey.boyd@kotebo.com",
        "mobile": "123456789000"
      }
    },
    {
      "uuid": "ze2345d54-75b4-3234-adb2-ajfs230948jsdf",
      "public_info":{
        "first_name": baolins,
        "last_name":  Shao,
        "email": "blshao@qq.com",
        "mobile": "18646655333"
      }
    }
  ]
}
```

Setting

User Setting

Get user's setting

```
GET /users/privacy/{user_mobile_or_uuid}
```

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X GET https://183
```

Response Example

```
HTTP/1.1 200 Created
Last-Modified: Sun, 01 Jan 2012 12:00:00 GMT
```

```
{
    "user_uuid":"",
    "auto_add_friend":true,
    "allow_stranger_view_timeline":true,
    "allow_searched":true
}
```

User Setting Update

Update user setting

```
POST /users/setting/update
```

Optional Parameters

Name	Type	Description	Example
auto_add_friend	boolean	allow anyone to add the user as friend without approval	true
allow_stranger_view_timeline	boolean	allow everyone to view the user's timeline	true
allow_searchedl	boolean	allow the user to be searched	true

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X POST /users/set
-H "Content-Type: application/json" \
-d '{
  "auto_add_friend": "true"
}'
```

Response Example

```
HTTP/1.1 200 Created
Last-Modified: Sun, 01 Jan 2012 12:00:00 GMT
```

```
HTTP/1.1 200 Created
Last-Modified: Sun, 01 Jan 2012 12:00:00 GMT
```

```
{
    "user_uuid":"",
    "auto_add_friend":true,
    "allow_stranger_view_timeline":true,
    "allow_searched":true
}
```

Social Graph

List Friends

Get a user's own friends list

```
 GET /users/friends
```

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X GET /users/frie
```

Response Example

```
HTTP/1.1 200 Created
Last-Modified: Sun, 01 Jan 2012 12:00:00 GMT
```

```
{
    "details":[
    {
      "uuid": "de305d54-75b4-431b-adb2-eb6b9e546013",
      "public_info":{
        "first_name": baolins,
        "last_name":  Boyd,
        "email": "tracey.boyd@kotebo.com",
        "mobile": "123456789000"
      }
    },
    {
      "uuid": "ze2345d54-75b4-3234-adb2-ajfs230948jsdf",
      "public_info":{
        "first_name": baolins,
        "last_name":  Shao,
        "email": "blshao@qq.com",
        "mobile": "18646655333"
      }
    }
  ]
}
```

Request Friend

Send a private message of type 'FriendRequest' to another user

```
PUT /users/friends/request
```

Required Parameters

Name	Type	Description	Example
user_uuid	String	user's uuid	de305d54-75b4-431b-adb2-eb6b9e546013

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X PUT https://183
-H "Content-Type: application/json" \
-d '{
    "user_uuid": "de305d54-75b4-431b-adb2-eb6b9e546013",
}'
```

Response Example

```
HTTP/1.1 200 Created
Last-Modified: Sun, 01 Jan 2012 12:00:00 GMT
```

```
{
    friend_request:{
        request_id:"",
        status:"pending",
        messages:[
            {
                msg_id:"de305d54-75b4-431b-adb2-eb6b9e546013",
                from_user:"de305d54-75b4-431b-adb2-eb6b9e546013",
                to_user:"de305d54-75b4-431b-adb2-eb6b9e546013",
                from_user_thumbnail:"http://upyun.com/puluo/xxxx",
                to_user_thumbnail:"http://upyun.com/puluo/xxxx",
                content:"hi, this is Tracy!",
                created_at:12345677889900 // unix timestamp
            }
        ],
        created_at:12345677889900, // unix timestamp
        updated_at:12345677889900 // unix timestamp
    }
}
```

Delete Friend

De-friend two users. Delete all their past messages.

```
POST /users/friends/delete
```

Required Parameters

Name	Type	Description	Example
user_uuid	String	the other user's	de305d54-75b4-431b-adb2-

		uuid	eb6b9e546013

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X POST https://18
-H "Content-Type: application/json" \
-d '{
  "user_uuid": "de305d54-75b4-431b-adb2-eb6b9e546013"
}'
```

Response Example

```
HTTP/1.1 200 Created
Last-Modified: Sun, 01 Jan 2012 12:00:00 GMT
```

```
{
    past_messages:[
        {
            msg_uuid:"de305d54-75b4-431b-adb2-eb6b9e546013"
        },
        {
            msg_uuid:"de305d54-75b4-431b-adb2-eb6b9e546013"
        }
    ]
}
```

Deny Friend

Deny a friend request.

```
POST /users/friends/deny
```

Required Parameters

Name	Type	Description	Example
user_uuid	String	a user's uuid	de305d54-75b4-431b-adb2-eb6b9e546013

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X POST https://18
-H "Content-Type: application/json" \
-d '{
  "user_uuid": "de305d54-75b4-431b-adb2-eb6b9e546013"
}'
```

Response Example

```
{
    friend_request:{
        request_id:"",
        status:"denied",
        messages:[
            {
                msg_id:"de305d54-75b4-431b-adb2-eb6b9e546013",
                from_user:"de305d54-75b4-431b-adb2-eb6b9e546013",
                to_user:"de305d54-75b4-431b-adb2-eb6b9e546013",
                from_user_thumbnail:"http://upyun.com/puluo/xxxx",
                to_user_thumbnail:"http://upyun.com/puluo/xxxx",
                content:"hi, this is Tracy!",
                created_at:12345677889900 // unix timestamp
            },
            {
                msg_id:"de305d54-75b4-431b-adb2-eb6b9e546013",
                from_user:"de305d54-75b4-431b-adb2-eb6b9e546013",
                to_user:"de305d54-75b4-431b-adb2-eb6b9e546013",
                from_user_thumbnail:"http://upyun.com/puluo/xxxx",
                to_user_thumbnail:"http://upyun.com/puluo/xxxx",
                content:"hi, this is Tracy!",
                created_at:12345677889900 // unix timestamp
            }
        ],
        created_at:12345677889900, // unix timestamp
        updated_at:12345677889900 // unix timestamp
    }
}
```

Approve Friend

Approve a friend request.

```
POST /users/friends/approve
```

Required Parameters

Name	Type	Description	Example
user_uuid	String	a user's uuid	de305d54-75b4-431b-adb2-eb6b9e546013

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X POST https://18
-H "Content-Type: application/json" \
-d '{
  "user_uuid": "de305d54-75b4-431b-adb2-eb6b9e546013"
}'
```

Response Example

```
{
    friend_request:{
        request_id:"",
        status:"approved",
        messages:[
            {
                msg_id:"de305d54-75b4-431b-adb2-eb6b9e546013",
                from_user:"de305d54-75b4-431b-adb2-eb6b9e546013",
                to_user:"de305d54-75b4-431b-adb2-eb6b9e546013",
                from_user_thumbnail:"http://upyun.com/puluo/xxxx",
                to_user_thumbnail:"http://upyun.com/puluo/xxxx",
                content:"hi, this is Tracy!",
                created_at:12345677889900 // unix timestamp
            },
            {
                msg_id:"de305d54-75b4-431b-adb2-eb6b9e546013",
                from_user:"de305d54-75b4-431b-adb2-eb6b9e546013",
                to_user:"de305d54-75b4-431b-adb2-eb6b9e546013",
                from_user_thumbnail:"http://upyun.com/puluo/xxxx",
                to_user_thumbnail:"http://upyun.com/puluo/xxxx",
                content:"hi, this is Tracy!",
                created_at:12345677889900 // unix timestamp
            }
        ],
        created_at:12345677889900, // unix timestamp
        updated_at:12345677889900 // unix timestamp
    }
}
```

Message

Send Message

Current user send a message to his/her friend
Currently, we only support text messages.

```
PUT /users/message/send
```

Required Parameters

Name	Type	Description	Example
to_uuid	String	the other user's uuid	de305d54-75b4-431b-adb2-eb6b9e546013
content	String	content of message	"hello"
content_type	String	type of content, e.g. Text or Image	"Text"

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X PUT https://183
-H "Content-Type: application/json" \
-d '{
  "to_uuid": "de305d54-75b4-431b-adb2-eb6b9e546013",
  "content": "hello",
  "content_type":"text"
}'
```

Response Example

```
{
    msg_id:"de305d54-75b4-431b-adb2-eb6b9e546013",
    from_user:"de305d54-75b4-431b-adb2-eb6b9e546013",
    to_user:"de305d54-75b4-431b-adb2-eb6b9e546013",
    from_user_thumbnail:"http://upyun.com/puluo/xxxx",
    to_user_thumbnail:"http://upyun.com/puluo/xxxx",
    content:"hi, this is Tracy!",
    created_at:12345677889900 // unix timestamp
}
```

List Message

Get messages from a specific user since a specific time

```
POST /users/messages
```

Optional Parameters

Name	Type	Description	Example
user_uuid	String	the other user's uuid	"de305d54-75b4-431b-adb2-eb6b9e546013"
since	Time	all returned messages should be after this time, unix timestamp	12345677889900

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X GET https://183
-H "Content-Type: application/json" \
-d '{
  "user_uuid": "de305d54-75b4-431b-adb2-eb6b9e546013",
  "since": 12345677889900 // unix timestamp
}'
```

Response Example

```
{
    messages:[
        {
            msg_id:"de305d54-75b4-431b-adb2-eb6b9e546013",
            from_user:"de305d54-75b4-431b-adb2-eb6b9e546013",
            to_user:"de305d54-75b4-431b-adb2-eb6b9e546013",
            from_user_thumbnail:"http://upyun.com/puluo/xxxx",
            to_user_thumbnail:"http://upyun.com/puluo/xxxx",
            content:"hi, this is Tracy!",
            created_at:12345677889900 // unix timestamp
        },
        {
            msg_id:"de305d54-75b4-431b-adb2-eb6b9e546013",
            from_user:"de305d54-75b4-431b-adb2-eb6b9e546013",
            to_user:"de305d54-75b4-431b-adb2-eb6b9e546013",
            from_user_thumbnail:"http://upyun.com/puluo/xxxx",
            to_user_thumbnail:"http://upyun.com/puluo/xxxx",
            content:"hi, this is Tracy!",
            created_at:12345677889900 // unix timestamp
        }
    ]
}
```

Event

Event Registration

Create orders and returns links to alipay

```
GET /events/payment/{event_uuid}
```

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X GET https://183
```

Response Example

```
{
    "link": "https://alipay.com/xxxx",
    "order_uuid":"de305d54-75b4-431b-adb2-eb6b9e546013"
}
```

Error Response

id	error_type	message	url	description
1	系统支付错误	Event不存在(uuid is 123456)		event not found
2	系统支付错误	订单已取消		order is canceled
3	系统支付错误	订单中的用户id与该用户不匹配		order doesn't match the user

Event Detail

Get detail information of an event

```
GET /events/detail/{event_uuid}
```

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X GET /events/det
```

Response Example

```
{
    "status" : "open", //status includes 'open', 'closed', 'cancel', 'full'.
    "event_name": "Weapons of Ass Reduction",
    "event_time": 12345677889900, // unix timestamp
    "address": "888 Happy Mansion",
    "city": "Beijing",
    "phone": "86-555-5555",
    "coach_name": "Mr. Bob Smith",
    "coach_uuid": "de305d54-75b4-431b-adb2-eb6b9e546013",
    "thumbnail": [
        "http://upyun.com/puluo/head.jpg",
    ]
    "registered_users": 23,
    "capacity": 30,
    "likes":1,
    "geo_location":{
        "lattitude":"",
        "":""
    },
    "details": "Get fit with friends.",
    "images": [
        "http://upyun.com/puluo/image1.jpg",
        "http://upyun.com/puluo/image2.jpg"
    ]
}
```

Event Memory

Get list of user image links of an event

```
POST /events/memory
```

Optional Parameters

Name	Type	Description	Example
max_count	Int	max number of links returned	10

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X POST https://18
-H "Content-Type: application/json" \
-d '{
  "max_count": "5"
}'
```

Response Example

```
{
    "memories": [
        "http://upyun.com/puluo/image1.jpg",
        "http://upyun.com/puluo/image2.jpg"
    ]
}
```

Events Search

If keyword is specified, search events from name and description and then return 'max_count' sorted and filtered events from the 'n'-th item.

If keyword is not specified, return all events with sorting and filter.

```
POST /events/search
```

Optional Parameters

Name	Type	Description	Example
event_date	Date	date of event	2015-01-08
keyword	String	keyword to search	"Boxing"
sort	String	sorting parameter	"distance"
sort_direction	String	direction of sort	"Asc" or "Desc"
user_lattitude	String	lattitude of user	""
user_xxx	String	xxx of user	""
range_from	Int	starting index of all query events	10

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X POST /events/se
-H "Content-Type: application/json" \
-d '{
  "keyword": "yoga",
  "event_date": "2014-02-18",
  "sort":"distance",
  "sort_direction":"desc",
  "user_lattitude":"xxx",
  "user_xx":"xxx",
  "range_from":"10"
}'
```

Response Example

```
{
    events:[
        {
            "status" : "open", //status includes 'open', 'closed', 'cancel',
            "event_name": "Weapons of Ass Reduction",
            "event_time": 12345677889900, // unix timestamp
            "address": "888 Happy Mansion",
            "city": "Beijing",
            "phone": "86-555-5555",
            "coach_name": "Mr. Bob Smith",
            "coach_uuid": "de305d54-75b4-431b-adb2-eb6b9e546013",
            "thumbnail": [
                "http://upyun.com/puluo/head.jpg"
            ]
            "registered_users": 23,
            "capacity": 30,
            "likes":1,
            "geo_location":{
                "lattitude":"",
                "":""
            },
            "details": "Get fit with friends.",
            "images": [
                "http://upyun.com/puluo/image1.jpg",
                "http://upyun.com/puluo/image2.jpg"
            ]
        }
    ]
}
```

Timeline

User Timeline

Get a list of events user attended/to attend

```
POST /users/timeline
```

Optional Parameters

Name	Type	Description	Example
user_uuid	String	user's uuid	de305d54-75b4-431b-adb2-eb6b9e546013
since_time	Time	returned events should be after 'since_time', unix timestamp	12345677889900

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X https://183.131
-H "Content-Type: application/json" \
-d '{
  "user_uuid": "de305d54-75b4-431b-adb2-eb6b9e546013"
}'
```

Response Example

```
{
    timelines:[
        {
            timeline_uuid:"",
            event:{
                event_uuid:"de305d54-75b4-431b-adb2-eb6b9e546013",
                event_name:"Weapon of big ass reduction",
                created_at:12345677889900 // unix timestamp
            },
            my_words:"This is an awesome event",
            likes:[
                {
                    user_uuid:"de305d54-75b4-431b-adb2-eb6b9e546013",
                    user_name:"Bob",
                    created_at:12345677889900 // unix timestamp
                },
                {
                    user_uuid:"de305d54-75b4-431b-adb2-eb6b9e546013",
                    user_name:"Bob",
                    created_at:12345677889900 // unix timestamp
                }
            ],
            comments:[
                {
                    comment_uuid:"",
                    reply_to_uuid:"",
                    user_uuid:"de305d54-75b4-431b-adb2-eb6b9e546013",
                    user_name:"Bob",
                    content:"",
                    read:"false",
                    created_at:12345677889900 // unix timestamp
                }
            ],
            created_at:12345677889900, // unix timestamp
            updated_at:12345677889900 // unix timestamp
        }
    ]
}
```

Like Timeline

Like an event on a user's timeline

```
PUT /users/timeline/like
```

Required Parameters

Name	Type	Description	Example
timeline_uuid	String	uuid of timeline	de305d54-75b4-431b-adb2-eb6b9e546013

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X PUT https://183
-H "Content-Type: application/json" \
-d '{
  "timeline_uuid": "de305d54-75b4-431b-adb2-eb6b9e546013"
}'
```

Response Example

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

Remove Timeline Like

Remove a like to an event on a user's timeline

```
PUT /users/timeline/delike
```

Required Parameters

Name	Type	Description	Example
timeline_uuid	String	uuid of timeline	de305d54-75b4-431b-adb2-eb6b9e546013

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X PUT https://183
-H "Content-Type: application/json" \
-d '{
  "timeline_uuid": "de305d54-75b4-431b-adb2-eb6b9e546013"
}'
```

Response Example

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

Comment Timeline

Add a new comment or reply to a comment from user's timeline

```
PUT /users/timeline/comment
```

Required Parameters

Name	Type	Description	Example
timeline_uuid	String	uuid of timeline	de305d54-75b4-431b-adb2-eb6b9e546013
content	String	content of comment	"awsome"

Optional Parameters

Name	Type	Description	Example
reply_to	String	comment uuid	de305d54-75b4-431b-adb2-eb6b9e546013

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X https://183.131
-H "Content-Type: application/json" \
-d '{
  "timeline_uuid": "de305d54-75b4-431b-adb2-eb6b9e546013",
  "content":"awesome",
  "reply_to": "de305d54-75b4-431b-adb2-eb6b9e546013"
}'
```

Response Example

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

Delete Timeline Comment

Delete a comment from user's timeline

```
PUT /users/timeline/comment/delete
```

Required Parameters

Name	Type	Description	Example
comment_uuid	String	uuid of comment	de305d54-75b4-431b-adb2-eb6b9e546013

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X https://183.131
-H "Content-Type: application/json" \
-d '{
  "comment_uuid": "de305d54-75b4-431b-adb2-eb6b9e546013"
}'
```

Response Example

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

Service

Email

Send notification email to users

```
PUT /services/email
```

Required Parameters

Name	Type	Description	Example
email_type	String	type of email	"order notification"

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X PUT https://183
-H "Content-Type: application/json" \
-d '{
  "email_type": "order_notification"
}'
```

Response Example

```
{
    email:"baolins@ms.com",
    status:"success"
}
```

Image Upload

Upload images to server

```
 POST /service/images
```

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X POST https://18
```

Response Example

```
{
    images:[
        {
            image_link:"http://upyun.com/puluo/xxxx",
            status:"success"
        },
        {
            image_link:"http://upyun.com/puluo/xxxx",
            status:"success"
        }
    ]
}
```

Send Registration Verification Code

Send regeistraiton verification code through SMS to users. This API doesn't require any authenticaiton.

```
PUT /services/sms/register
```

Required Parameters

Name	Type	Description	Example
mobile	String	user's mobile	"12345677889"

cURL Example

```
$ curl -n -X PUT https://183.131.76.93/services/sms/register
-H "Content-Type: application/json" \
-d '{
  "mobile": "12345677889"
}'
```

Response Example

```
{
    mobile:"12345677889",
    status:"success"
}
```

SMS

Send notification SMS to users

```
PUT /services/sms
```

Required Parameters

Name	Type	Description	Example
sms_type	String	type of sms	"order notification"

cURL Example

```
$ curl --cookie "JSESSIONID=14hx6i00llj1oi3fawse5rz3q" -n -X PUT https://183
-H "Content-Type: application/json" \
-d '{
  "sms_type": "order notification"
}'
```

Response Example

```
{
    mobile:"12345677889",
    status:"success"
}
```