API Design Specification 接口设计规范

Version	Author	Time	Update	review	mark
V1.0	tim.zhao	2021-01-21	Created	zhiyong.liu	ok
V1.1	shaozhu.lin	2022-06-02	add version information	zhiyong.liu	ok

协议Agreement

1. 对内对外提供接口都要使用https协议

It's advised to use https protocol when providing API both internally and externally

接口命名规范 API Naming Convention

1. 使用小写加下划线的命名方式Use lowercase and low dash
2. 输出JSON数据格式的api Output API in JSON format
    a. 如需要登陆态验证，需要放到/api/空间下If it requires login authentication, you need to put it in the /api/ space
    b. 如使用token验证，需要放到/openapi/空间下If it requires token authentication, you need to put it in the /openapi/ space
    c. 如有专用域名比如api.xx.com或openapi.xx.com那么可以不加/api或/openapi空间　　If there is a dedicated domain such as api.xx.com or openapi.xx.com, then the /api or /openapi space can be omitted

鉴权Authentication

1. 所有接口都要鉴权All interfaces must be authenticated
2. 有登陆态需要鉴权登陆态If there is a login status, then that requires login authentication
3. 服务器对服务器调用使用token机制，token强制使用JWT The server adapts the token mechanism for server calling, and the token mandatorily use JWT

返回协议Return Protocol

1. 必须使用JSON Must use JSON
2. JSON的格式使用通用头部 JSON format uses common headers

```
{
  "retcode": 0,  // 错误码，0成功，非0整数表示失败，定义为错误码。Error code, 0 success, non-0 integer means failure, defined as error code.

      "message": '', // 成功或失败时的错误信息Error message on success or failure

      "data": {        // 这个节点必须是object This node must be an object

      },

  }
```

1. JSON的字段命名都要是小写下划线分隔的形式JSON field names must be in lowercase and underscore-separated form
2. retcode不要复用http的状态码Do not reuse the status code of http for retcode
3. 对参数检查错误，需要明确参数的错误原因For parameter check errors, it is necessary to clarify the reason for the error.
4. 出于兼容性考虑（主要是 Android），三个最外层字段（"retcode"、"message"、"data"）都要求必填。 For compatibility reasons (mainly Android), the three outermost fields ("retcode", "message", "data") are all required.

通用返回码General Return Code

1. 成功的业务返回码固定是 0；The successful business return code is fixed to 0
2. 所有失败的业务返回码是非0整数，由业务系统自行指定，但是必须在接口文档里列举所有错误码及其错误原因说明。所有失败的业务返回码是非0整数，由业务系统自行指定，但是必须在接口文档里列举所有错误码及其错误原因说明。All failed business return codes are non-zero integers, which are specified by the business system, but all error codes and their error reasons must be listed in the API document. All failed business return codes are non-zero integers, which are specified by the business system, but all error codes and their error reasons must be listed in the API document.

通用请求参数Common request parameters

对于列表中的参数名不要改变用途Do not change the usage of parameter names in lists

- pageno
- count
- sortby
- callback

通用响应参数General Response Parameter

任何时候，接口响应的 data 参数都必须是一个 object，不可以是数组或者其他任何基本数据类型。At any time, the data parameter of the interface response must be an object, not an array or any other type of basic data.

列表型接口返回结果规范 API Return Result Specification

参数名 Parameter	类型 Type	释义 Explanation	是否必填 Required to fill
count	integer	页面大小Page Size	否No
pageno	integer	当前页数Current No. of Pages	否No
total	integer	数据集总记录数The total number of records in the dataset	否No
list	array	匹配请求内容的数据列表A list of data matching the requested content	是Yes
extra	object	额外的信息，所有无法由 list 容纳的数据，都要放在 extra Additional information or the data that cannot be accommodated by list, shall be placed in extra	否No

注意： 所有超出以上所列字段的参数，均视为不符合规范的 API 设计。

Note: All parameters beyond the fields listed above are considered as not comply with API design.

示例Sample：

```
{
 "retcode": 0,
 "message": "success",
 "data": {
  "count": 10,
  "pageno": 1,
  "total": 391,
  "extra": {
   "order_id": 39992,
   "order_status": 1
  },
  "list": [
```

```
    {
      "ctime": 33991933,
      "name": "Test"
    },
    {
      "ctime": 39919393,
      "name": "Test2"
    }
  ]
 }
}
```

资源详情类接口返回结果规范 [The resource details] interface returns the result specification

针对详情类页面的接口设计，返回的数据，应将资源本身信息直接展开放置在 "data" 字段下。

For the API design of the details page, the returned data should directly expand the information of the resource itself and place it under the "data" field.

示例Sample：

```
{
 "retcode": 0,
 "message": "success",
 "data": {
   "return_inbound_id": "RI2018070338919289",
   "status": 0,
   "inbound_type": "return_inbound",
   "whs_id": "IDG",
   "order_sn": "180102004071RQR",
   "return_awb": "SF29019021111",
   "shop_id": "33090192",
   "outbound_whs_id": "IDG",
   "return_reason": "Something is broken!",
   "ctime": 1530609620,
   "buyer_name": "Matthew Louis",
   "buyer_contact": "381928229",
   "buyer_address": "大有路558號9樓"
 }
}
```

接口文档要求 API Documentation Specification

1. 需要总体介绍接口背景，功能和用途A general introduction to the interface background, function and purpose is required
2. 需要包含各个环境的域名Need to include the domain name of each environment
3. 定义返回的错误码列表和错误码说明Define the returned error code list and error code description
4. 定义返回的数据结构，以及数据结构的字段说明(非通用性的接口，可以放到每一个接口）Define the returned data structure, as well as the field description of the data structure (non-generic interface, can be placed in each interface)
5. 对每一个接口For each API
    a. 需要说明功能Need to explain the function
    b. 调用逻辑，和调用序列(如果有)Call logic, and call sequence (if any)
    c. 输入参数说明和定义，是否必要，检查逻辑Enter parameter description and definition, if necessary, check logic
    d. 异常和异常说明，尤其对于特殊的逻辑错误Exceptions and exception descriptions, especially for special logic errors
    e. 输出数据和结构定义Output data and structure definitions
    f. 需要一个调用样例和返回样例Need a call sample and return sample
6. 不能包含token约定的账号和密钥Should not contain the account and key agreed by the token

接口设计文档样例Sample API Design Document

接口设计文档管理平台API Design Document Management Platform：apidoc

示例接口设计Sample API Design ：示例项目 Sample Project