**API定义**

接口地址：<http://210.51.190.172/threek/>

调用示例：登陆：

<http://210.51.190.172/threek/user/login?account=xx&password=x>

接口中成功或失败中是列出了错误码，返回格式为：

｛error\_code:””,error\_desc:””｝

1. **登陆**

URL: /user/login

HTTP请求方式：POST

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 必选 | 类型 | 说明 |
| account | true | string | 帐户 |
| password | true | string | 密码（用md5加密后传给服务器） |
|  |  |  |  |

返回结果：

成功返回：{uid:””, access\_token:””}

失败返回：10001

1. **注册**

URL: [/user/reg](http://0.0.0.0/api/user/reg)

HTTP请求方式：POST

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 必选 | 类型 | 说明 |
| account | true | long | 帐户(4-20位) |
| password | true | string(6-18位) | 密码（用md5加密后传给服务器） |

返回结果：

成功返回：{uid:””, access\_token:””}

失败返回：10002, 10003, 10004

1. **退出登陆**

URL: /user/logout

HTTP请求方式：POST

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 必选 | 类型 | 说明 |
| uid | true | long | 用户ID |
| access\_token | true | string |  |
|  |  |  |  |

返回结果：

成功返回：88888

1. **上线/离线**

URL: /user/status

HTTP请求方式：POST

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 必选 | 类型 | 说明 |
| uid | true | long | 用户ID |
| access\_token | true | string |  |
| op | true | byte | 1:上线， 2：离线 |

返回结果：

成功返回：88888

1. **获取个人信息**

URL：/user/profile

HTTP请求方式：GET

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 必选 | 类型 | 说明 |
| uid | true | long | 用户ID |
|  |  |  |  |

返回结果：

错误返回：10005

成功返回：

{

avatar: http://….test.jpg, 头像

nickname:test, 昵称

intro: “…” ， 简介

sex: “” ，性别

goods\_count:宝贝数量

online: 是否在线

}

1. **编辑个人信息**

URL：[/user/editProfile](http://0.0.0.0/api/user/editProfile)

HTTP请求方式：POST

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 必选 | 类型 | 说明 |
| uid | true | long | 用户ID |
| access\_token | true | string |  |
| nickname | false | string | 昵称 |
| intro | false | string | 简介 |
| sex | false | char | 性别 |
| push\_token | false | string | IOS push  用token |

返回结果：

错误返回：10005

**成功返回：**88888

1. **获取周边商品信息及用户发布中的商品（filter为2）**

URL: /goods/getList

HTTP请求方式：GET

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 必选 | 类型 | 说明 |
| access\_token |  |  |  |
| lon | true | float | 经度 |
| lat | true | float | 纬度 |
| count | false | int | 获取条数,默认10 |
| get\_time | false | int | 获取开始时间(不传此参数, 会返回前count条) |
| op | false | int | 1:获取最新商品（大于get\_time），  -1:获取更多商品（小于get\_time） |
| keyword | false | string | filter为1时此处传关键字，filter为2时此处传用户ID |
| filter | false | byte | 结果筛选：0：全部， 1：关键字， 2：用户。。。。。3：获取关注人发布的商品 |

返回结果：

[{

id：商品ID

img:””,商品小图

title:””,商品标题

money:0,商品价钱，为0表示免费

lon：经度

lat：纬度

publish\_time：发布时间

dist\_time:距现在发布时间(秒==当前时间-发布时间)

comment\_count: 评论条数

},………..]

1. **下架商品**

URL: /goods/offline

HTTP请求方式：POST

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 必选 | 类型 | 说明 |
| access\_token | true | string |  |
| goods\_id | true | long | 商品ID |

1. **商品详细信息**

URL： [/goods/detail](http://0.0.0.0/api/goods/detail)

HTTP请求方式：GET

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 必选 | 类型 | 说明 |
| ~~access\_token~~ | ~~true~~ | ~~string~~ |  |
| goods\_id | true | long | 商品ID |
|  |  |  |  |

返回结果：

｛

lon：经度

lat：纬度

big\_img:商品大图

publish\_time：发布时间

dist\_time:距现在发布时间(秒==当前时间-发布时间)

desc:描述

p\_uid : 发布人ID

｝

1. **~~关注某人~~**

URL: [user/watch](http://0.0.0.0/api/user/watch)

HTTP请求方式：GET

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 必选 | 类型 | 说明 |
| access\_token | true | string |  |
| c\_uid | true | long | 被关注人ID |
| uid | true | long | 关注人ID |
|  |  |  |  |

返回结果：

1. **~~获取关注的人~~**

URL:user/getWatchList

HTTP请求方式：GET

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 必选 | 类型 | 说明 |
| access\_token | true | string |  |
| uid | true | long | 用户ID |

返回结果：

[{

avatar: http://….test.jpg, 头像

nickname:test, 昵称

intro: “…” ， 简介

sex: “” ，性别

goods\_count:宝贝数量

online: 是否在线

}……….]

1. **~~给某人评价~~**

URL：[/user/score](http://0.0.0.0/api/user/score)

1. **发布或编辑商品**

URL：[/goods/publish](http://0.0.0.0/api/goods/publish)

HTTP请求方式：POST

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 必选 | 类型 | 说明 |
| access\_token | true | string |  |
| uid | true | long | ID |
| goods\_id | false |  | 修改的goods id |
| lon | true | float | 经度 |
| lat | true | float | 纬度 |
| money | true | int | 0为免费 |
| title | true | string | 商品标题 |
| desc | true | string | 商品描述 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

成功返回：

{

“uid”:”” UID

“goods\_id”:”” 商品ID

}

1. **商品咨询信息列表**

URL: /comment/getList

HTTP请求方式：GET

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 必选 | 类型 | 说明 |
| access\_token | true | string |  |
| goods\_id | true | string | ID |
| count | false | int | 获取条数,默认10 |
| get\_time | false | int | 获取开始时间(不传此参数, 会返回前count条) |
| op | false | int | 1:获取最新咨询（大于get\_time），  -1:获取更多咨询（小于get\_time） |

返回结果：

[{

uid: “” 咨询人ID

c\_id: 咨询ID

avatar:”” 咨询人头像

nickname:”” 咨询人昵称

ask:”” 咨询问题

ask\_time:”” 咨询时间

answer:”” 回复

answer\_time 回复时间

｝。。。。。。]

1. **回复咨询信息**

URL:/comment/reply

HTTP请求方式：POST

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 必选 | 类型 | 说明 |
| access\_token | true | string |  |
| c\_id | true | long | 咨询ID |
| answer | true | string | 回答内容 |

返回结果：

1. **咨询**

URL:/comment/post

HTTP请求方式：POST

请求参数:

|  |  |  |  |
| --- | --- | --- | --- |
|  | 必选 | 类型 | 说明 |
| access\_token | true | string |  |
| goods\_id | true | long | 咨询商品ID |
| uid | true | long | 咨询人ID |
| question | true | string | 问题 |

返回结果：

1. **发私信**

URL：/message/send

HTTP请求方式：POST

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 必选 | 类型 | 说明 |
| access\_token | true | string |  |
| from\_uid | true | long | 发送人ID |
| to\_uid | true | long | 接收人ID |
| content | true | String | 内容 |
|  |  |  |  |

返回结果：

**18，获取私信列表**

URL：/message/getList

HTTP请求方式：GET

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 必选 | 类型 | 说明 |
| access\_token | true | string |  |
| uid | true | long | 用户id |
| count | false | int | 获取条数,默认10 |
| get\_time | false | int | 获取开始时间(不传此参数, 会返回前count条) |
| op | false | int | 1:获取最新私信（大于get\_time），  -1:获取更多私信（小于get\_time） |
|  |  |  |  |

返回结果：[{

other\_uid:对方id

other\_nickname:对方昵称

other \_avatar: 对方头像

content:最后一条内容（最后一条可能是对方发的，也可能是自己 发的）

time:最后一条内容时间

count：与此用户聊天新消息数

}]

**19，获取与某人的私信列表**

URL：/message/getDetailList

HTTP请求方式：GET

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
|  | 必选 | 类型 | 说明 |
| access\_token | true | string |  |
| uid | true | long | 用户id |
| other\_uid | true | long | 私信对方uid |
| count | false | int | 获取条数,默认10 |
| get\_time | false | int | 获取开始时间(不传此参数, 会返回前count条) |
| op | false | int | 1:获取最新聊天（大于get\_time），  -1:获取更多聊天（小于get\_time） |

返回结果：[{

other\_uid: 2， 私信对方id

other\_avtar: 私信对方头像

other\_nickname: 私信对方昵称

send\_uid: 2， 发送方向（比如值与other\_uid相同为对方发给自己，值与自己uid相同即为自己发给对方）

content:内容

time:时间

totalcount:总条数

}]

1. **删除私信**

**URL: /message/del**

**HTTP请求方式： POST**

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
|  | 必选 | 类型 | 说明 |
| access\_token | true | string |  |
| uid | true | long | 当前用户uid |
| other\_uid | true | long | 私信对方uid |

**返回结果：**

1. **图片上传**

**URL：**[**/img/upload**](http://0.0.0.0/api/img/upload)

**HTTP请求方式：POST**

**请求参数：**

|  |  |  |  |
| --- | --- | --- | --- |
|  | 必选 | 类型 | 说明 |
| access\_token | true | string |  |
| uid | true | long | 用户id |
| img | true | file |  |
| ext | string |  | 如果type为2，此处为goods\_id |
| type | int |  | 1:上传个人头像  2：上传商品图片 |

返回结果：

｛

url: 返回上传后的图片地址

｝

1. **错误处理：**

｛

error\_code:”” //错误码

error\_desc:”” //错误描述

｝