<http://mask.cloudsdee.com/>

一、用户接口：

1、用户注册

<URL:api/account/register_process/>

> HTTP请求方式

- POST > 请求参数:

- user\_name (string)

- password (string)

- email (string)

> 返回结果: - uid用户编号

2、用户登录

> URL:api/account/login\_process/

> HTTP请求⽅方式

- POST

> 请求参数:

- user\_name (string) [可以是用户名也可以是邮箱]

- password (string)

> 返回结果:

- uid 用户编号

- user\_name 用户名

- avatar\_file 头像

3、用户头像

> URL:api/account/avatar\_upload/

> HTTP请求方式

- POST

> 请求参数:

- user\_avatar (图片⽂文件)

- uid⽤用户编号

> 返回结果:

- preview头像地址

4、获取用户信息

> URL:api/account/get\_userinfo/

> HTTP请求方式

- GET

> 请求参数:

> [In] Int(UID)

 > [Out] String(用户头像URI)

> [Out] String(用户名)

> [Out] String(签名)

> [Out] Int(我的话题数)

> [Out] Int(我关注的人数)

> [Out] Int(关注我的人数)

> [Out] Int(赞同我的次数)

> [Out] Int(感谢我的次数)

> [Out] Int(答案被收藏次数)

> [Out] Int(当前登录⽤用户是否关注了该⽤用户,如果已关注,则has\_focus为1,否则为0)

5、获取用户问答信息

> URL:api/user.php

> HTTP请求方式

- GET

 > 请求参数:

> [In] Int(UID)

> [Out] String(用户头像URI)

> [Out] String(用户名)

> [Out] Int(我的话题数)

 > [Out] Int(我关注的人数)

> [Out] Int(关注我的人数)

> [Out] Int(赞同我的次数)

> [Out] Int(感谢我的次数)

> [Out] Int(答案被收藏次数)

6、用户信息

> URL:/api/profile.php

> HTTP请求方式

- GET

 > 请求参数:

> [In] Int(UID)

> [Out] String(用户名)

> [Out] Int(性别ID)

> [Out] String(个⼈简介)

> [Out] Int(行业ID)

> [Out] Date(生⽇)

7、修改用户信息

 > <URL:/api/profile_setting.php>

> HTTP请求⽅方式

- POST

> 请求参数:

 > [In] Int(UID) uid (必须)

> [In] String(⽤用户名) user\_name (必须)

 > [In] Int(性别ID) sex (tinyint,1:男 2:女 3:保密)

 > [In] String(个人简介) signature (string,个人签名[简介])

> [In] Int(行业ID) job\_id(int)

 >[In]Date(生日) birthday(int,Unix时间戳)

8、用户关注,取消关注操作

> URL:follow/ajax/follow\_people/

> HTTP请求⽅方式

- GET

> Header

 - COOKIE

> 请求参数

- uid (⽤用户ID)

 - NOTE:不需要传其他参数,如果当前⽤用户已经关注该⽤用户,会取消关注,反之则关注

二、问答接口：

1、获取分类信息

> URL:/api/category/get\_category/

> HTTP请求方式

- GET

> 请求参数:

- type (string) 可选,默认question

- parent\_id (int) 可选,默认0

> 返回的信息:  分类数组

2、发现(最新,推荐,热⻔门[30天,7天,当天],等待回复)

> <URL:api/explore/>

> HTTP请求⽅方式

- GET

> 请求参数: category(int) 可选 分类id

- per\_page (int) 可选,默认10

- page (int) 可选,默认1

 - day (int) 可选,默认30

- is\_recommend (int) 可选,有1和0两种值,默认0 [如果你是要返回“推荐”栏目的数据,这个参数 值设为1,sort\_type可以不设]

- sort\_type (string) 可选,有new,hot,unresponsive三种值,默认new new:最新 hot:热⻔门 unresponsive:等待回复

> 返回的信息:

- answer\_users 发布者信息

- answer 此问题的最后⼀一个回答信息(包括回答者) 如果问题现在0回复,那就没有

- topics 问题关联的话题

- user\_info 提问者信息

-update\_time最后更新时间即最后一个回答发布时间 total\_rows是当前⻚页的信息总条数,total\_rows是0表示没有数据

3、问题详情

> <URL:/api/question/question/?id=1>

> HTTP请求⽅方式

- GET

> 请求参数:

> [In] Int(问题ID)

> [Out] String(问题标题)

 > [Out] String(问题正文)

> [Out] Int(问题关注数)

> [Out] 回答列表(

> > Int(ID)

> > Int(用户ID)

 > > String(用户姓名)

> > String(用户头像)

> > String(用户一句话签名)

> > Int(赞同的人数)

> > String(回答预览)

>)

> [Out] Tag列表(

> > Int(ID)

> > Int(名称)

>)

4、关注该问题的用户列表

 > URL: /api/question\_fans\_user.php?id=1

> HTTP请求方式

- GET

> 请求参数:

> [in] id 问题编号

> [out] 用户id

> [out] 用户名

> [out] 用户头像

> [out] 用户一句话签名

5、回答的详细

> <URL:api/question/answer_detail/id=1>

> HTTP请求⽅方式

- GET

> 请求参数:

> [In] id 回答编号

> [Out] Int(回答ID)

> [Out] Int(所属问题ID)

> [Out] Int(⽤用户ID)

> [Out] String(⽤用户名)

> [Out] String(⽤用户签名)

> [Out] String(⽤用户头像)

> [Out] Int(赞同数)

> [Out] String(正⽂文)

> [Out] Int(评论数)

> [Out] Date(⽇日期)

> [Out] Int(当前登录⽤用户是否赞或踩了该回答,如果已赞,则vote\_value为1,如果已踩,vote\_value 为-1,否则为0)

6、回答的评论列表

> URL:/api/answer\_comment.php?id=1

> HTTP请求方式

- GET

> 请求参数:

 > [In] id 回答编号

> [Out] Int(评论ID)

> [Out] Int(评论⼈人UID)

> [Out] Int(评论⼈人ID)

> [Out] String(评论人姓名)

> [Out] String(评论正文)

 > [Out] Date(日期)

> [Out] Array at\_user (被@用户信息,如果有则会返回这个信息)

7、上传附件

> URL:api/publish/attach\_upload/

> HTTP请求⽅方式

- GET

> 请求参数:

- id (string) [附件类型,可能的值:question,article,answer(默认)]

- attach\_access\_key (string) (任意32位字符串,为了避免与历史的重复,建议 md5( time().rand(1,99) ) 这样⽣生成, 发布⼀一个问题时有多个附件,上传每个附件时这个值必须保持⼀一致)

- POST

> 请求参数:

- qqfile (文件域名称)

> 返回结果:

- attach\_access\_key (其实可以无视,这是你给我的,现在还给你,待会点发布的时候,这个也要提交的)

- attach\_id (这个是供插入在正文里的,如:[attach]9[/attach] ) - thumb (如果是图片,会返回90\*90的缩略图供预览) > 可能返回的错误信息:

- 请选择要上传的文件

- 文件类型⽆无效

- 文件尺寸过大

 - 上传失败, 请与管理员联系

8、发布问题

> URL:api/publish/publish\_question/

> HTTP请求方式

- POST

> 请求参数:

- question\_content (问题标题)

- question\_detail (问题详情,即正文内容)

* - uid(提问者id)
* - category\_id(分类id)
* - attach\_access\_key (可选,如果传了附件的话,必须要有这个)
* - topics (string, 话题标题,多个话题标题请以英⽂文逗号隔开可选)
* - is\_recommend(必填,0为不推荐,1为推荐)
* > 返回结果:
* - question\_id (问题ID)
* > 可能返回的错误信息:
* - 请输⼊入问题标题
* - 问题标题字数不得少于 5 个字
* - 你所在的⽤用户组不允许发布站外链接
* - 请为问题添加话题

9、回答问题

* > URL:api/publish/save\_answer/
* > HTTP请求⽅方式
* - POST
* > 请求参数:
* - uid (用户id)
* - question\_id (问题ID)
* - answer\_content (回答的内容)
* -attach\_access\_key (附件id)
* > 返回结果:
* - answer\_id (回答ID)
* > 可能返回的错误信息:
* - 问题不存在 - 已经锁定的问题不能回复

- 请输⼊入回复内容

- 不能回复自⼰己发布的问题,你可以修改问题内容

 - 一个问题只能回复一次,你可以编辑回复过的回复

10、发布文章

> <URL:api/publish/publish_article/>

> HTTP请求方式

- POST

> 请求参数:

- title (⽂文章标题)

- pic 文章封面链接

- message (正文内容)

- uid(发布者id)

 - category\_id(分类id)

- attach\_access\_key (可选,如果传了附件的话,必须要有这个)

- topics (string, 话题标题,多个话题标题请以英⽂、文逗号隔开可选)

> 返回结果:

- qarticle\_id\_id (问题ID)

 > 可能返回的错误信息:

- 请输入文章标题

- 文章标题字数不得少于 5 个字

 - 你所在的用户组不允许发布站外链接

- 请为文章添加话题

11、编辑文章

> <URL:api/publish/modify_article/>

> HTTP请求方式

- POST

> 请求参数:

 - title (⽂文章标题) (必填)

- article\_id(⽂文章id)(必填)

- message (正⽂文内容)

- uid(发布者id)(必填)

- category\_id(分类id)必填

- do\_delete(是否删除,如果不删除就不传,删除传任意值,建议为1)

> 返回结果:

- errno (1为成功)

 > 可能返回的错误信息:

- 请输入文章标题

- 文章标题字数不得少于 5 个字

 - 你所在的⽤用户组不允许发布站外链接

* 请为文章添加话题

12、编辑问题

> <URL:api/publish/modify_question/>

> HTTP请求⽅方式

- POST

> 请求参数:

- question\_content (问题标题) question\_id(问题id)

- question\_detail (问题详情,即正⽂文内容)

* -  uid(提问者id)
* -  category\_id(分类id)
* -  attach\_access\_key (可选,如果传了附件的话,必须要有这个)
* -  topics (string, 话题标题,多个话题标题请以英⽂文逗号隔开可选)
* -  do\_delete(是否删除,如果不删除就不传,删除传任意值,建议为1)
* >返回结果:
* - errno (1为成功)
* > 可能返回的错误信息:
* - 请输入问题标题
* - 问题标题字数不得少于 5 个字
* - 你所在的用户组不允许发布站外链接
* - 请为问题添加话题

三、商城接口

1、商城分类

> URL: api/shop/get\_category/

> HTTP请求方式

- GET

> 请求参数

> > [out] 分类的信息(

> > [Out] Int(分类id)

> > [Out] String(分类名)

2、商品列表

> URL:api/shop/get\_goods/

> HTTP请求方式

- GET

> 请求参数

> [in] int(category\_id 分类id 非必填)

> [in] int(page 分页 非必填)

 > [in] int(limit 每页商品数 非必填)

> > [out] 商品的信息