目录

[服务器请求与回复格式 2](#_Toc21062)

[登录与注册 2](#_Toc10597)

[Register 2](#_Toc12659)

[Login 3](#_Toc7739)

[用户相关 3](#_Toc378)

[UserInformation 3](#_Toc29099)

[问题相关 4](#_Toc20283)

[PostQuest 4](#_Toc30796)

[QuestByQid 4](#_Toc4938)

[QuestList 5](#_Toc11633)

[QuestListByLid 6](#_Toc21641)

[QuestListByUid 7](#_Toc6602)

[答案相关 8](#_Toc2268)

[SetAnswer 8](#_Toc29653)

[SelectAmountAnswerByQid 9](#_Toc27724)

[SelectAmountAnswerByUid 10](#_Toc28915)

[SelectAmountAnswerOfAttention 10](#_Toc2630)

[评论相关 11](#_Toc13353)

[PostComment 11](#_Toc20959)

[CommentList 11](#_Toc28619)

[相互关注 12](#_Toc5454)

[SetAttention 12](#_Toc14758)

[SelectAttention 12](#_Toc5609)

[SelectFollower 12](#_Toc19819)

[标签相关 13](#_Toc2419)

[AddUserLabel 13](#_Toc5982)

[AddQuestLabel 13](#_Toc10882)

[SelectQuestLabel 13](#_Toc29084)

[SelectUserLabel 14](#_Toc13618)

[主界面 15](#_Toc4211)

[homepage 15](#_Toc31383)

# 服务器请求与回复格式

## 登录与注册

#### Register

注册

请求：

account

password

type （int）

name

photo （int）

回复：

"isOK" 请求服务器工作正常

"type" 如果isOK为1才会有此标签

"information" 调试信息，标识服务器出现的情况

DEMO：

("isOK", **true**);

("type", 0);

("information", "regist success");

("isOK", **true**);

("type", 1);

("information", "account is exist");

("isOK", **true**);

("type", 1);

("information", "name is exist");

#### Login

登陆

请求：

account

password

回复：

"isOK" 请求服务器工作正常

"type" 如果isOK为true才会有此标签

"information" 调试信息，标识服务器出现的情况

DEMO：

("type", 1)

("information", "login success")

("isOK", **true**)

("type", 0)

("isOK", **true**)

("information", "account or password is not correct")

("isOK", **false**)

("information", "service error")

## 用户相关

#### UserInformation

请求用户信息

请求：

id id为空且已登陆时，返回该登陆用户的信息

回复：

name

type

agree\_sum

photo

information 用户不存在时才有此项，且无以上4项

isOK

Demo:

略

## 问题相关

#### PostQuest

发表提问

请求：

title

detail

type

回复：

isOK

#### QuestByQid

请求：

qid

回复：

"isOK" 请求成功与否

"quests": 返回一个quests数组,该数组只含一个quest，包含下列属性值

[

"id"

"uid"

"title"

"detail"

“post\_time”

"answer\_sum"

“name”

]

DEMO

{

"isOK": true,

"quests": [

{

"id": "139",

"uid": "11",

"title": "title4",

"detail": "detial4",

"post\_time": "2017-06-29 14:51:46.0",

"answer\_sum": "0",

"name": "????"

},

{

"id": "140",

"uid": "12",

"title": "title4",

"detail": "detial4",

"post\_time": "2017-06-29 14:51:46.0",

"answer\_sum": "0",

"name": "unknow"

}

]

}

#### QuestList

动态页和公告页，使用type进行区分。

请求：

type quest类型，0或者1

page 页码

回复：

"isOK" 请求成功与否

"quests": 返回一个quests数组,包含下列属性值

[

"id"

"uid"

"title"

"detail"

“post\_time”

"answer\_sum"

“name”

]

DEMO

{

"isOK": true,

"quests": [

{

"id": "139",

"uid": "11",

"title": "title4",

"detail": "detial4",

"post\_time": "2017-06-29 14:51:46.0",

"answer\_sum": "0",

"name": "????"

},

{

"id": "140",

"uid": "12",

"title": "title4",

"detail": "detial4",

"post\_time": "2017-06-29 14:51:46.0",

"answer\_sum": "0",

"name": "unknow"

}

]

}

#### QuestListByLid

请求：

lid

page 页码

回复：

"isOK" 请求成功与否

"quests": 返回一个quests数组,包含下列属性值

[

"id"

"uid"

"title"

"detail"

“post\_time”

"answer\_sum"

“name”

]

DEMO

{

"isOK": true,

"quests": [

{

"id": "139",

"uid": "11",

"title": "title4",

"detail": "detial4",

"post\_time": "2017-06-29 14:51:46.0",

"answer\_sum": "0",

"name": "????"

},

{

"id": "140",

"uid": "12",

"title": "title4",

"detail": "detial4",

"post\_time": "2017-06-29 14:51:46.0",

"answer\_sum": "0",

"name": "unknow"

}

]

}

#### QuestListByUid

请求：

uid

page 页码

回复：

"isOK" 请求成功与否

"quests": 返回一个quests数组,包含下列属性值

[

"id"

"uid"

"title"

"detail"

“post\_time”

"answer\_sum"

“name”

]

DEMO

{

"isOK": true,

"quests": [

{

"id": "139",

"uid": "11",

"title": "title4",

"detail": "detial4",

"post\_time": "2017-06-29 14:51:46.0",

"answer\_sum": "0",

"name": "????"

},

{

"id": "140",

"uid": "12",

"title": "title4",

"detail": "detial4",

"post\_time": "2017-06-29 14:51:46.0",

"answer\_sum": "0",

"name": "unknow"

}

]

}

## 答案相关

#### SetAnswer

编辑回答

请求：

qid

uid

detail

回复：

isOK 成功与否

注：answer表数据库设计：插入时不允许 qid与uid同时重复

#### SelectAmountAnswerByQid

根据问题id获取指定数目的答案（按发布时间排序）

请求：

qid

page 页基数

回复：

"isOK" 请求成功与否

"answers": 返回一个answers数组,包含下列属性值

[

"detail"

"uid"

"post\_time"

"agree\_sum":

"comment\_sum"

]

DEMO

{

"isOK": true,

"answers": [

{

"detail": "asdda",

"uid": 100,

"post\_time": "2017-09-06 16:21:56.0",

"agree\_sum": 0,

"comment\_sum": 0

},

{

"detail": ".........................................................",

"uid": 103,

"post\_time": "2017-09-06 15:45:50.0",

"agree\_sum": 0,

"comment\_sum": 0

},

{

"detail": "测试",

"uid": 102,

"post\_time": "2017-07-05 16:02:15.0",

"agree\_sum": 0,

"comment\_sum": 0

}

]

}

#### SelectAmountAnswerByUid

根据用户id获取指定数目的答案（按发布时间排序）

请求：

uid 用户

page 页基数

回复：

格式同上

"isOK" 请求成功与否

"answers": 返回一个answers数组,包含下列属性值

[

"detail": ".............................................................",

"qid": 100,

"post\_time": "2017-09-06 15:45:50.0",

"agree\_sum": 0,

"comment\_sum": 0

]

#### SelectAmountAnswerOfAttention

需要时再实现

## 评论相关

#### PostComment

发表评论

请求

aid

detail

回复

"isOK" 请求成功与否

#### CommentList

请求

aid

page

回复

"isOK" 请求成功与否

"comments": 返回一个comments数组,包含下列属性值

[

"id"

"uid"

"detail"

“post\_time”

“name”

]

## 相互关注

#### SetAttention

当前用户关注 att\_id

请求

! 前提:需要用户先登录

att\_id

回复

isOK

#### SelectAttention

返回他的偶像集合

请求

id

回复

isOK

Attentions 他的偶像集合，包括下面属性

[

Attention 偶像id

name 偶像姓名

photo 头像编号

]

#### SelectFollower

返回他的粉丝集合

请求

att\_id

回复

isOK

Followers他的偶像集合，包括下面属性

[

Follower 偶像id

name 偶像姓名

photo 头像编号

]

## 标签相关

#### AddUserLabel

给用户加标签

请求

! 前提:需要用户先登录

lid 标签id

回复

isOK 返回false的话可能是没有登录

#### AddQuestLabel

给问题增加标签

请求

qid

uid

回复

isOK

#### SelectQuestLabel

查询问题的标签

请求:

qid

回复

isOK 查询成功与否

labels 返回标签数组，包含以下属性值:

[

"label" 标签值

]

DEMO

{

"isOK": true,

"labels": [

{

"label": 12

},

{

"label": 111

},

{

"label": 5

},

{

"label": 7

}

]

}

#### SelectUserLabel

查询用户感兴趣的标签

请求:

uid

回复

isOK 查询成功与否

labels 返回标签数组，包含以下属性值:

[

"label" 标签值

]

## 主界面

#### homepage

关注页按时间先后推送 关注人的quest和answer

请求

! 需要事先登录

page 页码

回复

"isOK" 类型为boolean型，表示是否请求成功

"answers": json数组，里面每个元素都是一个answer/question集合（用qtype属性区分）

包含下列属性值：

[ "id": answer/question的id

"uid": 提出该答案/问题相关的用户

"title": 小标题

"detail": 详细内容

"qtype": 0表示问题，-1表示答案

"post\_time": 发布时间

"lower\_level\_sum": qtype==0表示该提问“下层数目”即回答数，若qtype== -1表示该回答“下层数目”即评论的数目

"agree\_sum": -1 qtype==-1时，此处表示答案的赞数。

]

DEMO

{

"isOK": true,

"answers": [

{

"id": 32,

"uid": 103,

"title": "不乱码的标题",

"detail": "测试是否乱码",

"qtype": 0,

"post\_time": "2017-07-07 10:44:23.0",

"lower\_level\_sum": 0,

"agree\_sum": -1

},

{

"id": 11,

"uid": 101,

"title": "-1",

"detail": "测试",

"qtype": -1,

"post\_time": "2017-07-05 16:37:48.0",

"lower\_level\_sum": 0,

"agree\_sum": 0

}

]

}