软件总体功能规划与数据库设计

* **软件功能要点：**

1. 时间要单独的作为一个模块来返回时间（当天只返回：时:分:秒，当年返回：月-日 时:分:秒，其他的要返回：年-月-日 时:分:秒）
2. 获得回复需要有一个通知信息
3. 除了帖子下的回复楼主的留言，其他留言都有父系留言，其父系为无父系的留言的id

* **单独模块**

1. 根据帖子id获得该帖子的直系留言（直接回复帖子的留言）
2. 根据帖子id和father\_id获得帖子留言下面的回复信息
3. 根据留言id删除留言以及回复此留言的留言
4. 根据留言id获得留言的相关信息
5. 验证模块（用户身份、是否是自身）

* **数据库设计：**

1. 用户表（message\_user）

|  |  |  |
| --- | --- | --- |
| 字段 | 数据类型 | 描述 |
| id | mediumint | 主键（通过id查找用户） |
| username | varchar(50) | 用户昵称 |
| password | varchar(100) | 账号秘密 |
| account | varchar(100) | 用户账户 |
| image | varchar(100) | 用户头像保存路径 |
| sex | tinyint | 用户性别 |
| active | tinyint | 用户的状态（0 正常 1 封禁） |
| status | tinyint | 用户身份（1管理员，2普通用户） |

1. 帖子表（message\_theme）

|  |  |  |
| --- | --- | --- |
| 字段 | 数据类型 | 描述 |
| id | int | 主键 |
| user\_id | mediumint | 用户id（记录发布帖子用户） |
| title | text | 帖子的标题 |
| context | text | 帖子的内容 |
| response\_num | mediumint | 帖子回复数量 |
| time | datatime | x-x-x h:m:s |

1. 回复表（message\_response）

|  |  |  |
| --- | --- | --- |
| 字段 | 数据类型 | 描述 |
| id | int | 主键 |
| theme\_id | int | 回复的帖子id |
| father\_id | int | 父楼id |
| response\_id | int | 回复留言的留言id |
| user\_response | mediumint | 留言回复者id |
| user\_accept | mediumint | 留言接收者id |
| agree | mediumint | 留言点赞数（用于排序） |
| report | mediumint | 举报数 |
| context | text | 帖子的内容（可以是单纯的图片，也可以是纯文本） |
| time | datatime | x-x-x h:m:s |

1. 回复通知表

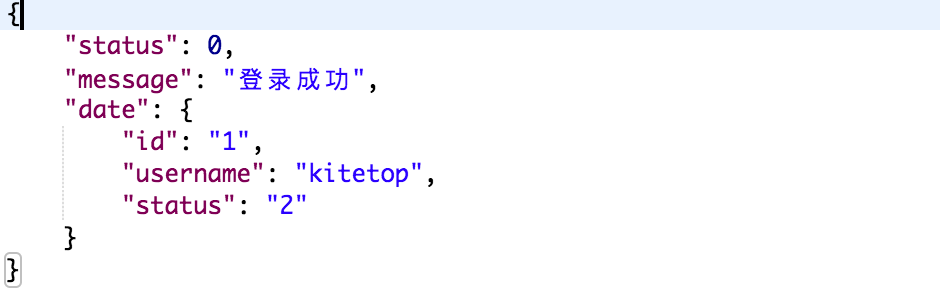
|  |  |  |
| --- | --- | --- |
| 字段 | 数据类型 | 描述 |
| id | int | 主键 |
| response\_id | int | 自己被回复的留言id（等于response表中的response\_id） |
| notice\_id | int | 别人回复的留言id |
| user\_response | mediumint | 留言回复者id |
| user\_accept | mediumint | 留言接收者id(检索条件，确定通知的人) |
| view | tinyint | 是否查看留言（2:查看 1:未查看） |

* 接口设计
* 用户登录

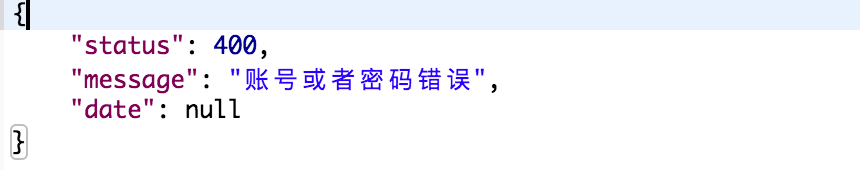
method = post url = /login

|  |  |
| --- | --- |
| 参数 | 说明 |
| account | （必须）用户账号 |
| password | （必须）用户密码 |

1. 登录成功



1. 账号或者密码不存在

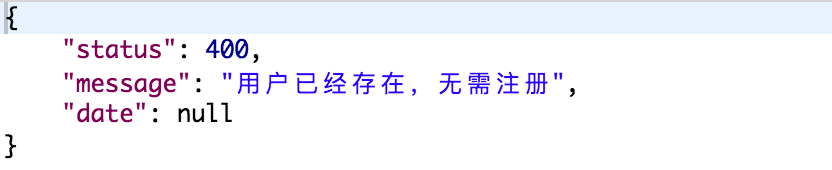


* 用户注册

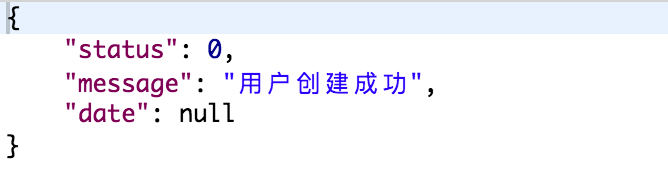
method = post url = /user/register

|  |  |
| --- | --- |
| 参数 | 说明 |
| account | （必须）登录账号 |
| username | （必须）用户昵称 |
| password | （必须）登录密码 |
| image | （必须）用户头像 |
| sex | （必须）用户性别 |

1. 账号重复注册



1. 注册成功

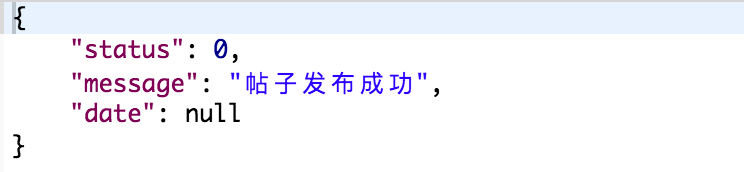


* 发布帖子

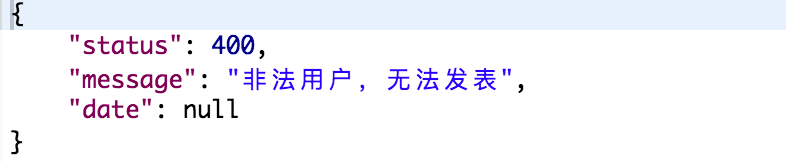
method = post url = /theme/add

|  |  |
| --- | --- |
| 参数 | 说明 |
| id | （必须）发布帖子的用户的id |
| title | （必须）帖子的标题 |
| context | （必须）帖子的内容 |

1. 发布成功



1. 非法用户

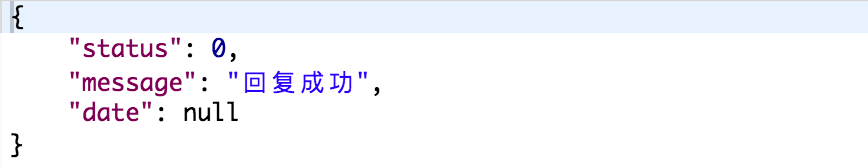


* 回复

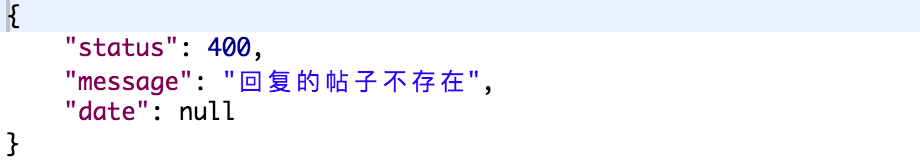
method = post url = /response/add

|  |  |
| --- | --- |
| 参数 | 说明 |
| userId | （必须）回复者的用户id |
| themeId | （必须）回复留言所属的帖子id |
| type | （必须）回复留言的内容的形式，0代表文本 1 代表图片 |
| text | （和image两者选一）回复的内容 |
| image | （和text两者选一）回复的内容 |
| fatherId | （可选）父楼的id |
| responseId | （可选）回复的留言的id |

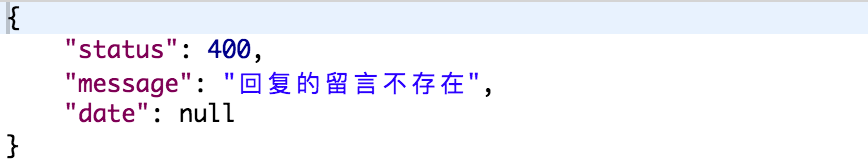
1. 回复成功



1. 帖子不存在



1. 回复留言不存在

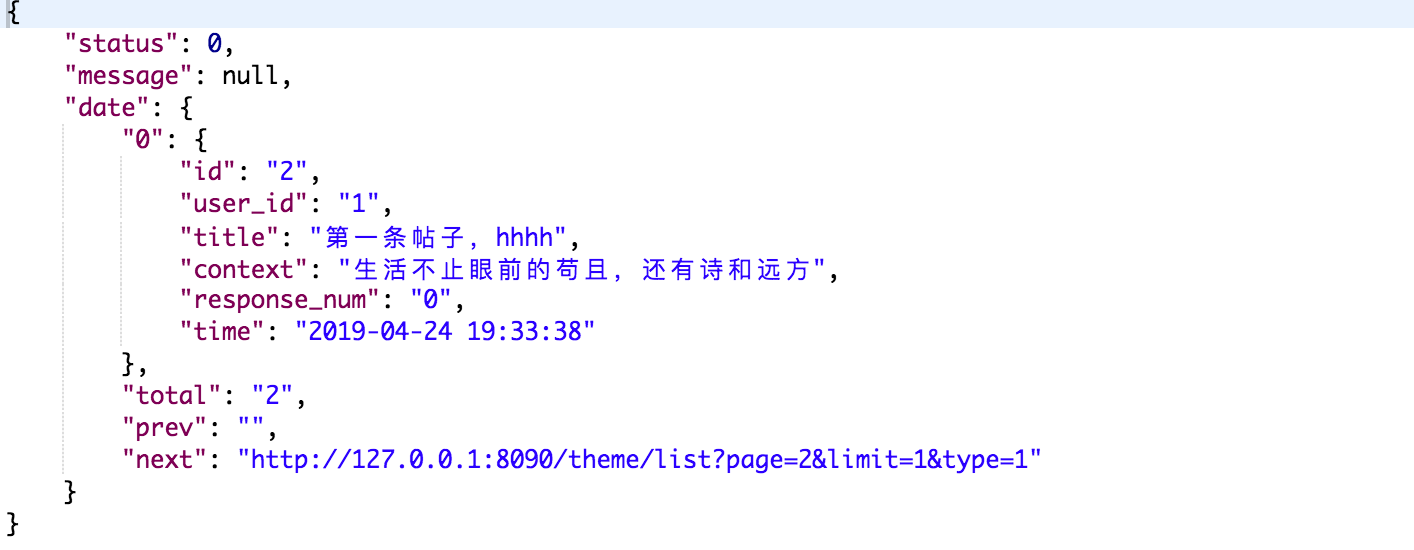


* 查看帖子列表

method = get url = /theme/list

|  |  |
| --- | --- |
| 参数 | 说明 |
| type | （必须）对帖子排行的区别 0 最活跃 1 最新 |
| page | （可选）当前的页码 |
| limit | （可选）每一页显示的条数 |

1. 查询成功

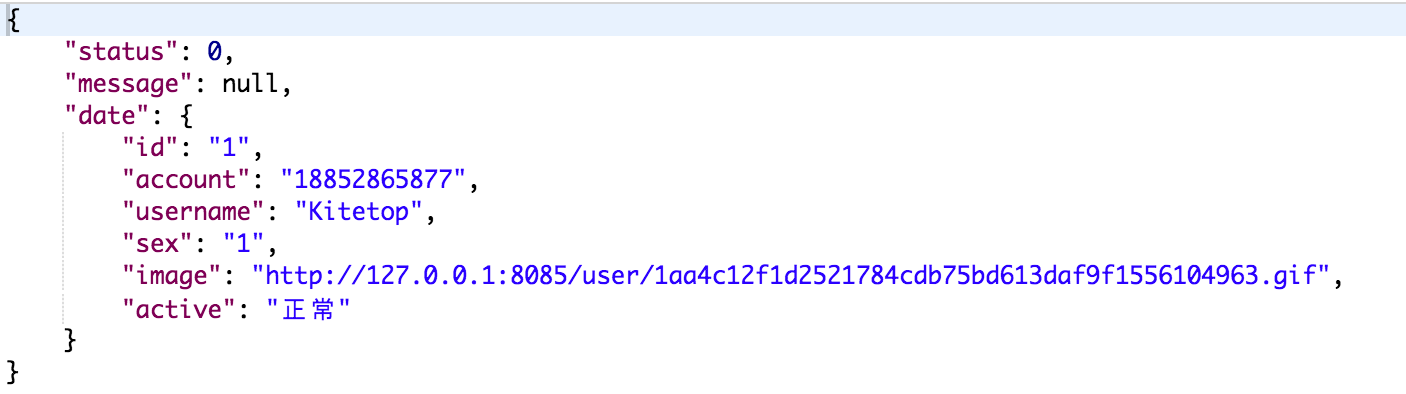


* 查看用户信息

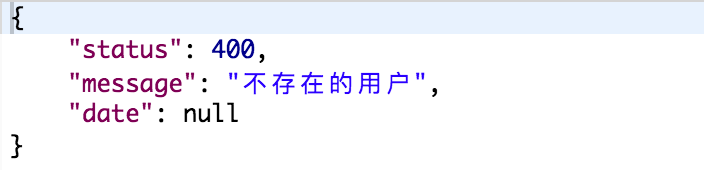
method = get url = /user/info

|  |  |
| --- | --- |
| 参数 | 说明 |
| id | （必须）想要查询的用户的id |

1. 查询成功



1. 用户不存在

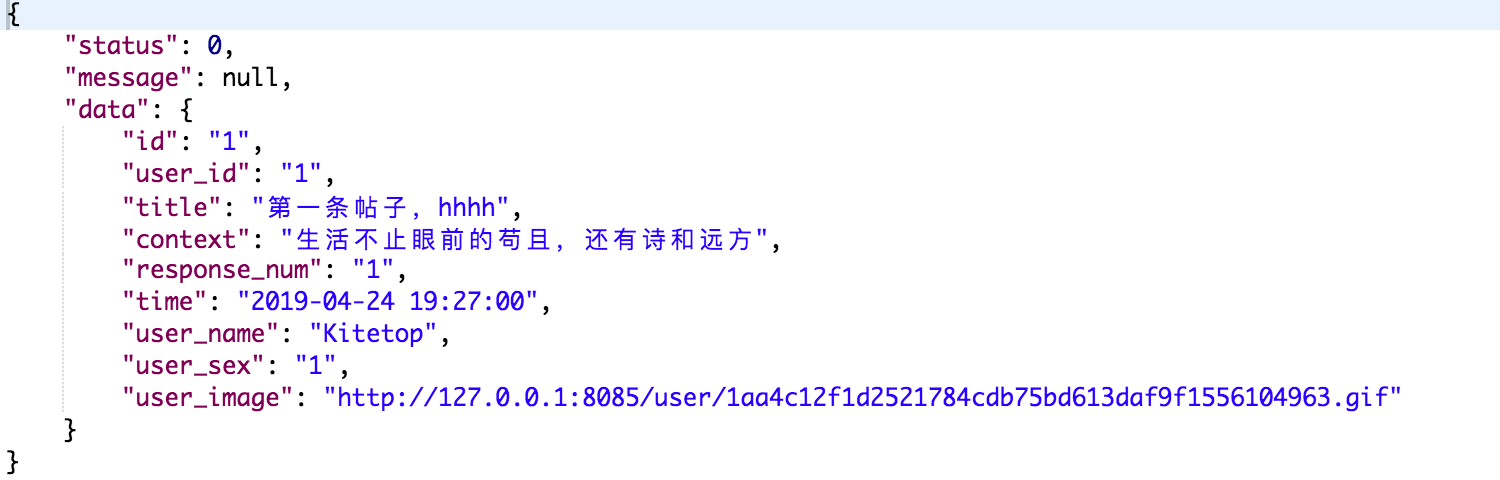


* 查询帖子详情

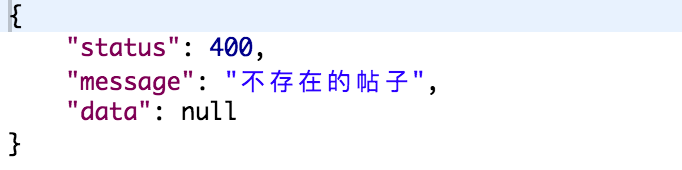
method = get url = /theme/details

|  |  |
| --- | --- |
| 参数 | 说明 |
| id | （必须）想要查询的帖子的id |

1. 查询成功



1. 帖子不存在



* 获得帖子下的直系留言

method = get url = /response/flist

|  |  |
| --- | --- |
| 参数 | 说明 |
| id | （必须）想要查询的帖子的id |

1. 查询成功



* 获得父楼下的子留言

method = get url = /response/clist

|  |  |
| --- | --- |
| 参数 | 说明 |
| id | （必须）想要查询的父楼留言的id |

1. 查找成功

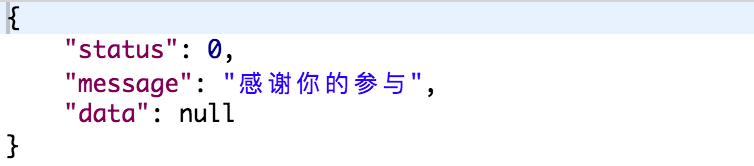


* 给留言点赞或者举报留言

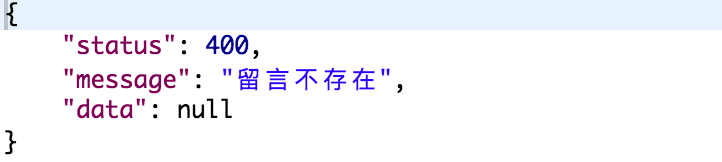
method = get url = /response/action

|  |  |
| --- | --- |
| 参数 | 说明 |
| id | （必须）留言的id |
| type | （必须）0 点赞 1 举报 |

1. 成功



1. 留言不存在

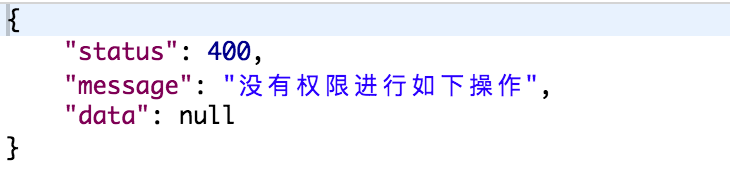


* 对用户进行封禁或者解封

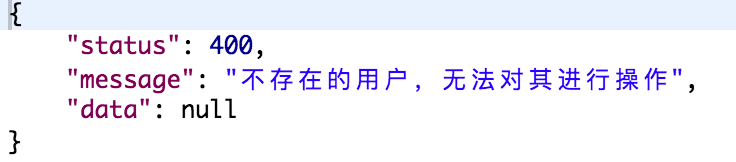
method = get url = /admin/active

|  |  |
| --- | --- |
| 参数 | 说明 |
| id | （必须）进行封禁 解封操作的管理员id |
| type | （必须）进行操作的类型 0 解封 1 封禁 |
| userId | （必须）执行封禁或者解封的用户id |

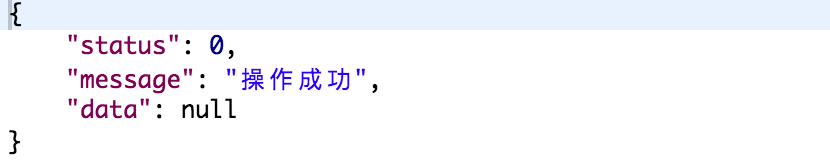
1. 不是管理员进行操作



1. 用户不存在



1. 成功



* 查看近期被举报最多的留言

method = get url = /report/list

|  |  |
| --- | --- |
| 参数 | 说明 |
| id | （必须）进行查看举报留言的管理员id |
| type | （必须）0 最近1天 1 最近一个礼拜 2 最近一个月 |
| page | （可选）要查看的当前页数 |
| limit | （可选）每页限制条数 |

1. 查看成功

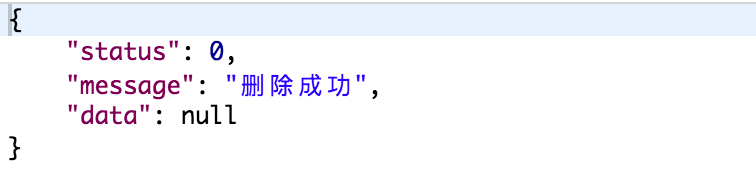


* 删除留言

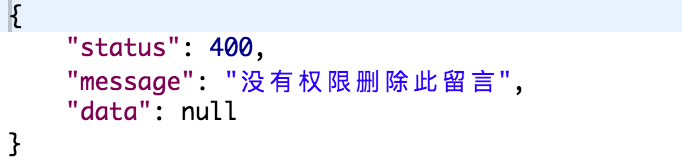
method = get url = /response/delete

|  |  |
| --- | --- |
| 参数 | 说明 |
| id | （必须）想要删除的留言的id |
| userId | （必须）进行删除操作的用户id |

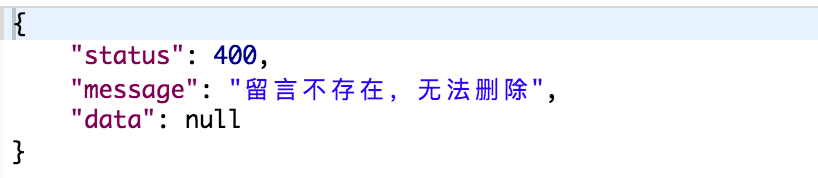
1. 删除成功



1. 飞管理员删除不是本人的留言



1. 留言不存在



* 删除帖子

method = get url = /theme/delete

|  |  |
| --- | --- |
| 参数 | 说明 |
| id | （必须）想要删除的贴子的id |
| userId | （必须）进行删除操作的用户id |