



链滴

社区客户端 API 文档

作者: [88250](#)

原文链接: <https://ld246.com/article/1488603534762>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

概述

该文档主要面向社区客户端开发者，如果你想开发一个移动端 APP 的话请仔细阅读该文档。如果你喜欢写博客，想要将博客和社区进行联动的话请参考[内容 API 开放](#)，[欢迎各位独立博客主进行连接](#)。

目前以下列出的 API 已经可用，大家有什么想补充的请跟帖。

使用原则

- 欢迎用于 APP、插件、数据分析研究等
- 禁止用于以内容填充为目的的应用或站点

公共约定

对于请求和响应的一些公共约定在这里统一进行描述。

HTTP

- 数据使用 UTF-8 编码
- 参数可能是查询字符串也可能是通过 body 传递
- 响应的 Status Code 可能是 **200** 也可能是其他值，客户端需要校验
 - **401**：需要登录
 - **403**：权限不足

User-Agent

不要使用 HTTP 客户端自带的 UA，请**一定要自定义 UA**，推荐格式 `{App}/{Ver}`，比如 `Solo/2.9.5`。如果不自定义 UA，请求将会被社区的流量监控系统阻断，并且对发起请求的 IP 进行一定时间的拦截。

鉴权

服务端目前有两种鉴权方式：

1. 使用 HTTP 标头 `Authorization` 进行验证，值为 `token {你的 API Token}`，可在 [设置 - 账号](#) 中到 API Token
2. 使用 HTTP Cookie `symphony` 进行验证，建议不要使用，该方式仅用于兼容旧的客户端，未来移除支持

列表分页

请求：

- 使用查询字符串 `p` 作为当前页号
- 服务端固定页大小为 **20**

响应:

返回 `pagination` 对象, 包含两个字段:

- `paginationPageCount`: 分页总页数
- `paginationPageNums`: 以参数 `p` 为中间的窗口页码列表, 窗口最大宽度为 15

响应结构

响应中的 HTTP body 为 JSON 结构, 固定包含 3 个字段:

```
{  
  "code": 0,  
  "msg": "",  
  "data": null  
}
```

其中:

- `code`: 类型是 `int`, 不会为 `null`, 表示状态码, 其值请参考 [StatusCode.java](#)
- `msg`: 类型是 `string`, 不会为 `null` (默认是 "" 空字符串), 表示消息提示
- `data`: 类型是 `JSONObject` 或者 `JSONArray`, 可能为 `null`, 表示返回的数据

注意事项

- 使用 HTTPS 协议
- 所有 API 都会做 **是否允许非登录请求**的校验, 如果返回 `401` 则请带上 Token
- 请勿频繁调用, 每个 IP 每分钟请求数不能超过 120 次, 每个 IP 只能建立一个连接

帖子

获取最新帖子列表

GET 方法:

- 按发布时间排序: <https://ld246.com/api/v2/articles/latest?p=1>
- 按热议排序: <https://ld246.com/api/v2/articles/latest/hot?p=1>
- 按好评排序: <https://ld246.com/api/v2/articles/latest/good?p=1>
- 按最近回帖排序: <https://ld246.com/api/v2/articles/latest/reply?p=1>
- 按优选排序: <https://ld246.com/api/v2/articles/latest/perfect?p=1>

获取领域帖子列表

- GET 方法: <https://ld246.com/api/v2/articles/domain/{domainURI}?p=1>
- 示例: <https://ld246.com/api/v2/articles/domain/Java?p=1>

获取标签帖子列表

- GET 方法: <https://ld246.com/api/v2/articles/tag/{tagURI}?p=1>

示例:

- 按发布时间排序: <https://ld246.com/api/v2/articles/tag/Java?p=1>
- 按热议排序: <https://ld246.com/api/v2/articles/tag/Java/hot?p=1>
- 按好评排序: <https://ld246.com/api/v2/articles/tag/Java/good?p=1>
- 按优选排序: <https://ld246.com/api/v2/articles/tag/Java/perfect?p=1>
- 按最近回帖排序: <https://ld246.com/api/v2/articles/tag/Java/reply?p=1>

获取帖子详情

- GET 方法: <https://ld246.com/api/v2/article/{articleId}?p=1>, 分页参数 `p` 是回帖的
- 示例: <https://ld246.com/api/v2/article/1488603534762?p=1>

发布帖子

- POST 方法: <https://ld246.com/api/v2/article>
- Body:

```
{
  "articleTitle": "",
  "articleTags": "", // 用英文逗号分隔
  "articleContent": "",
  "articleRewardContent": "" // 打赏区内容
  "articleRewardPoint": int // 打赏积分
}
```

获取帖子详情用于更新

- GET 方法: <https://ld246.com/api/v2/article/update/{articleId}>
- 示例: <https://ld246.com/api/v2/article/update/1488603534762>

更新帖子

- PUT 方法: <https://ld246.com/api/v2/article/{articleId}>
- Body:

```
{
  "articleTitle": "",
  "articleTags": "", // 用英文逗号分隔
  "articleContent": "",
  "articleType": int, // 帖子类型, 按获取帖子后的值传入即可
  "articleRewardContent": "" // 打赏区内容
  "articleRewardPoint": int // 打赏积分
}
```

```
}
```

回帖

发布回帖

- POST 方法: <https://ld246.com/api/v2/comment>
- Body:

```
{  
  "articleId": "",  
  "commentContent": "",  
  "commentOriginalCommentId": "", // 可选, 如果是回复则传入原回帖 id  
}
```

领域

获取领域列表

- GET 方法: <https://ld246.com/api/v2/domains?p=1>

获取领域详情

- GET 方法: <https://ld246.com/api/v2/domain/{domainURI}>
- 示例: <https://ld246.com/api/v2/domain/Java>

标签

获取标签列表

- GET 方法: <https://ld246.com/api/v2/tags?p=1>

获取标签详情

- GET 方法: <https://ld246.com/api/v2/tag/{tagURI}>
- 示例: <https://ld246.com/api/v2/tag/Java>

用户

获取当前登录用户详情

- GET 方法: <https://ld246.com/api/v2/user>
- 示例: <https://ld246.com/api/v2/user>

根据用户名获取用户详情

- GET 方法: <https://ld246.com/api/v2/user/{userName}>
- 示例: <https://ld246.com/api/v2/user/88250>

根据用户序号获取用户详情

- GET 方法: <https://ld246.com/api/v2/user/n/{userNo}>
- 示例: <https://ld246.com/api/v2/user/n/1>

获取用户帖子列表

- GET 方法: <https://ld246.com/api/v2/user/{userName}/articles?p=1>
- 示例: <https://ld246.com/api/v2/user/88250/articles?p=1>

获取用户回帖列表

- GET 方法: <https://ld246.com/api/v2/user/{userName}/comments?p=1>
- 示例: <https://ld246.com/api/v2/user/88250/comments?p=1>

获取用户近期动态列表

- GET 方法: <https://ld246.com/api/v2/user/{userName}/events?size=16>
- 示例: <https://ld246.com/api/v2/user/88250/events?size=16>

`size` 为获取条数, 最小为 1, 最大为 64, 不传该参数则默认为 16。

获取用户关注帖子列表

- GET 方法: <https://ld246.com/api/v2/user/{userName}/watching/articles?p=1>
- 示例: <https://ld246.com/api/v2/user/88250/watching/articles?p=1>

获取用户关注用户列表

- GET 方法: <https://ld246.com/api/v2/user/{userName}/following/users?p=1>
- 示例: <https://ld246.com/api/v2/user/88250/following/users?p=1>

获取用户关注标签列表

- GET 方法: <https://ld246.com/api/v2/user/{userName}/following/tags?p=1>
- 示例: <https://ld246.com/api/v2/user/88250/following/tags?p=1>

获取用户收藏帖子列表

- GET 方法: <https://ld246.com/api/v2/user/{userName}/following/articles?p=1>
- 示例: <https://ld246.com/api/v2/user/88250/following/articles?p=1>

获取用户关注者列表

- GET 方法: <https://ld246.com/api/v2/user/{userName}/followers?p=1>
- 示例: <https://ld246.com/api/v2/user/88250/followers?p=1>

通知

获取未读消息计数

- GET 方法: <https://ld246.com/api/v2/notifications/unread/count>
- 示例: <https://ld246.com/api/v2/notifications/unread/count>

获取收到的回帖消息列表

- GET 方法: <https://ld246.com/api/v2/notifications/commented?p=1>
- 示例: <https://ld246.com/api/v2/notifications/commented?p=1>

获取收到的评论消息列表

- GET 方法: <https://ld246.com/api/v2/notifications/comment2ed?p=1>
- 示例: <https://ld246.com/api/v2/notifications/comment2ed?p=1>

获取收到的回复消息列表

- GET 方法: <https://ld246.com/api/v2/notifications/reply?p=1>
- 示例: <https://ld246.com/api/v2/notifications/reply?p=1>

获取提及我的消息列表

- GET 方法: <https://ld246.com/api/v2/notifications/at?p=1>
- 示例: <https://ld246.com/api/v2/notifications/at?p=1>

获取我关注的消息列表

- GET 方法: <https://ld246.com/api/v2/notifications/following?p=1>
- 示例: <https://ld246.com/api/v2/notifications/following?p=1>

获取我的积分消息列表

- GET 方法: <https://ld246.com/api/v2/notifications/point?p=1>

- 示例: <https://ld246.com/api/v2/notifications/point?p=1>

获取同城广播消息列表

- GET 方法: <https://ld246.com/api/v2/notifications/broadcast?p=1>
- 示例: <https://ld246.com/api/v2/notifications/broadcast?p=1>

获取钱包消息列表

- GET 方法: <https://ld246.com/api/v2/notifications/wallet?p=1>
- 示例: <https://ld246.com/api/v2/notifications/wallet?p=1>

获取系统公告消息列表

- GET 方法: <https://ld246.com/api/v2/notifications/sys-announce?p=1>
- 示例: <https://ld246.com/api/v2/notifications/sys-announce?p=1>

标记消息列表为已读

- GET 方法: <https://ld246.com/api/v2/notifications/make-read/{type}>

其中 {type} 是消息类型:

- **commented**: 收到的回帖, 包括合并回帖
- **comment2ed**: 收到的评论
- **reply**: 收到的回复
- **at**: 提及我的
- **following**: 我关注的

同城广播、钱包和系统公告消息拉取后会被自动标记为已读状态, 无需调用此 API 进行标记已读。

通知关联的数据类型说明

每条获取到的消息都会有 **dataType** 字段, 表示与该消息关联的数据类型。

```
<details>
<summary>数据类型说明</summary>
```

dataType	dataId	备注
-1	没有该字段	关联数据已被删除
0		暂未使用
1		暂未使用
2	帖子 id 或回帖 id	此处为设计

陷, 建议用 **dataId** 先查回帖, 为空的话再查帖子

3	回帖 id	
4	帖子 id	关注的用户发新贴
5	积分转账 id	转账记录会关联
账类型, 目前暂时没有接口查询, 下同		
6	积分转账 id	
7	打赏 id	打赏记录会关联打赏
型, 目前暂时没有接口查询, 下同		
8	感谢 id	感谢记录会关联感谢
型, 目前暂时没有接口查询, 下同		
9	帖子 id	
10	积分转账 id	
11	积分转账 id	
12	感谢 id	
13	回复 id	
14	用户 id	
15	帖子 id	
16	帖子 id	
17	用户 id	
18	用户 id	
19	老角色 id - 新角色 id	
20	帖子 id	
21	回帖 id	
22	帖子 id	
23	帖子 id - 用户 id	
24	帖子 id - 用户 id	
25	回帖 id - 用户 id	
26	回帖 id - 用户 id	
27	帖子 id - 用户 id	
28	帖子 id - 用户 id	
29	货币转账 id	
30	货币转账 id	
31	货币转账 id	
32	货币转账 id	
33	回帖 id	
34	回帖 id	合并后所有的回帖 id
需要通过 data 字段获取		
35	回帖 id	同 34
36	积分转账 id	

37	聊天会话 id
38	评论 id - 用户 id
39	感谢 id
40	评论 id
41	评论 id
42	帖子 id
43	帖子 id
44	回帖 id
45	回帖 id
46	评论 id
47	评论 id
48	帖子 id
49	回帖 id
50	评论 id
51	帖子 id
52	回帖 id
53	评论 id
54	积分转账 id

</details>

榜单

获取新注册用户列表

- GET 方法: <https://ld246.com/api/v2/tops/users/newbies>
- 示例: <https://ld246.com/api/v2/tops/users/newbies>

获取粉丝榜

- GET 方法: <https://ld246.com/api/v2/tops/users/followers>
- 示例: <https://ld246.com/api/v2/tops/users/followers>

登录

密码验证

- POST 方法: <https://ld246.com/api/v2/login>
- Body:

```
{
```

```
"userName": "", // 用户名或者邮箱
"userPassword": "", // 使用 MD5 哈希后的值
"captcha": "" // 正常登录不用带该字段，登录失败次数过多时必填
}
```

- 返回:

```
{
  "code": 0, // 0 为成功, 10 为需要两步验证
  "msg": "",
  "token": "", // 成功时才有该值
  "userName": "", // 用户名
  "needCaptcha": "" // 登录失败次数过多会返回该值
}
```

登录成功后会返回 `token`，正常情况下请勿使用该值，除非需要进行两步验证。如果需要填验证码，使用 `needCaptcha` 返回的值作为参数 GET 请求一次 `/captcha/login?needCaptcha={needCaptcha}` 返回的图片就是验证码了。

两步验证

- POST 方法: <https://ld246.com/api/v2/login/2fa>
- Cookie: `Cookie: symphony={token}`，请求 Cookie 需要带 `symphony` 项，值为 login 接口回的 `token`
- Body:

```
{
  "twofactorAuthCode": "" // 身份验证器动态口令验证码，6 位纯数字
}
```

- 返回:

```
{
  "code": 0, // 0 为成功
  "msg": "",
  "token": "", // 成功时才有该值
  "userName": "" // 用户名
}
```

登出

- POST 方法: <https://ld246.com/api/v2/logout>

系统

获取系统当前时间

- GET 方法: <https://ld246.com/api/v2/system/time>
- 示例: <https://ld246.com/api/v2/system/time>