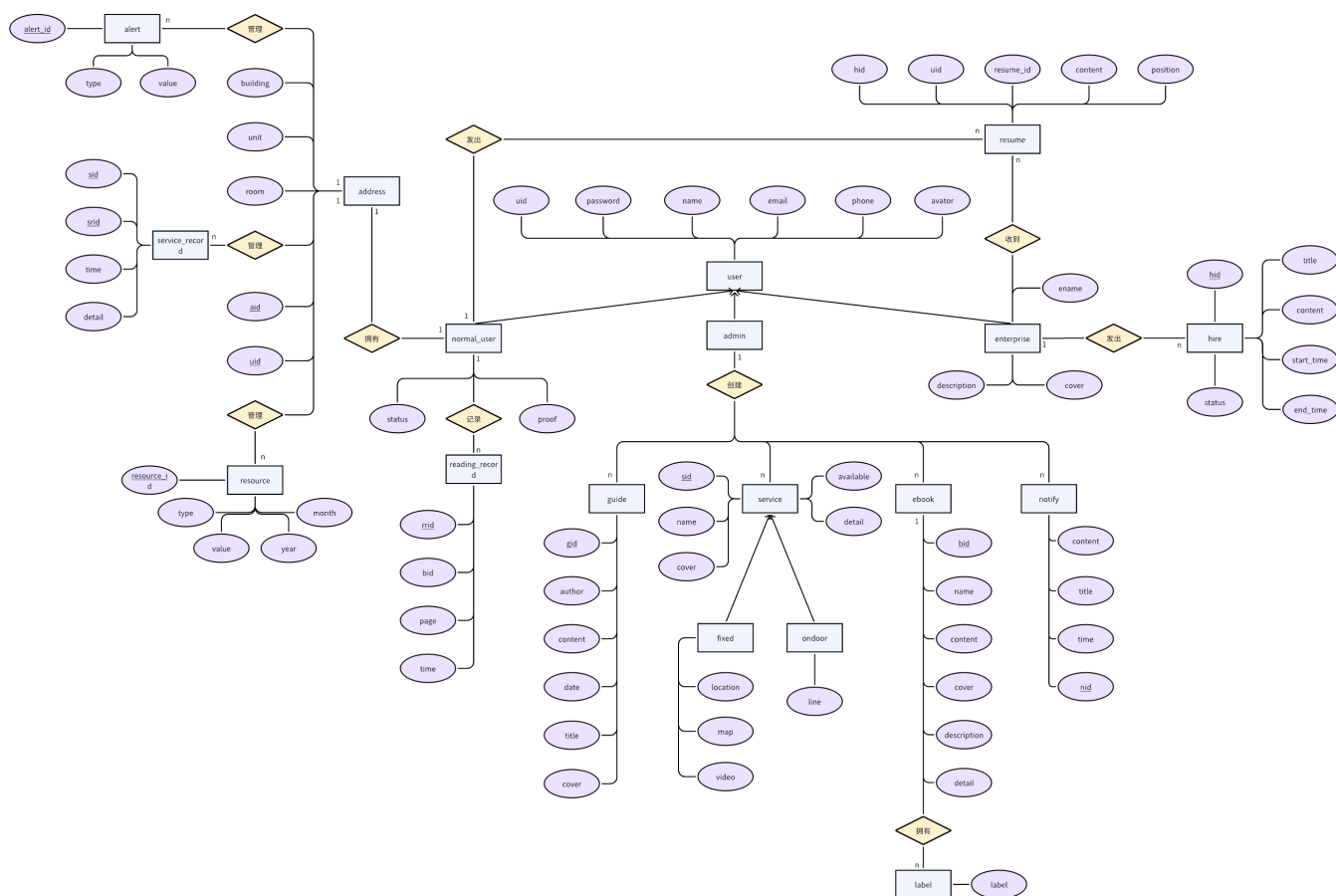


api接口



数据定义



Unauthorized

未授权



402

Payment Required

需付费



406

Not Acceptable

无法接受



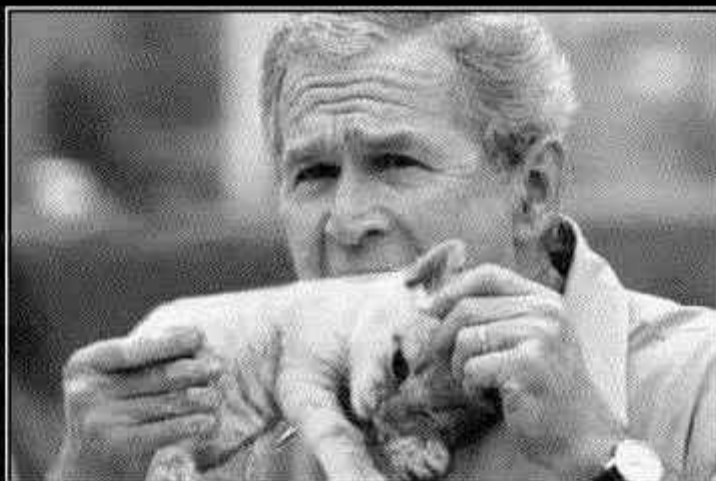
403

Forbidden
禁止访问



404

Not Found
无法找到



405

Method Not Allowed
方法不被允许





408

Request Timeout

请求超时



409

Conflict

指令冲突



413

Request Entity Too Large

请求实体过大





414

Request-URI Too Long

所请求的URL地址过长



416

Requested Range Not Satisfiable

请求的范围无法被满足



417

Expectation Failed
预期内容错误



426

Upgrade Required
需升级



429

Too Many Requests
请求数过多



431

Request Header Fields Too Large

请求的头字段过大



450

Blocked by Windows Parental Controls

已被Windows家长控制程序屏蔽



500

Internal Server Error

内部服务器错误





507

Insufficient Storage

存量不足



599

Network connect timeout error

错误：网络连接超时



200

OK





```
1 // 状态码
2 StatusCode {
3     200: 成功
4     404: 失败
5 }
6
7 // 用户类型
8 UserType {
9     NORMAL: 普通用户
10    ADMIN: 物业
11    ENTERPRISE: 企业用户
12 }
13
14 // 服务类型
15 ServiceType {
16     FIXED: 固定服务（健身器材啥的）
17     ONDOOR: 上门服务
18 }
19
20 // 资源类型
21 ResourceType {
22     不知道有啥类型
23     ELETRIC,
24     WATER,
25     GAS
26 }
27
28 // 书标签类型
29 LabelType {
30     不知道有啥类型
31 }
```

约定

- GET /users: 返回用户资源的列表
- GET /users/:id: 返回单个用户资源
- POST /users: 返回新生成的用户资源
- PUT /users/:id: 返回完整的用户资源
- PATCH /users/:id: 返回完整的用户资源
- DELETE /users/:id: 返回一个空文档,或者状态码 204

术语

- 1 审核状态: "approved" (已通过, 字符串), "pending" (审核中, 空值), "reason" (不过审, 原因信息);
- 2 用户不通过的内容可修改后重新发起审核请求, 招聘不能重新审核
- 3
- 4 户主认定: 房间第一个被认证用户
- 5
- 6 用户头像、电子书封面为图片格式
- 7
- 8 物业用户与企业用户只能由物业用户创建, 第一个物业用户写死在数据库
- 9
- 10 对于企业用户, user里面的name是hr名字, enterprise的ename是公司名称

接口

接口例子

```

1 // 目前考虑会用到GET,POST,DELETE,PATCH
2 // 改资源某一字段用PATCH
3 Method: GET
4 // 参数可能不存在, 这里以get请求单个资源
5 // 比如获取一个请求地址里关于房屋信息的资源,所有资源名称是复数
6 Path: /addresses/:aid
7 // 发送请求传参json
8 {
9     Authorization: ewoifnweiofwifjwi
10    params:{
11        // 实际api的字段
12        // 例如PATCH用户性别
13        sex: 0, 1, man, weman,,,,

```

```

14     },
15 }
16
17 // 返回参数json
18 {
19     code: statusCode
20     // 后端处理后的资源
21     data?: {
22         aid:
23         unit:
24         building:
25         host_nameame:
26     }
27     message?: "dadasdadadada"
28 }
29

```

用户

```

1  /***** 用户 *****/
2  // 注册普通用户
3  Method: POST
4  Path: /register/normal
5  {
6      Authorization: null
7      params:{
8          name: string;
9          password: string;
10         email: string;
11         phone: string;
12         building: string;
13         unit: string;
14         room: string;
15         proof: string;
16         avator: string;
17     },
18 }
19
20 {
21     code: StatusCode
22     data?: {
23         jwt: string;
24         uid: string;
25         type: UserType;
26     }

```

```
27     message?: string
28 }
29
30 // 注册物业用户
31 Method: POST
32 Path: /register/admin
33 {
34     Authorization: jwt
35     params:{
36         name: string;
37         password: string;
38         email: string;
39         phone: string;
40         avator: string;
41     },
42 }
43
44 {
45     code: StatusCode
46     data?: {
47         name: string;
48         avator: string;
49     }
50     message?: string
51 }
52
53 // 注册企业用户
54 Method: POST
55 Path: /register/enterprise
56 {
57     Authorization: jwt
58     params:{
59         name: string;
60         ename: string;
61         description: string;
62         cover: string;
63         password: string;
64         email: string;
65         phone: string;
66         avator: string;
67     },
68 }
69
70 {
71     code: StatusCode
72     data?: {
73         name: string;
```



```
74     ename: string;
75     avator: string;
76 }
77 message?: string
78 }
79
80 // 用户登陆
81 Method: POST
82 Path: /login
83 {
84     Authorization:
85     params:{
86         // 可能是email也可能是phone, 后端判断一下
87         account?: string;
88         password?: string;
89     },
90 }
91
92 {
93     code: StatusCode
94     data?: {
95         jwt: string;
96         uid: string;
97         // 用户类型
98         type: UserType;
99     }
100     message?: string
101 }
102
103 // 获取用户信息
104 Method: GET
105 Path: /users/:uid
106 {
107     Authorization: jwt
108     params:{
109     },
110 }
111
112 {
113     code: StatusCode
114     data?: {
115         uid: string;
116         name: string;
117         email: string;
118         phone: string;
119         avator: string;
120         // 后两个针对不同类型用户, 没有就为空
```

```
121         status: string;
122         ename: string;
123         // 对于普通用户
124         building: string;
125         room: string;
126         unit: string;
127     }
128     message?: string
129 }
130
131 // 删除用户
132 Method: DELETE
133 Path: /users/:uid
134 {
135     Authorization: jwt
136     params:{
137     },
138 }
139
140 {
141     code: StatusCode
142     data?: {
143         uid: string;
144     }
145     message?: string
146 }
147
148 // 修改用户信息 (只有管理员能修改status)
149 Method: PATCH
150 Path: /users/:uid
151 {
152     Authorization: jwt
153     params:{
154         name?: string;
155         email?: string;
156         phone?: string;
157         avator?: string;
158         ename?: string;
159         proof?: string;
160         password_pre?: string;
161         password_cur?: string;
162         status?: string;
163         // 对于普通用户
164         building: string;
165         room: string;
166         unit: string;
167     },
```

```

168 }
169
170 {
171     code: StatusCode
172     data?: {
173         uid: string;
174         name: string;
175         email: string;
176         phone: string;
177         avator: string;
178         // 后两个针对不同类型用户，没有就为空
179         status: string;
180         ename: string;
181         // 对于普通用户
182         building: string;
183         room: string;
184         unit: string;
185     }
186     message?: string
187 }
188
189 // 查看所有用户（分页）
190 Method: GET
191 Path: /users/all
192 {
193     Authorization: jwt
194     params:{
195         // 一页多少数据
196         offset: number;
197         // 当前所在页数
198         pageNum: number;
199     },
200 }
201
202 {
203     code: StatusCode
204     data?: {
205         users:[{
206             uid: string;
207             name: string;
208             // 后两个针对不同类型用户，没有就为空
209             status: string;
210             ename: string;
211         }, {...}, {...}],
212         // 一共多少页
213         allPages: number
214     }

```

```
215     message?: string
216 }
217
218 // 重新发起审核
219 // 审核状态变为pending
220 Method: PATCH
221 Path: /users/re_verify/:uid
222 {
223     Authorization: jwt
224     params:{
225     },
226 }
227 {
228     code: StatusCode
229     data?: {
230         uid: string;
231     }
232     message?: string
233 }
234
235 // 查看所有用户 (分类)
236 Method: GET
237 Path: /users/diff_type
238 {
239     Authorization: jwt
240     params:{
241         type: UserType;
242         // 一页多少数据
243         offset: number;
244         // 当前所在页数
245         pageNum: number;
246     },
247 }
248
249 {
250     code: StatusCode
251     data?: {
252         users:[{
253             uid: string;
254             name: string;
255             avator: string;
256             // 没有为空
257             ename: string;
258         }, {...}, {...}],
259         // 一共多少页
260         allPages: number
261     }
```



```
262     message?: string
263 }
```

指南

```
1  /***** 指南 *****/
2  // 添加指南
3  Method: POST
4  Path: /guides
5  {
6      Authorization: jwt
7      params:{
8          author: string;
9          title: string;
10         content: string;
11         date: string;
12         cover: string;
13     },
14 }
15
16 {
17     code: StatusCode
18     data?: {
19         gid: string;
20     }
21     message?: string
22 }
23
24 // 删除指南
25 Method: DELETE
26 Path: /guides/:gid
27 {
28     Authorization: jwt
29     params:{
30     },
31 }
32
33 {
34     code: StatusCode
35     data?: {
36         gid: string;
37     }
38     message?: string
39 }
40
```

```
41 // 修改指南
42 Method: PATCH
43 Path: /guides/:id
44 {
45     Authorization: jwt
46     params:{
47         author? : string;
48         title?: string;
49         content? : string;
50         date? : string;
51         cover?: string;
52     },
53 }
54
55 {
56     code: StatusCode
57     data?: {
58         gid: string;
59         author: string;
60         title: string;
61         content: string;
62         date: string;
63         cover: string;
64     }
65     message?: string
66 }
67
68 // 查看所有指南
69 Method: GET
70 Path: /guides
71 {
72     Authorization: jwt
73     params:{
74         // 一页多少数据
75         offset: number;
76         // 当前所在页数
77         pageNum: number;
78     },
79 }
80
81 {
82     code: StatusCode
83     data?:{guides: [{
84         gid: string;
85         author: string;
86         title: string;
87         // 节选, 大概几十字
```

```

88         content_slice: string;
89         cover: string;
90         date: string;
91     }, {...}, {...}],allPages:number}
92     message?: string
93 }
94
95 // 查看单个指南
96 Method: GET
97 Path: /guides/:gid
98 {
99     Authorization: jwt
100     params:{
101     },
102 }
103
104 {
105     code: StatusCode
106     data?: {
107         gid: string;
108         author: string;
109         title: string;
110         content: string;
111         date: string;
112         cover: string;
113     }
114     message?: string
115 }

```

服务

```

1  /***** 服务 *****/
2  // 添加服务
3  Method: POST
4  Path: /services
5  {
6      Authorization: jwt
7      params:{
8          name: string;
9          cover: string;
10         available: string;
11         detail: string;
12         type: ServiceType;
13         // 针对不同服务类型
14         location: string;

```

```
15     map: string;
16     video: string;
17     line: string;
18 },
19 }
20
21 {
22     code: StatusCode
23     data?: {
24         sid: string;
25     }
26     message?: string
27 }
28
29 // 删除服务
30 Method: DELETE
31 Path: /services/:sid
32 {
33     Authorization: jwt
34     params:{
35     },
36 }
37
38 {
39     code: StatusCode
40     data?: {
41         sid: string;
42     }
43     message?: string
44 }
45
46 // 修改服务
47 Method: PATCH
48 Path: /services/:id
49 {
50     Authorization: jwt
51     params:{
52         name?: string;
53         cover?: string;
54         available?: string;
55         detail?: string;
56         type?: ServiceType;
57         // 针对不同服务类型
58         location?: string;
59         map?: string;
60         video?: string;
61         line?: string;
```



```
62     },
63 }
64
65 {
66     code: StatusCode
67     data?: {
68         sid: string;
69         name: string;
70         cover: string;
71         available: string;
72         detail: string;
73         type: ServiceType;
74         // 针对不同服务类型
75         location: string;
76         map: string;
77         video: string;
78         line: string;
79     }
80     message?: string
81 }
82
83 // 查看不同服务
84 Method: GET
85 Path: /services
86 {
87     Authorization: jwt
88     params:{
89         type: ServiceType;
90         // 一页多少数据
91         offset: number;
92         // 当前所在页数
93         pageNum: number;
94     },
95 }
96
97 {
98     code: StatusCode
99     data?: {services:[{
100         sid: string;
101         name: string;
102         cover: string;
103         available: string;
104         // 切片
105         detail_slice: string;
106     }], {...}, {...]]}
107     message?: string
108 }
```

```
109
110 // 查看全部不同服务
111 Method: GET
112 Path: /services/all
113 {
114     Authorization: jwt
115     params:{
116         // 一页多少数据
117         offset: number;
118         // 当前所在页数
119         pageNum: number;
120     },
121 }
122
123 {
124     code: StatusCode
125     data?: {services:[{
126         sid: string;
127         name: string;
128         cover: string;
129         available: string;
130         // 切片
131         detail_slice: string;
132     }, {...}, {...}], allPages:number}
133     message?: string
134 }
135
136 // 查看单个服务
137 Method: GET
138 Path: /services/:sid
139 {
140     Authorization: jwt
141     params:{
142     },
143 }
144
145 {
146     code: StatusCode
147     data?: {
148         sid: string;
149         name: string;
150         cover: string;
151         available: string;
152         detail: string;
153         type: ServiceType;
154         // 针对不同服务类型
155         location: string;
```

```
156         map: string;
157         video: string;
158         line: string;
159     }
160     message?: string
161 }
162
163
164 // 订阅服务
165 Method: POST
166 Path: /services/subscribe
167 {
168     Authorization: jwt
169     params:{
170         sid: string;
171         time: string;
172         detail: string;
173     },
174 }
175
176 {
177     code: StatusCode
178     data?: {
179         sid: string;
180     }
181     message?: string
182 }
183
184 // 查看单个用户订阅
185 Method: GET
186 Path: /services/subscribe/:uid
187 {
188     Authorization: jwt
189     params:{
190     },
191 }
192
193 {
194     code: StatusCode
195     data?: {services:[{
196         building: string;
197         unit: string;
198         room: string;
199         name: string;
200         service_name: string;
201         phone: string;
202         line: string;
```

```
203     detail: string;
204     time: string;
205   }, {...}, {...}],allPages:number}
206   message?: string
207 }
208
209 // 查看全部订阅
210 Method: GET
211 Path: /services/subscribe
212 {
213   Authorization: jwt
214   params:{
215     offset: string;
216     pageNum: string;
217   },
218 }
219
220 {
221   code: StatusCode
222   data?:{services: [{
223     srid: string;
224     building: string;
225     unit: string;
226     room: string;
227     name: string;
228     service_name: string;
229     phone: string;
230     line: string;
231     detail: string;
232     time: string;
233   }, {...}, {...}],allPages:number}
234   message?: string
235 }
236
237 // 查看单个订阅
238 Method: GET
239 Path: /services/subscribe/:srid
240 {
241   Authorization: jwt
242   params:{
243   },
244 }
245
246 {
247   code: StatusCode
248   data?:{
249     srid: string;
```



```

250         avator: string;
251         cover: string;
252         building: string;
253         unit: string;
254         room: string;
255         name: string;
256         service_name: string;
257         phone: string;
258         line: string;
259         detail: string;
260         time: string;
261     }
262     message?: string
263 }
264
265

```

公告

```

1  /***** 公告 *****/
2  // 添加公告
3  Method: POST
4  Path: /notifies
5  {
6      Authorization: jwt
7      params:{
8          title: string;
9          time: string;
10         content: string;
11     },
12 }
13
14 {
15     code: StatusCode
16     data?: {
17         nid: string;
18     }
19     message?: string
20 }
21
22 // 删除公告
23 Method: DELETE
24 Path: /notifies/:nid
25 {
26     Authorization: jwt

```

```
27     params:{
28     },
29 }
30
31 {
32     code: StatusCode
33     data?: {
34         nid: string;
35     }
36     message?: string
37 }
38
39 // 修改公告
40 Method: PATCH
41 Path: /notifies/:nid
42 {
43     Authorization: jwt
44     params:{
45         title?: string;
46         time?: string;
47         content?: string;
48     },
49 }
50
51 {
52     code: StatusCode
53     data?: {
54         nid: string;
55         title: string;
56         time: string;
57         content: string;
58     }
59     message?: string
60 }
61
62 // 查看所有公告
63 Method: GET
64 Path: /notifies/all
65 {
66     Authorization: jwt
67     params:{
68         // 一页多少数据
69         offset: number;
70         // 当前所在页数
71         pageNum: number;
72     },
73 }
```

```

74
75 {
76     code: StatusCode
77     data?: {notifies:[{
78         nid: string;
79         title: string;
80         time: string;
81         // 节选, 大概几十字
82         content_slice: string;
83     }, {...}, {...}],allPages:number}
84     message?: string
85 }
86
87 // 查看单个公告
88 Method: GET
89 Path: /notifies/:nid
90 {
91     Authorization: jwt
92     params:{
93     },
94 }
95
96 {
97     code: StatusCode
98     data?: {
99         nid: string;
100         title: string;
101         time: string;
102         content: string;
103     }
104     message?: string
105 }
106

```

教育

```

1  /***** 教育 *****/
2  // 添加电子书
3  Method: POST
4  Path: /education/ebook
5  {
6      Authorization: jwt
7      params:{
8          name: string;
9          description: string;

```

```
10     detail: string;
11     content: string;
12     cover: string;
13     labels: LabelType[];
14 },
15 }
16
17 {
18     code: StatusCode
19     data?: {
20         bid: string;
21     }
22     message?: string
23 }
24
25 // 删除电子书
26 Method: DELETE
27 Path: /education/ebook/:bid
28 {
29     Authorization: jwt
30     params:{
31     },
32 }
33
34 {
35     code: StatusCode
36     data?: {
37         bid: string;
38     }
39     message?: string
40 }
41
42 // 修改电子书信息
43 Method: PATCH
44 Path: /education/ebook/:bid
45 {
46     Authorization: jwt
47     params:{
48         name?: string;
49         description?: string;
50         detail: string;
51         content?: string;
52         cover?: string;
53         labels?: LabelType[];
54     },
55 }
56
```

```
57 {
58     code: StatusCode
59     data?: {
60         bid: string;
61         name: string;
62         description: string;
63         detail: string;
64         content: string;
65         cover: string;
66         labels: LabelType[];
67     }
68     message?: string
69 }
70
71 // 按标签查看所有书
72 Method: GET
73 Path: /education/ebook/search/label
74 {
75     Authorization: jwt
76     params:{
77         labels: LabelType;
78         // 一页多少数据
79         offset: number;
80         // 当前所在页数
81         pageNum: number;
82     },
83 }
84
85 {
86     code: StatusCode
87     data?: {ebooks:[{
88         bid: string;
89         name: string;
90         description: string;
91         cover: string;
92         labels: LabelType[];
93     }, {...}, {...}],allPages:number}
94     message?: string
95 }
96
97 // 按书名查看所有书（模糊搜索）
98 Method: GET
99 Path: /education/ebook/search/name
100 {
101     Authorization: jwt
102     params:{
103         name: string;
```

```
104         // 一页多少数据
105         offset: number;
106         // 当前所在页数
107         pageNum: number;
108     },
109 }
110
111 {
112     code: StatusCode
113     data?: {ebooks:[{
114         bid: string;
115         name: string;
116         description: string;
117         cover: string;
118         label: LabelType[];
119     }, {...}, {...}],allPages:number}
120     message?: string
121 }
122
123 // 按随机所有书 (“推荐”或“换一批”) —(不是
124 Method: GET
125 Path: /education/ebook/search/recommand
126 {
127     Authorization: jwt
128     params:{
129         // 一页多少数据
130         offset: number;
131         // 当前所在页数
132         pageNum: number;
133     },
134 }
135
136 {
137     code: StatusCode
138     data?: {ebooks:[{
139         bid: string;
140         name: string;
141         description: string;
142         cover: string;
143         labels: LabelType[];
144     }, {...}, {...}],allPages:number}
145     message?: string
146 }
147
148 // 查看单个电子书
149 Method: GET
150 Path: /education/ebook/:bid
```

```

151 {
152     Authorization: jwt
153     params:{
154     },
155 }
156
157 {
158     code: StatusCode
159     data?: {
160         bid: string;
161         name: string;
162         description: string;
163         detail: string;
164         content: string;
165         cover: string;
166         labels: LabelType[];
167     }
168 }
169 message?: string
170 }
171
172
173
174

```

招聘

```

1  /***** 招聘 *****/
2  // 添加招聘
3  Method: POST
4  Path: /hires/create
5  {
6      Authorization: jwt
7      params:{
8          title: string;
9          content: string;
10         start_time: string;
11         end_time: string;
12     },
13 }
14
15 {
16     code: StatusCode
17     data?: {
18         nid: string;

```



```
19     }
20     message?: string
21 }
22
23 // 删除招聘
24 Method: DELETE
25 Path: /hires/:hid
26 {
27     Authorization: jwt
28     params:{
29     },
30 }
31
32 {
33     code: StatusCode
34     data?: {
35         hid: string;
36     }
37     message?: string
38 }
39
40 // 修改招聘 (只有物业用户能修改status)
41 Method: PATCH
42 Path: /hires/:hid
43 {
44     Authorization: jwt
45     params:{
46         title?: string;
47         content?: string;
48         start_time?: string;
49         end_time?: string;
50         status?: string;
51     },
52 }
53
54 {
55     code: StatusCode
56     data?: {
57         hid: string;
58         ename: string;
59         title: string;
60         content: string;
61         start_time: string;
62         end_time: string;
63         status: string;
64     }
65     message?: string
```

```
66 }
67
68
69 // 查看所有招聘
70 Method: GET
71 Path: /hires/all
72 {
73     Authorization: jwt
74     params:{
75         status?: string;
76         // 公司名称, 模糊匹配
77         ename?: string;
78         // 是否已开始已结束或正在招聘
79         // 0: 正在招聘
80         // 1: 未开始
81         // 2: 已结束
82         available?: string;
83         // 一页多少数据
84         offset: number;
85         // 当前所在页数
86         pageNum: number;
87     },
88 }
89
90 {
91     code: StatusCode
92     data?:{hires: [{
93         hid: string;
94         title: string;
95         ename: string;
96         start_time: string;
97         end_time: string;
98         // 节选
99         content_slice: string;
100         status: string;
101     }, {...}, {...}],allPages:number}
102     message?: string
103 }
104
105 // 查看特定公司的招聘
106 Method: GET
107 Path: /hires/all/:uid
108 {
109     Authorization: jwt
110     params:{
111         // 一页多少数据
112         offset: number;
```

```
113 // 当前所在页数
114 pageNum: number;
115 },
116 }
117
118 {
119   code: StatusCode
120   data?: {hires:[{
121     hid: string;
122     title: string;
123     ename: string;
124     start_time: string;
125     end_time: string;
126     content_slice: string;
127     status: string;
128   }, {...}, {...}],allPages:number}
129   message?: string
130 }
131 Method: GET
132 Path: /hires/:hid
133 {
134   Authorization: jwt
135   params:{
136   },
137 }
138
139 {
140   code: StatusCode
141   data?: {
142     hid: string;
143     title: string;
144     ename: string;
145     start_time: string;
146     end_time: string;
147     content: string;
148     status: string;
149   }
150   message?: string
151 }
152
153
154
155 // 创建简历
156 Method: POST
157 Path: /resume
158 {
159   Authorization: jwt
```

```
160     params:{
161         hid: number;
162         content: string;
163         position: string;
164     },
165 }
166
167 {
168     code: StatusCode
169     data?: {
170         resume_id: string;
171     }
172     message?: string
173 }
174
175 // 查看所有简历
176 Method: GET
177 Path: /resume/all
178 {
179     Authorization: jwt
180     params:{
181         // 一页多少数据
182         offset: number;
183         // 当前所在页数
184         pageNum: number;
185     },
186 }
187
188 {
189     code: StatusCode
190     data?:{resumes: [{
191         resume_id: string;
192         // 用户姓名
193         name: string;
194         position: string;
195         phone: string;
196         email: string;
197         content: string;
198     }, {...}, {...}],allPages:number}
199     message?: string
200 }
201
202 // 查看单个简历
203 Method: GET
204 Path: /resume/:resume_id
205 {
206     Authorization: jwt
```

```

207     params:{
208     },
209 }
210
211 {
212     code: StatusCode
213     data?: [{
214         resume_id: string;
215         // 用户姓名
216         name: string;
217         position: string;
218         phone: string;
219         email: string;
220         content: string;
221     }, {...}, {...}]
222     message?: string
223 }
224
225

```

资源

```

1  /***** 资源 *****/
2  // address不需要创建，后端运行脚本mock一组社区数据
3  // 查看所有address
4  Method: GET
5  Path: /addresses
6  {
7      Authorization: jwt
8      params:{
9          // 一页多少数据
10         offset: number;
11         // 当前所在页数
12         pageNum: number;
13     },
14 }
15
16 {
17     code: StatusCode
18     data?: {addresses:[{
19         aid: string;
20         building: string;
21         unit: string;
22         room: string;
23         user_name: string;

```

```
24     phone: string;
25   }, {...}, {...}],allPages:number}
26   message?: string
27 }
28
29 // 添加资源
30 Method: POST
31 Path: /addresses/resources/:aid
32 {
33   Authorization: jwt
34   params:{
35     type: ResourceType;
36     value: number;
37     month: string;
38     year: string;
39   },
40 }
41
42 {
43   code: StatusCode
44   data?: {
45     resource_id: string;
46   }
47   message?: string
48 }
49
50 // 删除资源
51 Method: DELETE
52 Path: /addresses/resources/:resource_id
53 {
54   Authorization: jwt
55   params:{
56   },
57 }
58
59 {
60   code: StatusCode
61   data?: {
62     resource_id: string;
63   }
64   message?: string
65 }
66
67
68 // 查看一个用户对应address的资源
69 Method: GET
70 Path: /addresses/resources/:uid
```

```
71 {
72     Authorization: jwt
73     params:{
74         // 一页多少数据
75         offset: number;
76         // 当前所在页数
77         pageNum: number;
78     },
79 }
80
81 {
82     code: StatusCode
83     data?: {resources:[{
84         resource_id: string;
85         type: ResourceType;
86         value: string;
87         month: string;
88         year: string;
89         // 是否超过阈值
90         comparison: boolean;
91     }, {...}, {...}],
92     allPages:number
93     }
94     message?: string
95 }
96
97 // 创建阈值
98 Method: POST
99 Path: /addresses/alert/:uid
100 {
101     Authorization: jwt
102     params:{
103         type: ResourceType;
104         value: number;
105     },
106 }
107
108 {
109     code: StatusCode
110     data?: {
111         alert_id: string;
112     }
113     message?: string
114 }
115
116 // 删除阈值
117 Method: DELETE
```



```
118 Path: /addresses/alert/:alert_id
119 {
120     Authorization: jwt
121     params:{
122     },
123 }
124
125 {
126     code: StatusCode
127     data?: {
128         alert_id: string;
129     }
130     message?: string
131 }
132
133 // 修改阈值
134 Method: PATCH
135 Path: /addresses/alert/:alert_id
136 {
137     Authorization: jwt
138     params:{
139         type: ResourceType;
140         value: number;
141     },
142 }
143
144 {
145     code: StatusCode
146     data?: {
147         alert_id: string;
148     }
149     message?: string
150 }
151
152 // 查看阈值
153 Method: GET
154 Path: /addresses/alert/:uid
155 {
156     Authorization: jwt
157     params:{
158     },
159 }
160
161 {
162     code: StatusCode
163     data?: [{
164         alert_id: string;
```

```
165         type: ResourceType;
166         value: number;
167     }, {}, {}]
168     message?: string
169 }
170
171
```

人工智障接口

```
1 // 目前考虑会用到GET,POST,DELETE,PATCH
2 // 改资源某一字段用PATCH
3 Method: POST
4 // 参数可能不存在, 这里以get请求单个资源
5 // 比如获取一个请求地址里关于房屋信息的资源,所有资源名称是复数
6 Path: //:aid
7 // 发送请求传参json
8 {
9     Authorization: ewoifnweiofwifjwi
10    params:{
11        // 实际api的字段
12        // 例如PATCH用户性别
13        sex: 0, 1, man, weman,,,,
14    },
15 }
16
17 // 返回参数json
18 {
19     code: statusCode
20     // 后端处理后的资源
21     data?: {
22         aid:
23         unit:
24         building:
25         host_nameame:
26     }
27     message?: "dadasdadadada"
28 }
```

```
1 // 获取已有的消息,可做可不做, 如果消息不存数据库就不做
2 Method: GET
3 Path: /chat
4
5 {
6     Authorization: jwt
7     params:{
8         uid: int
9     },
10 }
11
12 // 返回参数json
13 {
14     code: statusCode
15     // 按先后顺序排布
16     data?: [
17         {
18             // 消息谁发的 0是user 1是chat
19             from:0 | 1
20             content: string
21         }
22     ]
23
24     message?: "dadasdadadada"
25 }
26
27
28
29
30 // 发布消息, 并接受到回复
31 Method: POST
32 Path: /chat
33
34 {
35     Authorization: jwt
36     params:{
37         content: string
38     },
39 }
40
41 // 返回参数json
42 {
43     code: statusCode
44     // 按先后顺序排布
45     data?: {
46         content: string
47     }
```

```
48
49
50     message?: "dadasdadadada"
51 }
```

新加的接口

```
1 // 查看是否已经审核用户（分页）
2 // filter: status是否为pending
3 //    0: 已经审核
4 //    1: status为pending
5 Method: GET
6 Path: /users/review
7 {
8     Authorization: jwt
9     params:{
10         filter: string;
11         // 一页多少数据
12         offset: number;
13         // 当前所在页数
14         pageNum: number;
15     },
16 }
17
18 {
19     code: StatusCode
20     data?: {
21         users:[{
22             uid: string;
23             name: string;
24             status: string;
25             building: string;
26             unit: string;
27             room: string;
28         }, {...}, {...}],
29         // 一共多少页
30         allPages: number
31     }
32     message?: string
33 }
34
35 // 查看是否已经审核招聘（分页）
36 // filter: status是否为pending
37 //    0: 已经审核
38 //    1: status为pending
```

```
39 Method: GET
40 Path: /hires/review
41 {
42     Authorization: jwt
43     params:{
44         filter: string;
45         offset: number;
46         // 当前所在页数
47         pageNum: number;
48     },
49 }
50
51 {
52     code: StatusCode
53     data?:{
54         hires: [{
55             hid: string;
56             title: string;
57             ename: string;
58             status: string;
59         }, {...}, {...}],
60         allPages:number
61     }
62     message?: string
63 }
```