

1. POST Register
 - 1.1. 请求参数
 - 1.2. 返回结果
 - 1.3. 返回数据结构
2. POST Login
 - 2.1. 请求参数
 - 2.2. 返回结果
 - 2.3. 返回数据结构
3. GET UpdateSend
 - 3.1. 请求参数
 - 3.2. 返回结果
 - 3.3. 返回数据结构
4. GET SendMessage
 - 4.1. 请求参数
 - 4.2. 返回结果
 - 4.3. 返回数据结构
5. GET GetUserInfo
 - 5.1. 请求参数
 - 5.2. 返回结果
 - 5.3. 返回数据结构
6. POST Opinion
 - 6.1. 请求参数
 - 6.2. 返回结果
 - 6.3. 返回数据结构
7. POST UpdatePassword
 - 7.1. 请求参数
 - 7.2. 返回结果
 - 7.3. 返回数据结构
8. GET GetWeather
 - 8.1. 请求参数
 - 8.2. 返回结果
 - 8.3. 返回数据结构
9. POST SetUserInfo
 - 9.1. 请求参数
 - 9.2. 返回结果
 - 9.3. 返回数据结构
10. GET GetAlertInfo
 - 10.1. 请求参数
 - 10.2. 返回结果
 - 10.3. 返回数据结构
11. GET GetTemperature
 - 11.1. 请求参数
 - 11.2. 返回结果
 - 11.3. 返回数据结构
12. GET GetLightIntensity
 - 12.1. 请求参数
 - 12.2. 返回结果
 - 12.3. 返回数据结构
13. GET GetHumidity
 - 13.1. 请求参数

- 13.2. [返回结果](#)
 - 13.3. [返回数据结构](#)
- 14. [GET GetlightIntensityAll](#)
 - 14.1. [请求参数](#)
 - 14.2. [返回结果](#)
 - 14.3. [返回数据结构](#)
- 15. [GET GethumidityAll](#)
 - 15.1. [请求参数](#)
 - 15.2. [返回结果](#)
 - 15.3. [返回数据结构](#)
- 16. [GET Temperature-all](#)
 - 16.1. [请求参数](#)
 - 16.2. [返回结果](#)
 - 16.3. [返回数据结构](#)
- 17. [GET GetHumidityMath](#)
 - 17.1. [请求参数](#)
 - 17.2. [返回结果](#)
 - 17.3. [返回数据结构](#)
- 18. [GET GetMathTemperature](#)
 - 18.1. [请求参数](#)
 - 18.2. [返回结果](#)
 - 18.3. [返回数据结构](#)
- 19. [GET GetLightIntensityMath](#)
 - 19.1. [请求参数](#)
 - 19.2. [返回结果](#)
 - 19.3. [返回数据结构](#)
- 20. [GET GetTemperatureWeek](#)
 - 20.1. [请求参数](#)
 - 20.2. [返回结果](#)
 - 20.3. [返回数据结构](#)
- 21. [GET GetLightIntensityWeek](#)
 - 21.1. [请求参数](#)
 - 21.2. [返回结果](#)
 - 21.3. [返回数据结构](#)
- 22. [GET GetHumidityWeek](#)
 - 22.1. [请求参数](#)
 - 22.2. [返回结果](#)
 - 22.3. [返回数据结构](#)
- 23. [GET GetDeviceState](#)
 - 23.1. [请求参数](#)
 - 23.2. [返回结果](#)
 - 23.3. [返回数据结构](#)
- 24. [GET GetLightStatus](#)
 - 24.1. [请求参数](#)
 - 24.2. [返回结果](#)
 - 24.3. [返回数据结构](#)
- 25. [GET GetFanStatus](#)
 - 25.1. [请求参数](#)
 - 25.2. [返回结果](#)
 - 25.3. [返回数据结构](#)
- 26. [GET GetPumpStatus](#)
 - 26.1. [请求参数](#)
 - 26.2. [返回结果](#)
 - 26.3. [返回数据结构](#)
- 27. [GET Lightctl](#)
 - 27.1. [请求参数](#)

- 27.2. 返回结果
 - 27.3. 返回数据结构
- 28. GET Pumpctl
 - 28.1. 请求参数
 - 28.2. 返回结果
 - 28.3. 返回数据结构
- 29. GET Windctl
 - 29.1. 请求参数
 - 29.2. 返回结果
 - 29.3. 返回数据结构
- 30. GET GetLightAuto
 - 30.1. 请求参数
 - 30.2. 返回结果
 - 30.3. 返回数据结构
- 31. GET GetPumpAuto
 - 31.1. 请求参数
 - 31.2. 返回结果
 - 31.3. 返回数据结构
- 32. GET GetFenAuto
 - 32.1. 请求参数
 - 32.2. 返回结果
 - 32.3. 返回数据结构
- 33. registerInfo
 - 33.1. 属性
- 34. UserInfo
 - 34.1. 属性
- 35. User
 - 35.1. 属性

loginandregister

1. POST Register

POST /register

用户通过使用手机号进行验证注册，并且设置密码

Body 请求参数

```
1 {
2   "phoneNumber": "19184646504",
3   "password": "xyz096013",
4   "code": "3597"
5 }
```

1.1. 请求参数

名称	位置	类型	必选	说明
body	body	object	否	none
» phoneNumber	body	string	是	用户手机号
» password	body	string	是	用户设置密码
» code	body	string	是	注册时获取的验证码

返回示例

成功

```
1 {
2   "code": 200,
3   "data": "注册成功",
4   "msg": "ok"
5 }
```

1.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

1.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码

名称	类型	必选	约束	中文名	说明
» data	string	true	none		响应数据
» msg	string	true	none		响应信息

2. POST Login

POST /login

登陆验证的接口，通过对应登陆信息的验证，生成token为用户赋予登陆的状态。

Body 请求参数

```
1 {
2   "phoneNumber": "19184646504",
3   "password": "123456"
4 }
```

2.1. 请求参数

名称	位置	类型	必选	说明
body	body	object	否	none
» phoneNumber	body	string	是	用户的手机号
» password	body	string	是	用户设置的密码

返回示例

成功

```
1 {
2   "code": 200,
3   "data": {
4     "token":
5     "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6MywiZXhwIjoxNzEyNDk1MzkzLCJpYXQ
6     iOjE3MTI0ODgxOTN5.ma5moB8uxUrnVH3E2DXCWV6vFJkrM6a1ZGBnSGSWTCw",
7     "user": {
8       "id": 3,
9       "phone": "1111111111"
10    }
11  },
12  "msg": "登录成功！"
```

```
1 {
2   "code": 400,
3   "data": null,
4   "msg": "密码错误！"
5 }
```

2.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

2.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		请求成功的响应码
» data	object	true	none		返回的数据
»» token	string	true	none		由用户手机号与密码生成的唯一token，进行登陆状态的验证
»» user	object	true	none		none
»»» id	integer	true	none		用户的唯一id
»»»» phoneNumber	string	true	none		用户的手机号
» msg	string	true	none		none

smsSendMessage

1. GET UpdateSend

GET /api/msm/updateSend

获取修改的密码的验证码

1.1. 请求参数

名称	位置	类型	必选	说明
phoneNumber	query	string	否	none
Authorization	header	string	是	用户验证生成的token进行后端身份验证，由登陆时生成，前端进行保存携带验证

返回示例

成功

```
1 {
2   "code": 200,
3   "data": {
4     "code": "1267",
5     "phoneNumber": "19184646504"
6   },
7   "msg": "验证码发送成功！"
8 }
```

1.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

1.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	string	true	none		响应码
» data	object	true	none		none
»» phoneNumber	string	true	none		用户手机号
»» code	string	true	none		验证码
» msg	string	true	none		响应信息

2. GET SendMessage

GET /api/msm/RegisterSend

用户注册时，通过手机号获取验证码进行验证

2.1. 请求参数

名称	位置	类型	必选	说明
phoneNumber	query	string	是	接收验证码的手机号

返回示例

成功

```
1 {
2   "code": 200,
3   "data": {
4     "code": "4729",
5     "phoneNumber": "19184646504"
6   },
7   "msg": "验证码发送成功！"
8 }
```

2.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

2.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	false	none		响应状态码
» data	object	true	none		none
»» phoneNumber	string	true	none		用户手机号
»» code	string	true	none		验证码返回
» msg	string	true	none		响应信息

UserManagementController

1. GET GetUserInfo

GET /secure/getUserInfo

获取用户的信息，用于信息页的展示。

1.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	否	none

返回示例

成功

```
1 {
2   "code": 200,
3   "data": {
4     "id": 4,
5     "nickname": "崔强",
6     "email": "w.phfo@qq.com",
7     "address": "福建省和田地区宝坻区"
8   },
9   "msg": "查询成功"
10 }
```

1.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

1.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	object	true	none		none
»» id	integer	true	none		ID 编号
»» nickname	string	false	none		昵称
»» email	string	false	none		电子邮箱
»» address	string	false	none		住址

名称	类型	必选	约束	中文名	说明
» msg	string	true	none		响应信息

2. POST Opinion

POST /secure/opinion

用户发送用户反馈。

2.1. 请求参数

名称	位置	类型	必选	说明
phone_number	query	string	否	用户号码
description	query	string	否	反馈信息内容
Authorization	header	string	否	none

返回示例

成功

```
1 {
2   "code": 200,
3   "data": {
4     "phone_number": "12345678991",
5     "description": "测试信息"
6   },
7   "msg": "意见存储成功！"
8 }
```

2.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

2.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	object	true	none		none
»» phone_number	string	true	none		用户联系方式
»» description	string	true	none		反馈了内容
» msg	string	true	none		响应信息

3. POST UpdatePassword

POST /secure/UpdatePassword

通过手机验证后，用户对密码进行修改

3.1. 请求参数

名称	位置	类型	必选	说明
code	query	string	否	手机验证码
phoneNumber	query	string	否	手机号
NewPassword	query	string	否	新密码
Authorization	header	string	是	none

返回示例

成功

```
1 {
2   "code": 500,
3   "data": null,
4   "msg": "验证码错误"
5 }
```

```
1 {
2   "code": 200,
3   "data": "19184646504",
4   "msg": "密码更新成功"
5 }
```

3.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

3.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	string	true	none	码	响应码
» data	string	true	none		none
» msg	string	true	none		响应信息

4. GET GetWeather

GET /secure/weather

获取实时天气，code表示为地区代码

4.1. 请求参数

名称	位置	类型	必选	说明
cityCode	query	string	否	为气象局发布的地区代码
Authorization	header	string	否	none

返回示例

成功

```
1 {
2   "code": 200,
3   "data": [
4     {
5       "city": "渝北区",
6       "time": "2024-04-10 16:57:54",
7       "shidu": "82%",
8       "quality": "优",
9       "wendu": "19",
10      "high": "高温 21℃",
11      "low": "低温 16℃",
12      "sunrise": "06:31",
13      "sunset": "19:17",
14      "aqi": "39",
15      "fx": "东北风",
16      "fl": "1级",
17      "type": "多云",
18      "notice": "阴晴之间，谨防紫外线侵扰"
19    }
20  ],
21  "msg": "查询成功"
22 }
```

4.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

4.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		none
» data	[object]	true	none		none
»» city	string	false	none		城市名
»» time	string	false	none		时间
»» shidu	string	false	none		湿度
»» quality	string	false	none		空气质量
»» wendu	string	false	none		温度
»» high	string	false	none		最高气温
»» low	string	false	none		最低气温
»» sunrise	string	false	none		日出时间
»» sunset	string	false	none		日落时间
»» aqi	string	false	none		空气指数
»» fx	string	false	none		风向
»» fl	string	false	none		风力
»» type	string	false	none		天气类型
»» notice	string	false	none		温馨提示
» msg	string	true	none		none

5. POST SetUserInfo

POST /secure/SetUserInfo

以json形式传入对应的userInfo的属性来进行用户信息的修改。

Body 请求参数

```
1 {
2   "nickname": "xyz",
3   "email": "w.phfo@qq.com",
4   "address": "重庆市渝北区"
5 }
```

5.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	否	用户验证生成的token进行后端身份验证，由登陆时生成，前端进行保存携带验证

名称	位置	类型	必选	说明
body	body	object	否	none
» nickname	body	string	否	昵称
» email	body	string	否	电子邮箱
» address	body	string	否	地址

返回示例

成功

```
1 {
2   "code": 200,
3   "data": {
4     "id": 4,
5     "nickname": "user2",
6     "email": "v.swrn@qq.com",
7     "address": "海南省宿迁市东源县"
8   },
9   "msg": "用户个人信息保存成功"
10 }
```

```
1 {
2   "code": 400,
3   "data": {
4     "id": 3,
5     "nickname": "user2",
6     "email": "v.swrn@qq.com",
7     "address": "海南省宿迁市东源县"
8   },
9   "msg": "用户 id 不存在"
10 }
```

```
1 {
2   "code": 200,
3   "data": {
4     "id": 4,
5     "nickname": "崔强",
6     "email": "w.phfo@qq.com",
7     "address": "福建省和田地区宝坻区"
8   },
9   "msg": "用户个人信息保存成功"
10 }
```

5.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

5.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	object	true	none		none
»» id	integer	true	none		用户id
»» nickname	string	true	none		昵称
»» email	string	true	none		电子邮箱
»» address	string	true	none		住址
» msg	string	true	none		响应信息

MonitorApi

1. GET GetAlertInfo

GET /secure/alert-info

//alertType 预警类型 1:光照, 2: 土壤湿度, 3: 温度

1.1. 请求参数

名称	位置	类型	必选	说明
alertType	query	integer	是	预警信息的类型
Authorization	header	string	否	none

返回示例

成功

```
1 {
2   "code": 200,
3   "data": {
4     "id": 10,
5     "alertType": 2,
6     "alertData": "835.93",
7     "alertMessage": "当前土壤湿度异常",
8     "alertTime": "2024-04-15T02:57:52.000+0000"
9   },
10  "msg": "获取成功"
11 }
```

1.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

1.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	object	true	none		none
»» id	integer	true	none		数据id
»» alertType	integer	true	none		预警类型
»» alertData	string	true	none		预警数据

名称	类型	必选	约束	中文名	说明
»» alertMessage	string	true	none		数据信息
»» alertTime	string	true	none		数据产生时间
» msg	string	true	none		响应信息

MonitorApi/实时

1. GET GetTemperature

GET /secure/temperature

获取实时的温度

1.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	是	none

返回示例

成功

```
1 {
2   "code": 200,
3   "data": {
4     "temperature": 23
5   },
6   "msg": "获取成功"
7 }
```

1.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

1.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	object	true	none		none
»» temperature	integer	true	none		实时的温度
» msg	string	true	none		响应信息

2. GET GetLightIntensity

GET /secure/light-intensity

获取实时的光照强度信息

2.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	是	none

返回示例

成功

```
1 {
2   "code": 200,
3   "data": {
4     "lightIntensity": 155
5   },
6   "msg": "获取成功"
7 }
```

2.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

2.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	object	true	none		none
»» lightIntensity	integer	true	none		光照强度
» msg	string	true	none		响应信息

3. GET GetHumidity

GET /secure/humidity

获取实时的湿度

3.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	是	用户验证生成的token进行后端身份验证，由登陆时生成，前端进行保存携带验证

返回示例

```
1 {
2   "code": 200,
3   "data": {
4     "humidity": 64
5   },
6   "msg": "获取成功"
7 }
```

3.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

3.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	object	true	none		none
»» humidity	integer	true	none		实时的湿度信息
» msg	string	true	none		响应信息

MonitorApi/获取历史信息

1. GET GetlightIntensityAll

GET /secure/light-intensity-all

获取历史光照信息

1.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	否	none

返回示例

成功

```
1  {
2    "code": 200,
3    "data": [
4      {
5        "id": 11,
6        "lightIntensity": 155,
7        "readingTime": "2024-04-10T08:02:47.000+0000"
8      },
9      {
10       "id": 10,
11       "lightIntensity": 177,
12       "readingTime": "2024-04-10T08:02:43.000+0000"
13     },
14     {
15       "id": 9,
16       "lightIntensity": 178,
17       "readingTime": "2024-04-10T08:02:39.000+0000"
18     },
19     {
20       "id": 8,
21       "lightIntensity": 169,
22       "readingTime": "2024-04-10T08:02:37.000+0000"
23     },
24     {
25       "id": 7,
26       "lightIntensity": 186,
27       "readingTime": "2024-04-10T08:02:35.000+0000"
28     },
29     {
30       "id": 6,
31       "lightIntensity": 167,
32       "readingTime": "2024-04-10T08:02:32.000+0000"
33     },
34     {
35       "id": 5,
36       "lightIntensity": 164,
```

```
37     "readingTime": "2024-04-10T08:02:30.000+0000"
38   },
39   {
40     "id": 4,
41     "lightIntensity": 121,
42     "readingTime": "2024-04-10T08:02:27.000+0000"
43   },
44   {
45     "id": 3,
46     "lightIntensity": 111,
47     "readingTime": "2024-04-10T08:02:24.000+0000"
48   },
49   {
50     "id": 2,
51     "lightIntensity": 110,
52     "readingTime": "2024-04-10T08:02:17.000+0000"
53   },
54   {
55     "id": 1,
56     "lightIntensity": 116,
57     "readingTime": "2024-04-10T06:46:28.000+0000"
58   }
59 ],
60 "msg": "获取成功"
61 }
```

1.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

1.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	[object]	true	none		none
»» id	integer	true	none		数据id
»» lightIntensity	integer	true	none		光照强度
»» readingTime	string	true	none		数据产生时间
» msg	string	true	none		响应信息

2. GET GethumidityAll

GET /secure/humidity-all

获取历史的湿度信息

2.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	是	用户验证生成的token进行后端身份验证，由登陆时生成，前端进行保存携带验证

返回示例

成功

```
1  {
2    "code": 200,
3    "data": [
4      {
5        "id": 1,
6        "soilHumidity": 102.25,
7        "readingTime": "2024-04-09T03:22:25.000+0000"
8      },
9      {
10       "id": 2,
11       "soilHumidity": 60,
12       "readingTime": "2024-04-09T08:42:08.000+0000"
13     },
14     {
15       "id": 3,
16       "soilHumidity": 60,
17       "readingTime": "2024-04-10T06:44:50.000+0000"
18     },
19     {
20       "id": 4,
21       "soilHumidity": 60,
22       "readingTime": "2024-04-10T07:59:43.000+0000"
23     },
24     {
25       "id": 5,
26       "soilHumidity": 65,
27       "readingTime": "2024-04-10T08:00:42.000+0000"
28     },
29     {
30       "id": 6,
31       "soilHumidity": 68,
32       "readingTime": "2024-04-10T08:00:54.000+0000"
33     },
34     {
35       "id": 7,
36       "soilHumidity": 64,
37       "readingTime": "2024-04-10T08:01:03.000+0000"
38     },
39     {
40       "id": 8,
41       "soilHumidity": 63,
42       "readingTime": "2024-04-10T08:01:08.000+0000"
43     }
44   ]
45 }
```

```
43     },
44     {
45         "id": 9,
46         "soilHumidity": 61,
47         "readingTime": "2024-04-10T08:01:14.000+0000"
48     },
49     {
50         "id": 10,
51         "soilHumidity": 61,
52         "readingTime": "2024-04-10T08:01:22.000+0000"
53     },
54     {
55         "id": 11,
56         "soilHumidity": 61,
57         "readingTime": "2024-04-10T08:01:27.000+0000"
58     },
59     {
60         "id": 12,
61         "soilHumidity": 61,
62         "readingTime": "2024-04-10T08:01:30.000+0000"
63     },
64     {
65         "id": 13,
66         "soilHumidity": 64,
67         "readingTime": "2024-04-10T08:01:33.000+0000"
68     },
69     {
70         "id": 14,
71         "soilHumidity": 64,
72         "readingTime": "2024-04-10T08:01:36.000+0000"
73     }
74 ],
75 "msg": "获取成功"
76 }
```

2.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

2.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	[object]	true	none		none
»» id	integer	true	none		数据id
»» soilHumidity	number	true	none		湿度

名称	类型	必选	约束	中文名	说明
»» readingTime	string	true	none		数据产生的时间
» msg	string	true	none		响应信息

3. GET Temperature-all

GET /secure/temperature-all

获取历史的温度信息

3.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	是	用户验证生成的token进行后端身份验证，由登陆时生成，前端进行保存携带验证

返回示例

成功

```
1  {
2    "code": 200,
3    "data": [
4      {
5        "id": 12,
6        "airTemperature": 23,
7        "readingTime": "2024-04-10T08:01:36.000+0000"
8      },
9      {
10       "id": 11,
11       "airTemperature": 22,
12       "readingTime": "2024-04-10T08:01:33.000+0000"
13     },
14     {
15       "id": 10,
16       "airTemperature": 22,
17       "readingTime": "2024-04-10T08:01:30.000+0000"
18     },
19     {
20       "id": 9,
21       "airTemperature": 23,
22       "readingTime": "2024-04-10T08:01:28.000+0000"
23     },
24     {
25       "id": 8,
26       "airTemperature": 23,
27       "readingTime": "2024-04-10T08:01:22.000+0000"
28     },
29     {
30       "id": 7,
```

```
31     "airTemperature": 20,
32     "readingTime": "2024-04-10T08:01:14.000+0000"
33 },
34 {
35     "id": 6,
36     "airTemperature": 19,
37     "readingTime": "2024-04-10T08:01:08.000+0000"
38 },
39 {
40     "id": 5,
41     "airTemperature": 23,
42     "readingTime": "2024-04-10T08:01:03.000+0000"
43 },
44 {
45     "id": 4,
46     "airTemperature": 22,
47     "readingTime": "2024-04-10T08:00:54.000+0000"
48 },
49 {
50     "id": 3,
51     "airTemperature": 22,
52     "readingTime": "2024-04-10T08:00:42.000+0000"
53 },
54 {
55     "id": 2,
56     "airTemperature": 20,
57     "readingTime": "2024-04-10T07:59:43.000+0000"
58 },
59 {
60     "id": 1,
61     "airTemperature": 20,
62     "readingTime": "2024-04-10T06:44:50.000+0000"
63 }
64 ],
65 "msg": "获取成功"
66 }
```

3.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

3.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	[object]	true	none		none
»» id	integer	true	none		数据id

名称	类型	必选	约束	中文名	说明
»» airTemperature	integer	true	none		温度
»» readingTime	string	true	none		数据产生的时间
» msg	string	true	none		响应信息

MonitorApi/获取科学计算后的数据

1. GET GetHumidityMath

GET /secure/humidity-math

获取最近两百条湿度数据的最大值，最小值，平均值。

1.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	是	none

返回示例

成功

```
1 {
2   "code": 200,
3   "data": {
4     "max": 102.25,
5     "min": 60,
6     "avg": 65.30357
7   },
8   "msg": "获取成功"
9 }
```

1.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

1.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	object	true	none		none
»» max	number	true	none		最大值
»» min	integer	true	none		最小值
»» avg	number	true	none		平均值
» msg	string	true	none		响应信息

2. GET GetMathTemperature

GET /secure/temperature-math

给出平均值，最大值，最小值。

2.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	是	用户验证生成的token进行后端身份验证，由登陆时生成，前端进行保存携带验证

返回示例

成功

```
1 {
2   "code": 200,
3   "data": {
4     "max": 23,
5     "min": 19,
6     "avg": 21.583334
7   },
8   "msg": "获取成功"
9 }
```

2.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

2.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	object	true	none		none
»» max	integer	true	none		最大值
»» min	integer	true	none		最小值
»» avg	number	true	none		平均值
» msg	string	true	none		响应信息

3. GET GetLightIntensityMath

GET /secure/light-intensity-math

获取最近两百条光照数据的最大值，最小值，平均值。

3.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	是	用户验证生成的token进行后端身份验证，由登陆时生成，前端进行保存携带验证

返回示例

成功

```
1 {
2   "code": 200,
3   "data": {
4     "max": 186,
5     "min": 110,
6     "avg": 150.36363
7   },
8   "msg": "获取成功"
9 }
```

3.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

3.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	object	true	none		none
»» max	integer	true	none		最大值
»» min	integer	true	none		最小值
»» avg	number	true	none		平均值
» msg	string	true	none		响应信息

MonitorApi/一周数据

1. GET GetTemperatureWeek

GET /secure/temperature-week

获取最近一周的数据，每天都取平均值

1.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	是	用户验证生成的token进行后端身份验证，由登陆时生成，前端进行保存携带验证

返回示例

成功

```
1 {
2   "code": 200,
3   "data": [
4     {
5       "airTemperature": 22,
6       "readingTime": "2024-04-11T16:00:00.000+0000"
7     },
8     {
9       "airTemperature": 9.5,
10      "readingTime": "2024-04-10T16:00:00.000+0000"
11    },
12    {
13      "airTemperature": 20.934546,
14      "readingTime": "2024-04-09T16:00:00.000+0000"
15    },
16    {
17      "airTemperature": 19.413,
18      "readingTime": "2024-04-08T16:00:00.000+0000"
19    },
20    {
21      "airTemperature": 20.503,
22      "readingTime": "2024-04-07T16:00:00.000+0000"
23    }
24  ],
25  "msg": "获取成功"
26 }
```

1.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

1.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	[object]	true	none		none
»» airTemperature	integer	true	none		温度
»» readingTime	string	true	none		数据日期
» msg	string	true	none		响应信息

2. GET GetLightIntensityWeek

GET /secure/light-intensity-week

获取最近一周的数据，每天都取平均值

2.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	是	用户验证生成的token进行后端身份验证，由登陆时生成，前端进行保存携带验证

返回示例

成功

```
1 {
2   "code": 200,
3   "data": [
4     {
5       "lightIntensity": 168,
6       "readingTime": "2024-04-11T16:00:00.000+0000"
7     },
8     {
9       "lightIntensity": 178,
10      "readingTime": "2024-04-10T16:00:00.000+0000"
11    },
12    {
13      "lightIntensity": 150.36363,
14      "readingTime": "2024-04-09T16:00:00.000+0000"
15    },
16    {
17      "lightIntensity": 136.103,
18      "readingTime": "2024-04-08T16:00:00.000+0000"
19    },
20    {
21      "lightIntensity": 147.7806,
```



```
22     "readingTime": "2024-04-07T16:00:00.000+0000"
23   }
24 ],
25   "msg": "获取成功"
26 }
```

2.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

2.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	[object]	true	none		none
»» lightIntensity	integer	true	none		光照
»» readingTime	string	true	none		日期
» msg	string	true	none		响应信息

3. GET GetHumidityWeek

GET /secure/humidity-week

获取最近一周的数据，每天都取平均值

3.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	是	用户验证生成的token进行后端身份验证，由登陆时生成，前端进行保存携带验证

返回示例

成功

```
1  {
2    "code": 200,
3    "data": [
4      {
5        "soilHumidity": 81.5,
6        "readingTime": "2024-04-10T16:00:00.000+0000"
7      },
8    ]
9  }
```

```
9      "soilHumidity": 60.44182,
10     "readingTime": "2024-04-09T16:00:00.000+0000"
11   },
12   {
13     "soilHumidity": 53.78909,
14     "readingTime": "2024-04-08T16:00:00.000+0000"
15   },
16   {
17     "soilHumidity": 44.215,
18     "readingTime": "2024-04-07T16:00:00.000+0000"
19   }
20 ],
21 "msg": "获取成功"
22 }
```

3.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

3.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	[object]	true	none		none
»» soilHumidity	number	true	none		湿度
»» readingTime	string	true	none		日期
» msg	string	true	none		响应信息

MqttController/获取状态

1. GET GetDeviceState

GET /secure/get_device_state

获取各类设备的状态数据（风扇，水泵，灯光）

1.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	是	用户验证生成的token进行后端身份验证，由登陆时生成，前端进行保存携带验证

返回示例

成功

```
1 {
2   "code": 200,
3   "data": {
4     "pumpCtrlState": 1,
5     "pumpPowerState": 1,
6     "fanMode": 1,
7     "fanLevel": 1,
8     "lightMode": 1,
9     "light_level": 0
10  },
11  "msg": "获取成功"
12 }
```

1.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

1.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	object	true	none		none
»» pumpCtrlState	integer	true	none		水泵的控制状态
»» pumpPowerState	integer	true	none		水泵的电源状态

名称	类型	必选	约束	中文名	说明
» fanMode	integer	true	none		风扇的控制状态
» fanLevel	integer	true	none		风扇的挡位
» lightMode	integer	true	none		灯光的控制状态
» light_level	integer	true	none		灯光的挡位
» msg	string	true	none		响应信息

2. GET GetLightStatus

GET /secure/getLightStatus

获取灯光设备的状态

2.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	是	用户验证生成的token进行后端身份验证，由登陆时生成，前端进行保存携带验证

返回示例

成功

```
1 {
2   "code": 200,
3   "data": {
4     "lightMode": 1,
5     "light_level": 0
6   },
7   "msg": "查询成功"
8 }
```

2.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

2.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码

名称	类型	必选	约束	中文名	说明
» data	object	true	none		none
»» lightMode	integer	true	none		灯光状态
»» light_level	integer	true	none		灯光挡位，在lightMode为1时才生效
» msg	string	true	none		响应信息

3. GET GetFanStatus

GET /secure/getFanStatus

获取风扇状态信息。

3.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	是	用户验证生成的token进行后端身份验证，由登陆时生成，前端进行保存携带验证

返回示例

成功

```
1 {
2   "code": 200,
3   "data": {
4     "fanMode": 1,
5     "fanLevel": 51
6   },
7   "msg": "查询成功"
8 }
```

3.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

3.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	object	true	none		none

名称	类型	必选	约束	中文名	说明
»» fanMode	integer	true	none		风扇控制状态
»» fanLevel	integer	true	none		风扇挡位，当fanMode为1时才会生效
» msg	string	true	none		响应信息

4. GET GetPumpStatus

GET /secure/getPumpStatus

获取水泵的状态

4.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	是	用户验证生成的token进行后端身份验证，由登陆时生成，前端进行保存携带验证

返回示例

成功

```
1 {
2   "code": 200,
3   "data": {
4     "pumpCtrlState": 0,
5     "pumpPowerState": 0
6   },
7   "msg": "查询成功"
8 }
```

4.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

4.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	object	true	none		none

名称	类型	必选	约束	中文名	说明
»» pumpCtrlState	integer	true	none		水泵的控制状态，当 pumpPowerState为1时生效
»» pumpPowerState	integer	true	none		水泵的电源
» msg	string	true	none		响应信息

MqttController/控制

1. GET Lightctl

GET /secure/lightctl

灯光设备的状态控制

1.1. 请求参数

名称	位置	类型	必选	说明
light_level	query	integer	否	灯光挡位，当light_mode为1时生效。
light_mode	query	integer	否	灯光状态
Authorization	header	string	否	none

返回示例

成功

```
1 {
2   "code": 200,
3   "data": {
4     "light_mode": 1,
5     "light_level": 1
6   },
7   "msg": "指令已发送"
8 }
```

1.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

1.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	object	true	none		none
»» light_mode	integer	true	none		灯光挡位，当light_mode为1时生效。
»» light_level	integer	true	none		灯光状态

名称	类型	必选	约束	中文名	说明
» msg	string	true	none		响应信息

2. GET Pumpctl

GET /secure/Pumpctl

对水泵设备进行配置

2.1. 请求参数

名称	位置	类型	必选	说明
p_c_state	query	string	是	水泵控制状态
Authorization	header	string	是	用户验证生成的token进行后端身份验证，由登陆时生成，前端进行保存携带验证

返回示例

成功

```
1 {
2   "code": 200,
3   "data": {
4     "pump_ctrl_state": 1
5   },
6   "msg": "指令已发送"
7 }
```

2.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

2.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	object	true	none		none
»» pump_ctrl_state	integer	true	none		水泵控制状态
» msg	string	true	none		响应信息

3. GET Windctl

GET /secure/Windctl

对风扇设备进行控制

3.1. 请求参数

名称	位置	类型	必选	说明
f_mode	query	integer	是	风扇的状态
f_level	query	string	是	风扇的挡位
Authorization	header	string	是	用户验证生成的token进行后端身份验证，由登陆时生成，前端进行保存携带验证

返回示例

成功

```
1 {
2   "code": 200,
3   "data": {
4     "fan_mode": 1,
5     "fan_level": 40
6   },
7   "msg": "指令已发送"
8 }
```

3.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

3.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	object	true	none		none
»» fan_mode	integer	true	none		风扇控制状态
»» fan_level	integer	true	none		风扇的挡位
» msg	string	true	none		响应信息

MqttController/自动化

1. GET GetLightAuto

GET /secure/Get_Light_auto

打开灯光得自动化状态。

1.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	否	none

返回示例

成功

```
1 {
2   "code": 200,
3   "data": false,
4   "msg": "灯光自动化模式已开"
5 }
```

1.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

1.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	boolean	true	none		打开后返回true，关闭返回false
» msg	string	true	none		响应信息

2. GET GetPumpAuto

GET /secure/Get_Pump_auto

水泵得自动化开关

2.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	是	用户验证生成的token进行后端身份验证，由登陆时生成，前端进行保存携带验证

返回示例

成功

```
1 {
2   "code": 200,
3   "data": true,
4   "msg": "水泵自动化模式已开"
5 }
```

2.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

2.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	boolean	true	none		打开后返回true，关闭返回false
» msg	string	true	none		响应信息

3. GET GetFenAuto

GET /secure/Get_Fen_auto

开启风扇的自动化控制

3.1. 请求参数

名称	位置	类型	必选	说明
Authorization	header	string	是	用户验证生成的token进行后端身份验证，由登陆时生成，前端进行保存携带验证

返回示例

成功

```
1 {  
2   "code": 200,  
3   "data": true,  
4   "msg": "风扇自动化模式已开"  
5 }
```

3.2. 返回结果

状态码	状态码含义	说明	数据模型
200	OK	成功	Inline

3.3. 返回数据结构

状态码 200

名称	类型	必选	约束	中文名	说明
» code	integer	true	none		响应码
» data	boolean	true	none		打开后返回true，关闭返回false
» msg	string	true	none		响应信息

数据模型

1. registerInfo

```
1 {
2   "phoneNumber": "string",
3   "password": "string",
4   "code": "string"
5 }
6
```

1.1. 属性

名称	类型	必选	约束	中文名	说明
phoneNumber	string	true	none		用户手机号
password	string	true	none		用户设置密码
code	string	true	none		注册时获取的验证码

2. UserInfo

```
1 {
2   "id": 0,
3   "nickname": "string",
4   "email": "string",
5   "address": "string"
6 }
7
```

2.1. 属性

名称	类型	必选	约束	中文名	说明
id	integer	false	none		ID 编号
nickname	string	false	none		昵称
email	string	false	none		电子邮箱
address	string	false	none		地址

3. User

```
1 {
2   "id": 0,
3   "password": "string",
4   "phoneNumber": "string"
5 }
6
```

3.1. 属性

名称	类型	必选	约束	中文名	说明
id	integer	true	none		id编号
password	string	true	none		密码
phoneNumber	string	true	none		手机号