

物联网设备管理平台 接口文档

1 后端统一响应接口定义

2 前端各页面接口定义

2.1 登录Login

2.1.1 登录请求

2.2 注册Register

2.2.1 注册请求

2.3 首页Home

2.3.1 图表数据请求

2.4 设备管理Device

2.4.1 设备信息请求

2.4.2 设备信息保存或更改请求

2.4.3 删除设备请求

2.4.4 批量删除设备请求

2.5 设备消息Message

2.5.1 设备消息请求

2.5.2 删除设备消息请求

2.5.3 批量删除设备消息请求

2.6 历史位置History

2.6.1 请求所有设备名称

2.6.2 请求某设备历史位置

2.7 个人信息Person

2.7.1 获取用户信息请求

2.7.2 更新用户信息（除密码）请求

2.7.3 更新用户密码请求

2.8 框架界面Manage

2.8.1 获取用户名请求

1 后端统一响应接口定义

- 在 `BS/BS_Backend/src/main/java/com/bs/BS_Backend/common/Result.java` 中定义

▼ Result.java

Java

```
1 public class Result {
2     private String code; // 代码
3     private String msg; // 信息
4     private Object data; // 数据
5 }
```

- 具体响应类型代码在 `BS/BS_Backend/src/main/java/com/bs/BS_Backend/common/Constants.java` 中定义

▼ Contants.java

Java

```
1 public interface Constants {
2     String CODE_PERMISSION_ERROR = "-1"; // 权限错误
3     String CODE_SUCCESS = "0"; // 成功
4     String CODE_ERROR = "1"; // 错误
5     String CODE_NAME_NOT_EXIST = "2"; // 用户名不存在
6     String CODE_PASSWORD_ERROR = "3"; // 密码错误
7     String CODE_NAME_DUPLICATE = "4"; // 用户名或设备名重复
8     String CODE_EMAIL_DUPLICATE = "5"; // 邮箱重复
9 }
```

2 前端各页面接口定义

- 其中，所有前端请求（除登录与注册请求外）均带有头部 `token` 用于身份验证

2.1 登录 Login

2.1.1 登录请求

- 请求 URL : `/user/login`
- 方法: `POST`
- 请求参数: `user`
- 其中，`user` 定义如下，将发送用户名与密码：

```
▼ user JavaScript |
1 {
2   username: // 用户名
3   password: // 密码
4 }
```

- 响应值: `Result`

code	message	data	含义
0		userDTO	登录成功
2	用户名不存在		用户名不存在
3	密码错误		密码错误

- 其中, `userDTO` 定义如下, 将返回用户 `ID`、用户名、密码、`token`

```
▼ userDTO.java Java |
1 public class UserDTO {
2     private Integer userID; // ID
3     private String username; // 用户名
4     private String password; // 密码
5     private String token; // token
6 }
```

2.2 注册 Register

2.2.1 注册请求

- 请求 URL : `/user/register`
- 方法: `POST`
- 请求参数: `data`
- 其中, `data` 定义如下, 将发送用户名、密码与邮箱:

```
▼ data JavaScript |
1 {
2   username: // 用户名
3   password: // 密码
4   email:    // 邮箱
5 }
```

- 响应值: `Result`

code	message	data	含义
0			注册成功
1	系统错误		数据库操作失败
4	用户名重复		用户名重复
5	邮箱重复		邮箱重复

2.3 首页 Home

2.3.1 图表数据请求

- 请求 URL : `/echarts/info`
- 方法: `GET`
- 响应值: `Result`

code	message	data	含义
0		map	请求图表数据成功
-1	token错误, 请重新登录		权限错误或登录过期
1	系统错误		数据库操作失败

- 其中, `map` (为 `Map` 类型) 定义如下, 将返回前端图表所需的数据:

map

Java

```

1  Map<String, Object>;
2  {
3      "x": ArrayList<String>; // x 轴, 设备类型列表
4      "y": ArrayList<Integer>; // y 轴, 设备类型数量列表
5      "deviceNum": Integer; // 设备总数
6      "normalNum": Integer; // 正常设备总数
7      "messageNormalNum": Integer; // 正常消息总数
8      "messageWarningNum": Integer; // 警告消息总数
9      "messageErrorNum": Integer; // 错误消息总数
10     "deviceName": ArrayList<String>; // 设备名称列表
11     "normal": ArrayList<Integer>; // 每个设备正常消息数量列表
12     "warning": ArrayList<Integer>; // 每个设备警告消息数量列表
13     "error": ArrayList<Integer>; // 每个设备错误消息数量列表
14 }

```

2.4 设备管理 Device

2.4.1 设备信息请求

- 请求 URL : /device/page
- 方法: GET
- 请求参数定义如下:

params

JavaScript

```

1  {
2      pageNum:      // 页码
3      pageSize:     // 每页大小
4      deviceName:   // 设备名称
5      type:         // 设备类型
6  }

```

- 响应值: Result

code	message	data	含义
0		Page	请求设备数据成功
-1	token错误, 请重新登录		权限错误或登录过期

1	系统错误	数据库操作失败
---	------	---------

- 其中，`Page` 定义如下：

▼ Page	JSON
<pre> 1 { 2 "records": [3 { 4 "deviceID": , // 设备 ID 5 "deviceName": , // 设备名称 6 "description": , // 设备描述 7 "type": , // 设备类型 8 "userID": // 用户 ID 9 }, 10 ... // 其余设备信息 11], 12 "total": , // 总数 13 "size": , // 每页大小 14 "current": , // 页码 15 "orders": [], 16 "optimizeCountSql": true, 17 "searchCount": true, 18 "countId": null, 19 "maxLimit": null, 20 "pages": // 页数 21 }</pre>	

2.4.2 设备信息保存或更改请求

- 请求 URL : `/device/saveOrUpdate`
- 方法: `POST`
- 请求参数定义如下：

▼ params	JavaScript
<pre> 1 { 2 deviceName: // 设备名称 3 type: // 设备类型 4 description: // 设备描述 5 userID: // 用户 ID 6 }</pre>	

- 响应值: **Result**

code	message	data	含义
0			保存或更改设备数据成功
-1	token错误, 请重新登录		权限错误或登录过期
1	系统错误		数据库操作失败
4	设备名重复		设备名重复

2.4.3 删除设备请求

- 请求 URL : **/device/delete/{deviceID}**
- 方法: **DELETE**
- 其中, **deviceID** 为所需删除设备 **ID**
- 响应值: **Result**

code	message	data	含义
0			删除设备成功
-1	token错误, 请重新登录		权限错误或登录过期
1	系统错误		数据库操作失败

2.4.4 批量删除设备请求

- 请求 URL : **/device/delete/batch**
- 方法: **POST**
- 请求参数: 存有所需删除设备 **ID** 的列表, 即 **[ID1, ID2, ID3, ...]**
- 响应值: **Result**

code	message	data	含义
0			批量删除设备成功

-1	token错误, 请重新登录		权限错误或登录过期
1	系统错误		数据库操作失败

2.5 设备消息 Message

2.5.1 设备消息请求

- 请求 URL : /message/page
- 方法: GET
- 请求参数定义如下:

params JavaScript

```
1 {
2   pageNum: ,      // 页码
3   pageSize: ,     // 每页大小
4   deviceName: ,   // 设备名称
5   type: ,         // 设备类型
6   messageTime: ,  // 消息时间
7   check: check,   // 查询编码值
8   ascend: ascend  // 是否按时间正序
9 }
```

- 其中, 查询编码值规定如下:

查询类型编码值: 正常为 001, 警告为010, 错误为100
三者相加得到最终将查询的类型组合, 如 101 代表 正常 + 错误消息

- 响应值: Result

code	message	data	含义
0		Page	请求消息数据成功
-1	token错误, 请重新登录		权限错误或登录过期
1	系统错误		数据库操作失败

- 其中， `Page` 定义如下：

▼

Page

JSON

```

1  {
2    "records": [
3      {
4        "deviceID": , // 设备 ID
5        "info": , // 消息内容
6        "alert": , // 状态值
7        "longitude": , // 经度
8        "latitude": , // 纬度
9        "messageTime": , // 消息时间
10       "deviceName": , // 设备名称
11       "type": // 设备类型
12     },
13     ... // 其余设备消息
14   ],
15   "total": , // 总数
16   "size": , // 每页大小
17   "current": , // 页码
18   "orders": [],
19   "optimizeCountSql": true,
20   "searchCount": true,
21   "countId": null,
22   "maxLimit": null,
23   "pages": // 页数
24 }

```

2.5.2 删除设备消息请求

- 请求 URL : `/message/delete/{deviceID}/{messageTime}`
- 方法: `DELETE`
- 请求参数: `deviceID` 与 `messageTime` 分别对应所需删除消息的设备 ID 与时间
- 响应值: `Result`

code	message	data	含义
0			删除消息成功
-1	token错误, 请重新登录		权限错误或登录过期
1	系统错误		数据库操作失败

2.5.3 批量删除设备消息请求

- 请求 URL : /message/delete/batch
- 方法: POST
- 请求参数: IDTimeList
- 其中, IDTimeList 定义如下:

IDTimeList

JavaScript

```
1  [
2    [
3      deviceID1,    // 设备 ID 1
4      messageTime1 // 消息时间 1
5    ],
6    [
7      deviceID2,    // 设备 ID 2
8      messageTime2 // 消息时间 2
9    ],
10   ...
11  ]
```

- 响应值: Result

code	message	data	含义
0			批量删除消息成功
-1	token错误, 请重新登录		权限错误或登录过期
1	系统错误		数据库操作失败

2.6 历史位置 History

2.6.1 请求所有设备名称

- 请求 URL : /device/name
- 方法: GET
- 响应值: Result

code	message	data	含义
0		data	批量删除消息成功
-1	token错误, 请重新登录		权限错误或登录过期
1	系统错误		数据库操作失败

- 其中, `data` 定义如下:

data	Java
<pre> 1 Map<String, List<String>>; 2 // 对应 --> 3 "deviceName": deviceName1, deviceName2, ... </pre>	

2.6.2 请求某设备历史位置

- 请求 URL : `/device/location/{deviceName}`
- 方法: `GET`
- 请求参数: `deviceName` 即为所请求设备的名称
- 响应值: `Result`

code	message	data	含义
0		data	请求设备历史位置成功
-1	token错误, 请重新登录		权限错误或登录过期
1	系统错误		数据库操作失败

- 其中, `data` 定义如下:

```
▼ data Java
1  Map<String, List<String>>;
2  // 对应 -->
3  {
4      "longitude": longitude1, longitude2, ...; // 经度列表
5      "latitude": latitude1, latitude2, ...; // 纬度列表
6      "alert": alert1, alert2, ...; // 状态列表
7      "messageTime": messageTime1, messageTime2, ...; // 消息时间列表
8      "info": info1, info2, ...; // 消息内容列表
9  }
```

2.7 个人信息 **Person**

2.7.1 获取用户信息请求

- 请求 URL : `user/userinfo`
- 方法: `GET`
- 响应值: `Result`

code	message	data	含义
0		user	请求设备历史位置成功
-1	token错误, 请重新登录		权限错误或登录过期
1	系统错误		数据库操作失败

- 其中, `user` 即为 `User` 类, 其定义如下:

```
▼ user Java
1  public class User {
2      private Integer userID; // 用户 ID
3      private String username; // 用户名
4      private String password; // 密码
5      private String email; // 邮箱
6      private String phone; // 手机号
7      private String address; // 地址
8  }
```

2.7.2 更新用户信息（除密码）请求

- 请求 URL： /user/update
- 方法： POST
- 请求参数：即 user 类
- 响应值： Result

code	message	data	含义
0			注册成功
-1	token错误, 请重新登录		权限错误或登录过期
1	系统错误		数据库操作失败
4	用户名重复		用户名重复
5	邮箱重复		邮箱重复

2.7.3 更新用户密码请求

- 请求 URL： /user/updatePassword
- 方法： POST
- 请求参数： [oldPassworld, newPassworld] ，即包含旧密码与新密码的 []
- 响应值： Result

code	message	data	含义
0			注册成功
-1	token错误, 请重新登录		权限错误或登录过期
1	系统错误		数据库操作失败
3	原密码错误		提供的原密码错误

2.8 框架界面 Manage

2.8.1 获取用户名请求

- 请求 URL : user/username
- 方法: GET
- 响应值: Result

code	message	data	含义
0		username	注册成功
-1	token错误, 请重新登录		权限错误或登录过期
1	系统错误		数据库操作失败

- 其中, 返回的 username 为 String 类型, 即所请求的用户名