

# Table of Contents

源..... 1

Function-controller ..... 1

Machinery-controller ..... 3

Role-controller ..... 13

Stat-controller..... 21

User-controller ..... 25

Work-controller ..... 39

# 源

## Function-controller

方法操作控制

根据角色添加方法 限

POST /function/allotFunciton

明

根据角色添加方法 限接口

参数

型	名称	明	型
Body	<b>functiondata</b> 必填	方法的ID集合	string
Body	<b>roleID</b> 必填	角色的ID	Integer

HTTP代	明	型
200	OK	Result
201	Created	无内容
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- application/json

生成

- /

HTTP 求示例

求 path

```
/function/allotFunciton
```

求 body

```
{ }
```

HTTP 示例

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

取所有 限

```
GET /function/getAll
```

明

取所有 限接口

HTTP代	明	型
200	OK	<a href="#">Result</a>
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- `application/json`

生成

• /

HTTP 求示例

求 path

```
/function/getAll
```

HTTP 示例

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

# Machinery-controller

机械操作

取所有 机械信息

```
GET /machinery/getAll
```

明

取所有 机械信息接口

参数

型	名称	明	型
Query	count 必填	分 - 一 几条数据	ref
Query	page 必填	分 - 第几	ref
Body	Machinery 可	条件	<a href="#">com.puyan.shengren.agricultural.Enity.Machinery</a>

HTTP代	明	型
200	OK	Result
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- application/json

生成

- /

HTTP 求示例

求 path

```
/machinery/getAll
```

求 query

```
{
  "count" : "ref",
  "page" : "ref"
}
```

HTTP 示例

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

添加 机械

POST /machinery/machineryInsert

明

添加 机械接口

参数

型	名称	明	型
Query	<b>machinery</b> 必填	机械 体	ref
Query	<b>machineryBrand</b> 可	机械品牌	string
Query	<b>machineryLicense</b> 可	机械 牌号	string
Query	<b>machineryNO</b> 可	机械 号	string
Query	<b>machineryOwner</b> 可	机械所有者ID	integer (int32)
Query	<b>machineryStatus</b> 可	机状 1正常 2 坏 3 修 4 除	integer (int32)
Query	<b>machineryWidth</b> 可	机械 度	number (double)
Query	<b>user.city</b> 可	市	string
Query	<b>user.cooperative</b> 可	合作社	string
Query	<b>user.county</b> 可		string
Query	<b>user.createDate</b> 可	建	string (date-time)
Query	<b>user.creator</b> 可	建人	integer (int32)

型	名称	明	型
Query	user.email 可	箱	string
Query	user.lastLogin Time 可	上一次登	string (date-time)
Query	user.passWor d 可	密	string
Query	user.phone 可		string
Query	user.province 可	省	string
Query	user.status 可	用 状 1正常 -1禁用	integer (int32)
Query	user.town 可		string
Query	user.type 可	人事 型	integer (int32)
Query	user.updated ate 可	更新	string (date-time)
Query	user.updator 可	更新人	integer (int32)
Query	user.userName 可	用 名	string
Query	user.village 可	村	string

HTTP代	明	型
200	OK	<a href="#">Result</a>
201	Created	无内容
401	Unauthorized	无内容
403	Forbidden	无内容

HTTP代	明	型
404	Not Found	无内容

消耗

- application/json

生成

- /

HTTP 求示例

求 path

/machinery/machineryInsert

求 query

```
{
  "machinery" : "ref",
  "machineryBrand" : "string",
  "machineryLicense" : "string",
  "machineryNO" : "string",
  "machineryOwner" : 0,
  "machineryStatus" : 0,
  "machineryWidth" : 0.0,
  "user.city" : "string",
  "user.cooperative" : "string",
  "user.county" : "string",
  "user.createDate" : "string",
  "user.creator" : 0,
  "user.email" : "string",
  "user.lastLoginTime" : "string",
  "user.passWord" : "string",
  "user.phone" : "string",
  "user.province" : "string",
  "user.status" : 0,
  "user.town" : "string",
  "user.type" : 0,
  "user.updateDate" : "string",
  "user.updator" : 0,
  "user.userName" : "string",
  "user.village" : "string"
}
```



HTTP 示例

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

修改 机械信息

PUT /machinery/update

明

修改 机械信息接口

参数

型	名称	明	型
Query	machineryBrand 可	机械品牌	string
Query	machineryLicense 可	机械 牌号	string
Query	machineryNO 可	机械 号	string
Query	machineryOwner 可	机械所有者ID	integer (int32)
Query	machineryStatus 可	机状 1正常 2 坏 3 修 4 除	integer (int32)
Query	machineryWidth 可	机械 度	number (double)
Query	user.city 可	市	string

型	名称	明	型
Query	<b>user.cooperative</b> 可	合作社	string
Query	<b>user.county</b> 可		string
Query	<b>user.createDate</b> 可	建	string (date-time)
Query	<b>user.creator</b> 可	建人	integer (int32)
Query	<b>user.email</b> 可	箱	string
Query	<b>user.lastLoginTime</b> 可	上一次登	string (date-time)
Query	<b>user.password</b> 可	密	string
Query	<b>user.phone</b> 可		string
Query	<b>user.province</b> 可	省	string
Query	<b>user.status</b> 可	用 状 1正常 -1禁用	integer (int32)
Query	<b>user.town</b> 可		string
Query	<b>user.type</b> 可	人事 型	integer (int32)
Query	<b>user.updatedDate</b> 可	更新	string (date-time)
Query	<b>user.updator</b> 可	更新人	integer (int32)
Query	<b>user.userName</b> 可	用 名	string

型	名称	明	型
Query	<b>user.village</b> 可	村	string
Body	<b>machinery</b> 必填	机械 体	<a href="#">com.puyan.shengren.agricultural.Enity.Machinery</a>

HTTP代	明	型
200	OK	<a href="#">Result</a>
201	Created	无内容
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- `application/json`

生成

- `/`

HTTP 求示例

求 path

`/machinery/update`

求 query

```
{
  "machineryBrand" : "string",
  "machineryLicense" : "string",
  "machineryNO" : "string",
  "machineryOwner" : 0,
  "machineryStatus" : 0,
  "machineryWidth" : 0.0,
  "user.city" : "string",
  "user.cooperative" : "string",
  "user.county" : "string",
  "user.createDate" : "string",
  "user.creator" : 0,
  "user.email" : "string",
  "user.lastLoginTime" : "string",
  "user.passWord" : "string",
  "user.phone" : "string",
  "user.province" : "string",
  "user.status" : 0,
  "user.town" : "string",
  "user.type" : 0,
  "user.updateDate" : "string",
  "user.updator" : 0,
  "user.userName" : "string",
  "user.village" : "string"
}
```

## HTTP 示例

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

## 更改 机械状

PUT /machinery/updateStatus

明

更改 机械状 接口

参数

型	名称	明	型
Body	<b>machineryID</b> 必填	机械ID	<a href="#">Integer</a>
Body	<b>machinerySta tus</b> 必填	机械状	<a href="#">Integer</a>

HTTP代	明	型
200	OK	<a href="#">Result</a>
201	Created	无内容
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- `application/json`

生成

- `/`

HTTP 求示例

求 path

```
/machinery/updateStarus
```

HTTP 示例

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

# Role-controller

角色操作

用 添加角色

POST /role/allotRole

明

用 添加角色接口

参数

型	名称	明	型
Body	roleID 必填	角色ID集合	String
Body	userID 必填	用 ID	Integer

HTTP代	明	型
200	OK	Result
201	Created	无内容
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- application/json

生成

- /

HTTP 求示例

求 path

/role/allotRole

HTTP 示例

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

所有角色

GET /role/getAll

明

所有角色接口

参数

型	名称	明	型
Query	<b>createDate</b> 可	建	string (date-time)
Query	<b>creator</b> 可	建人	integer (int32)
Query	<b>remark</b> 可	注	string
Query	<b>roleName</b> 可	角色名称	string
Query	<b>status</b> 可	状 1正常 -1禁用	integer (int32)
Query	<b>updateDate</b> 可	修改	string (date-time)
Query	<b>updater</b> 可	修改人	integer (int32)

型	名称	明	型
Body	role 必填	条件	<a href="#">com.puyan.shengren.agricultural.Enity.Role</a>

HTTP代	明	型
200	OK	<a href="#">Result</a>
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- `application/json`

生成

- `/`

HTTP 求示例

求 path

```
/role/getAll
```

求 query

```
{
  "createDate" : "string",
  "creator" : 0,
  "remark" : "string",
  "roleName" : "string",
  "status" : 0,
  "updateDate" : "string",
  "updater" : 0
}
```

HTTP 示例



```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

添加角色

POST /role/register

明

添加角色接口

参数

型	名称	明	型
Query	createDate 可	建	string (date-time)
Query	creator 可	建人	integer (int32)
Query	remark 可	注	string
Query	roleName 可	角色名称	string
Query	status 可	状 1正常 -1禁用	integer (int32)
Query	updateDate 可	修改	string (date-time)
Query	updater 可	修改人	integer (int32)
Body	role 必填	角色 体	<a href="#">com.puyan.shengren.agricultural.Enity.Role</a>

HTTP代	明	型
200	OK	Result
201	Created	无内容
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- application/json

生成

- /

HTTP 求示例

求 path

```
/role/register
```

求 query

```
{
  "createDate" : "string",
  "creator" : 0,
  "remark" : "string",
  "roleName" : "string",
  "status" : 0,
  "updateDate" : "string",
  "updator" : 0
}
```

HTTP 示例

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

修改角色

PUT /role/update

明

修改角色接口

参数

型	名称	明	型
Query	<b>createDate</b> 可	建	string (date-time)
Query	<b>creator</b> 可	建人	integer (int32)
Query	<b>remark</b> 可	注	string
Query	<b>roleName</b> 可	角色名称	string
Query	<b>status</b> 可	状 1正常 -1禁用	integer (int32)
Query	<b>updateDate</b> 可	修改	string (date-time)
Query	<b>updater</b> 可	修改人	integer (int32)
Body	<b>role</b> 必填	角色 体	<a href="#">com.puyan.shengren.agricultural.Enity.Role</a>

HTTP代	明	型
200	OK	<a href="#">Result</a>

HTTP代	明	型
201	Created	无内容
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- application/json

生成

- /

HTTP 求示例

求 path

```
/role/update
```

求 query

```
{
  "createDate" : "string",
  "creator" : 0,
  "remark" : "string",
  "roleName" : "string",
  "status" : 0,
  "updateDate" : "string",
  "updater" : 0
}
```

HTTP 示例

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

更改角色状

PUT /role/updateStatus

明

更改角色状 接口

参数

型	名称	明	型
Body	<b>roleID</b> 必填	角色ID	Integer
Body	<b>status</b> 必填	状	Integer

HTTP代	明	型
200	OK	Result
201	Created	无内容
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- application/json

生成

- /

HTTP 求示例

求 path

/role/updateStatus

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

# Stat-controller

信息控制

/区

```
GET /stat/getAllByCounty
```

明

/区 接口

参数

型	名称	明	型
Body	count 必填	分 - 一 几条数据	Integer
Body	page 必填	分 - 第几	Integer

HTTP代	明	型
200	OK	Result
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- application/json

生成

- /

HTTP 求示例

求 path

```
/stat/getAllByCounty
```

HTTP 示例

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

机 型

```
GET /stat/getAllByMachineryID
```

明

机 型

参数

型	名称	明	型
Body	count 必填	分 - 一 几条数据	Integer
Body	page 必填	分 - 第几	Integer

HTTP代	明	型
200	OK	<a href="#">Result</a>
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- `application/json`

生成

- `/`

HTTP 求示例

求 path

```
/stat/getAllByMachineryID
```

HTTP 示例

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

去7天的作 面

```
GET /stat/getPastSeven
```

明

去7天的作 面 接口



HTTP代	明	型
200	OK	Result
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- application/json

生成

- /

HTTP 求示例

求 path

```
/stat/getPastSeven
```

HTTP 示例

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

所有/昨天/今天/ 的 机数，与作 面

```
GET /stat/getStatsWookNum
```

明

接口

HTTP代	明	型
200	OK	Result
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- application/json

生成

- /

HTTP 求示例

求 path

```
/stat/getStatsWookNum
```

HTTP 示例

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

# User-controller

用 操作控制

修改密

```
PUT /user/forgetResetPassword
```

明

修改密 接口

参数

型	名称	明	型
Body	<b>newPassWord</b> 必填	新密	String
Body	<b>passWord</b> 必填	旧密	String
Body	<b>updater</b> 必填	修改人-当前登 的ID	Integer
Body	<b>userName</b> 必填	用 名	String

HTTP代	明	型
200	OK	ServerResponse«string»
201	Created	无内容
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- application/json

生成

- /

HTTP 求示例

求 path

```
/user/forgetResetPassword
```

HTTP 示例

200

```
{
  "data" : "string",
  "msg" : "string",
  "status" : 0,
  "success" : true
}
```

## 用 信息

GET /user/getAll

明

用 信息接口

参数

型	名称	明	型
Query	<b>city</b> 可	市	string
Query	<b>cooperative</b> 可	合作社	string
Query	<b>county</b> 可		string
Query	<b>createDate</b> 可	建	string (date-time)
Query	<b>creator</b> 可	建人	integer (int32)
Query	<b>email</b> 可	箱	string
Query	<b>lastLoginTime</b> 可	上一次登	string (date-time)
Query	<b>passWord</b> 可	密	string
Query	<b>phone</b> 可		string
Query	<b>province</b> 可	省	string

型	名称	明	型
Query	<b>status</b> 可	用 状 1正常 -1禁用	integer (int32)
Query	<b>town</b> 可		string
Query	<b>type</b> 可	人事 型	integer (int32)
Query	<b>updateDate</b> 可	更新	string (date-time)
Query	<b>updater</b> 可	更新人	integer (int32)
Query	<b>userName</b> 可	用 名	string
Query	<b>village</b> 可	村	string
Body	<b>count</b> 必填	分 - 一 几条数据	Integer
Body	<b>page</b> 必填	分 - 第几	Integer
Body	<b>user</b> 可	条件	com.puyan.shengren. agricultural.Enity.User

HTTP代	明	型
200	OK	Result
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- application/json

生成

- /

## HTTP 求示例

求 path

```
/user/getAll
```

求 query

```
{
  "city" : "string",
  "cooperative" : "string",
  "county" : "string",
  "createDate" : "string",
  "creator" : 0,
  "email" : "string",
  "lastLoginTime" : "string",
  "passWord" : "string",
  "phone" : "string",
  "province" : "string",
  "status" : 0,
  "town" : "string",
  "type" : 0,
  "updateDate" : "string",
  "updater" : 0,
  "userName" : "string",
  "village" : "string"
}
```

## HTTP 示例

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

根据ID 用

```
GET /user/getUserById
```

明

根据ID 用 接口

参数

型	名称	明	型
Body	userID 必填	用 ID	Integer

HTTP代	明	型
200	OK	Result
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- application/json

生成

- /

HTTP 求示例

求 path

/user/getUserById

HTTP 示例

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

用 登

GET /user/login

明

用 登 接口

参数

型	名称	明	型
Body	passWord 必填	密	string
Body	userName 必填	用 名	string

HTTP代	明	型
200	OK	<a href="#">ServerResponse«user 象»</a>
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- application/json

生成

- /

HTTP 求示例

求 path

/user/login

求 body



```
{ }
```

## HTTP 示例

200

```
{
  "data" : {
    "city" : "string",
    "cooperative" : "string",
    "county" : "string",
    "createDate" : "string",
    "creator" : 0,
    "email" : "string",
    "lastLoginTime" : "string",
    "passWord" : "string",
    "phone" : "string",
    "province" : "string",
    "status" : 0,
    "town" : "string",
    "type" : 0,
    "updateDate" : "string",
    "updator" : 0,
    "userName" : "string",
    "village" : "string"
  },
  "msg" : "string",
  "status" : 0,
  "success" : true
}
```

## 建用

POST /user/register

明

建用 接口

参数

型	名称	明	型
Query	<b>city</b> 可	市	string
Query	<b>cooperative</b> 可	合作社	string
Query	<b>county</b> 可		string
Query	<b>createDate</b> 可	建	string (date-time)
Query	<b>creator</b> 可	建人	integer (int32)
Query	<b>email</b> 可	箱	string
Query	<b>lastLoginTime</b> 可	上一次登	string (date-time)
Query	<b>passWord</b> 可	密	string
Query	<b>phone</b> 可		string
Query	<b>province</b> 可	省	string
Query	<b>status</b> 可	用 状 1正常 -1禁用	integer (int32)
Query	<b>town</b> 可		string
Query	<b>type</b> 可	人事 型	integer (int32)
Query	<b>updateDate</b> 可	更新	string (date-time)
Query	<b>updater</b> 可	更新人	integer (int32)
Query	<b>userName</b> 可	用 名	string
Query	<b>village</b> 可	村	string

型	名称	明	型
Body	<b>user</b> 必填	用 体	<a href="#">com.puyan.shengren.agricultural.Enity.User</a>

HTTP代	明	型
200	OK	<a href="#">ServerResponse«string»</a>
201	Created	无内容
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- [application/json](#)

生成

- [/](#)

HTTP 求示例

求 path

```
/user/register
```

求 query

```
{
  "city" : "string",
  "cooperative" : "string",
  "county" : "string",
  "createDate" : "string",
  "creator" : 0,
  "email" : "string",
  "lastLoginTime" : "string",
  "passWord" : "string",
  "phone" : "string",
  "province" : "string",
  "status" : 0,
  "town" : "string",
  "type" : 0,
  "updateDate" : "string",
  "updater" : 0,
  "userName" : "string",
  "village" : "string"
}
```

## HTTP 示例

200

```
{
  "data" : "string",
  "msg" : "string",
  "status" : 0,
  "success" : true
}
```

## 修改用 信息

PUT /user/update

明

修改用 信息接口

参数

型	名称	明	型
Query	<b>city</b> 可	市	string
Query	<b>cooperative</b> 可	合作社	string
Query	<b>county</b> 可		string
Query	<b>createDate</b> 可	建	string (date-time)
Query	<b>creator</b> 可	建人	integer (int32)
Query	<b>email</b> 可	箱	string
Query	<b>lastLoginTime</b> 可	上一次登	string (date-time)
Query	<b>passWord</b> 可	密	string
Query	<b>phone</b> 可		string
Query	<b>province</b> 可	省	string
Query	<b>status</b> 可	用 状 1正常 -1禁用	integer (int32)
Query	<b>town</b> 可		string
Query	<b>type</b> 可	人事 型	integer (int32)
Query	<b>updateDate</b> 可	更新	string (date-time)
Query	<b>updater</b> 可	更新人	integer (int32)
Query	<b>userName</b> 可	用 名	string
Query	<b>village</b> 可	村	string

型	名称	明	型
Body	<b>user</b> 必填	用 体	<a href="#">com.puyan.shengren. agricultural.Enity.User</a>

HTTP代	明	型
200	OK	<a href="#">Result</a>
201	Created	无内容
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- [application/json](#)

生成

- [/](#)

HTTP 求示例

求 path

/user/update

求 query

```
{
  "city" : "string",
  "cooperative" : "string",
  "county" : "string",
  "createDate" : "string",
  "creator" : 0,
  "email" : "string",
  "lastLoginTime" : "string",
  "passWord" : "string",
  "phone" : "string",
  "province" : "string",
  "status" : 0,
  "town" : "string",
  "type" : 0,
  "updateDate" : "string",
  "updater" : 0,
  "userName" : "string",
  "village" : "string"
}
```

## HTTP 示例

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

## 修改用 状

PUT /user/updateStatus

明

## 修改用 状 接口

### 参数

型	名称	明	型
Body	status 必填	需要修改成的状	<a href="#">Integer</a>

型	名称	明	型
Body	userID 必填	用 ID	Integer

HTTP代	明	型
200	OK	Result
201	Created	无内容
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- application/json

生成

- /

HTTP 求示例

求 path

```
/user/updateStatus
```

HTTP 示例

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

# Work-controller

作 信息控制



根据ID 除作 信息

```
DELETE /work/delete
```

明

根据ID 除作 信息接口

参数

型	名称	明	型
Body	workID 必填	作 信息ID	Integer

HTTP代	明	型
200	OK	Result
204	No Content	无内容
401	Unauthorized	无内容
403	Forbidden	无内容

消耗

- application/json

生成

- /

HTTP 求示例

求 path

```
/work/delete
```

HTTP 示例

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

所有作 信息

GET /work/getAll

明

所有作 信息接口

参数

型	名称	明	型
Query	city 可	市	string
Query	count 必填	分 - 一 几条数据	ref
Query	county 可		string
Query	machinery.machineryBrand 可	机械品牌	string
Query	machinery.machineryLicense 可	机械 牌号	string
Query	machinery.machineryNO 可	机械 号	string
Query	machinery.machineryOwner 可	机械所有者ID	integer (int32)
Query	machinery.machineryStatus 可	机状 1正常 2 坏 3 修 4 除	integer (int32)

型	名称	明	型
Query	<b>machinery.machineryWidth</b> 可	机械 度	number (double)
Query	<b>machinery.user.city</b> 可	市	string
Query	<b>machinery.user.cooperative</b> 可	合作社	string
Query	<b>machinery.user.county</b> 可		string
Query	<b>machinery.user.createDate</b> 可	建	string (date-time)
Query	<b>machinery.user.creator</b> 可	建人	integer (int32)
Query	<b>machinery.user.email</b> 可	箱	string
Query	<b>machinery.user.lastLoginTime</b> 可	上一次登	string (date-time)
Query	<b>machinery.user.passWord</b> 可	密	string
Query	<b>machinery.user.phone</b> 可		string
Query	<b>machinery.user.province</b> 可	省	string
Query	<b>machinery.user.status</b> 可	用 状 1正常 -1禁用	integer (int32)

型	名称	明	型
Query	<b>machinery.user.town</b> 可		string
Query	<b>machinery.user.type</b> 可	人事 型	integer (int32)
Query	<b>machinery.user.updateDate</b> 可	更新	string (date-time)
Query	<b>machinery.user.updator</b> 可	更新人	integer (int32)
Query	<b>machinery.user.userName</b> 可	用 名	string
Query	<b>machinery.user.village</b> 可	村	string
Query	<b>page</b> 必填	分 - 第几	ref
Query	<b>province</b> 可	省	string
Query	<b>town</b> 可		string
Query	<b>verificationResults</b> 可	核 果	integer (int32)
Query	<b>village</b> 可	村	string
Query	<b>workArea</b> 可	作 面	number (double)
Query	<b>workDepth</b> 可	作 深度	number (double)
Query	<b>workEndMap</b> 可	作 束坐	string
Query	<b>workEndTime</b> 可	作 束	string (date-time)

型	名称	明	型
Query	<b>workLength</b> 可	作 里程	number (double)
Query	<b>workMachine</b> <b>ryID</b> 可	作 的机械ID	integer (int32)
Query	<b>workStartMap</b> 可	作 始坐	string
Query	<b>workStartTime</b> 可	作 始	string (date-time)
Body	<b>work</b> 可	条件	<a href="#">com.puyan.shengren.agricultural.Entity.Work</a>

HTTP代	明	型
200	OK	<a href="#">Result</a>
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- `application/json`

生成

- `/`

HTTP 求示例

求 `path`

`/work/getAll`

求 `query`

```

{
  "city" : "string",
  "count" : "ref",
  "county" : "string",
  "machinery.machineryBrand" : "string",
  "machinery.machineryLicense" : "string",
  "machinery.machineryNO" : "string",
  "machinery.machineryOwner" : 0,
  "machinery.machineryStatus" : 0,
  "machinery.machineryWidth" : 0.0,
  "machinery.user.city" : "string",
  "machinery.user.cooperative" : "string",
  "machinery.user.county" : "string",
  "machinery.user.createDate" : "string",
  "machinery.user.creator" : 0,
  "machinery.user.email" : "string",
  "machinery.user.lastLoginTime" : "string",
  "machinery.user.passWord" : "string",
  "machinery.user.phone" : "string",
  "machinery.user.province" : "string",
  "machinery.user.status" : 0,
  "machinery.user.town" : "string",
  "machinery.user.type" : 0,
  "machinery.user.updateDate" : "string",
  "machinery.user.updator" : 0,
  "machinery.user.userName" : "string",
  "machinery.user.village" : "string",
  "page" : "ref",
  "province" : "string",
  "town" : "string",
  "verificationResults" : 0,
  "village" : "string",
  "workArea" : 0.0,
  "workDepth" : 0.0,
  "workEndMap" : "string",
  "workEndTime" : "string",
  "workLength" : 0.0,
  "workMachineryID" : 0,
  "workStartMap" : "string",
  "workStartTime" : "string"
}

```

**HTTP**    示例

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

修改作 信息

PUT /work/update

明

修改作 信息接口

参数

型	名称	明	型
Query	city 可	市	string
Query	county 可		string
Query	machinery.ma chineryBrand 可	机械品牌	string
Query	machinery.ma chineryLicens e 可	机械 牌号	string
Query	machinery.ma chineryNO 可	机械 号	string
Query	machinery.ma chineryOwner 可	机械所有者ID	integer (int32)
Query	machinery.ma chineryStatus 可	机状 1正常 2 坏 3 修 4 除	integer (int32)

型	名称	明	型
Query	<b>machinery.machineryWidth</b> 可	机械 度	number (double)
Query	<b>machinery.user.city</b> 可	市	string
Query	<b>machinery.user.cooperative</b> 可	合作社	string
Query	<b>machinery.user.county</b> 可		string
Query	<b>machinery.user.createDate</b> 可	建	string (date-time)
Query	<b>machinery.user.creator</b> 可	建人	integer (int32)
Query	<b>machinery.user.email</b> 可	箱	string
Query	<b>machinery.user.lastLoginTime</b> 可	上一次登	string (date-time)
Query	<b>machinery.user.passWord</b> 可	密	string
Query	<b>machinery.user.phone</b> 可		string
Query	<b>machinery.user.province</b> 可	省	string
Query	<b>machinery.user.status</b> 可	用 状 1正常 -1禁用	integer (int32)



型	名称	明	型
Query	<b>machinery.user.town</b> 可		string
Query	<b>machinery.user.type</b> 可	人事 型	integer (int32)
Query	<b>machinery.user.updateDate</b> 可	更新	string (date-time)
Query	<b>machinery.user.updator</b> 可	更新人	integer (int32)
Query	<b>machinery.user.userName</b> 可	用 名	string
Query	<b>machinery.user.village</b> 可	村	string
Query	<b>province</b> 可	省	string
Query	<b>town</b> 可		string
Query	<b>verificationResults</b> 可	核 果	integer (int32)
Query	<b>village</b> 可	村	string
Query	<b>workArea</b> 可	作 面	number (double)
Query	<b>workDepth</b> 可	作 深度	number (double)
Query	<b>workEndMap</b> 可	作 束坐	string
Query	<b>workEndTime</b> 可	作 束	string (date-time)
Query	<b>workLength</b> 可	作 里程	number (double)

型	名称	明	型
Query	<b>workMachine ryID</b> 可	作 的机械ID	integer (int32)
Query	<b>workStartMap</b> 可	作 始坐	string
Query	<b>workStartTim e</b> 可	作 始	string (date-time)
Body	<b>work</b> 必填	作 信息 体	<a href="#">com.puyan.shengren. agricultural.Enity.Wo rk</a>

HTTP代	明	型
200	OK	<a href="#">Result</a>
201	Created	无内容
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- [application/json](#)

生成

- [/](#)

HTTP 求示例

求 path

/work/update

求 query

```

{
  "city" : "string",
  "county" : "string",
  "machinery.machineryBrand" : "string",
  "machinery.machineryLicense" : "string",
  "machinery.machineryNO" : "string",
  "machinery.machineryOwner" : 0,
  "machinery.machineryStatus" : 0,
  "machinery.machineryWidth" : 0.0,
  "machinery.user.city" : "string",
  "machinery.user.cooperative" : "string",
  "machinery.user.county" : "string",
  "machinery.user.createDate" : "string",
  "machinery.user.creator" : 0,
  "machinery.user.email" : "string",
  "machinery.user.lastLoginTime" : "string",
  "machinery.user.passWord" : "string",
  "machinery.user.phone" : "string",
  "machinery.user.province" : "string",
  "machinery.user.status" : 0,
  "machinery.user.town" : "string",
  "machinery.user.type" : 0,
  "machinery.user.updateDate" : "string",
  "machinery.user.updator" : 0,
  "machinery.user.userName" : "string",
  "machinery.user.village" : "string",
  "province" : "string",
  "town" : "string",
  "verificationResults" : 0,
  "village" : "string",
  "workArea" : 0.0,
  "workDepth" : 0.0,
  "workEndMap" : "string",
  "workEndTime" : "string",
  "workLength" : 0.0,
  "workMachineryID" : 0,
  "workStartMap" : "string",
  "workStartTime" : "string"
}

```

## HTTP 示例

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

上传图片文件,支持多张图片

POST /work/uploadImgs

参数

型	名称	说明	类型
Query	workID 必填	需要属于那一条的ID	ref
Body	file 必填	文件流对象,接收数据格式	< MultipartFile > array

HTTP代码	说明	类型
200	OK	Result
201	Created	无内容
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- application/json

生成

- /

HTTP 请求示例

请求 path

/work/uploadImgs

求 query

```
{
  "workID" : "ref"
}
```

求 body

```
[ null ]
```

HTTP 示例

200

```
{
  "code" : 0,
  "data" : "object",
  "msg" : "string"
}
```

入作 信息

POST /work/work\_insert

明

入作 信息接口

参数

型	名称	明	型
Query	city 可	市	string
Query	county 可		string
Query	machinery.ma chineryBrand 可	机械品牌	string

型	名称	明	型
Query	<b>machinery.machineryLicense</b> 可	机械 牌号	string
Query	<b>machinery.machineryNO</b> 可	机械 号	string
Query	<b>machinery.machineryOwner</b> 可	机械所有者ID	integer (int32)
Query	<b>machinery.machineryStatus</b> 可	机状 1正常 2 坏 3 修 4 除	integer (int32)
Query	<b>machinery.machineryWidth</b> 可	机械 度	number (double)
Query	<b>machinery.user.city</b> 可	市	string
Query	<b>machinery.user.cooperative</b> 可	合作社	string
Query	<b>machinery.user.county</b> 可		string
Query	<b>machinery.user.createDate</b> 可	建	string (date-time)
Query	<b>machinery.user.creator</b> 可	建人	integer (int32)
Query	<b>machinery.user.email</b> 可	箱	string
Query	<b>machinery.user.lastLoginTime</b> 可	上一次登	string (date-time)

型	名称	明	型
Query	<b>machinery.user.passWord</b> 可	密	string
Query	<b>machinery.user.phone</b> 可		string
Query	<b>machinery.user.province</b> 可	省	string
Query	<b>machinery.user.status</b> 可	用 状 1正常 -1禁用	integer (int32)
Query	<b>machinery.user.town</b> 可		string
Query	<b>machinery.user.type</b> 可	人事 型	integer (int32)
Query	<b>machinery.user.updateDate</b> 可	更新	string (date-time)
Query	<b>machinery.user.updator</b> 可	更新人	integer (int32)
Query	<b>machinery.user.userName</b> 可	用 名	string
Query	<b>machinery.user.village</b> 可	村	string
Query	<b>province</b> 可	省	string
Query	<b>town</b> 可		string
Query	<b>verificationResults</b> 可	核 果	integer (int32)
Query	<b>village</b> 可	村	string

型	名称	明	型
Query	<b>workArea</b> 可	作 面	number (double)
Query	<b>workDepth</b> 可	作 深度	number (double)
Query	<b>workEndMap</b> 可	作 束坐	string
Query	<b>workEndTime</b> 可	作 束	string (date-time)
Query	<b>workLength</b> 可	作 里程	number (double)
Query	<b>workMachine</b> <b>ryID</b> 可	作 的机械ID	integer (int32)
Query	<b>workStartMap</b> 可	作 始坐	string
Query	<b>workStartTime</b> <b>e</b> 可	作 始	string (date-time)
Body	<b>work</b> 必填	作 信息 体	<a href="#">com.puyan.shengren.agricultural.Enity.Work</a>

HTTP代	明	型
200	OK	<a href="#">Result</a>
201	Created	无内容
401	Unauthorized	无内容
403	Forbidden	无内容
404	Not Found	无内容

消耗

- [application/json](#)

生成

- [/](#)



## HTTP 求示例

求 **path**

```
/work/work_insert
```

求 **query**

```

{
  "city" : "string",
  "county" : "string",
  "machinery.machineryBrand" : "string",
  "machinery.machineryLicense" : "string",
  "machinery.machineryNO" : "string",
  "machinery.machineryOwner" : 0,
  "machinery.machineryStatus" : 0,
  "machinery.machineryWidth" : 0.0,
  "machinery.user.city" : "string",
  "machinery.user.cooperative" : "string",
  "machinery.user.county" : "string",
  "machinery.user.createDate" : "string",
  "machinery.user.creator" : 0,
  "machinery.user.email" : "string",
  "machinery.user.lastLoginTime" : "string",
  "machinery.user.passWord" : "string",
  "machinery.user.phone" : "string",
  "machinery.user.province" : "string",
  "machinery.user.status" : 0,
  "machinery.user.town" : "string",
  "machinery.user.type" : 0,
  "machinery.user.updateDate" : "string",
  "machinery.user.updator" : 0,
  "machinery.user.userName" : "string",
  "machinery.user.village" : "string",
  "province" : "string",
  "town" : "string",
  "verificationResults" : 0,
  "village" : "string",
  "workArea" : 0.0,
  "workDepth" : 0.0,
  "workEndMap" : "string",
  "workEndTime" : "string",
  "workLength" : 0.0,
  "workMachineryID" : 0,
  "workStartMap" : "string",
  "workStartTime" : "string"
}

```

## HTTP 示例

200

```
{  
  "code" : 0,  
  "data" : "object",  
  "msg" : "string"  
}
```