**阡陌——详细设计规约**

|  |  |  |  |
| --- | --- | --- | --- |
| 修改日期 | 版本 | 说明 | 参与人员 |
| 2023.12.26 | 1.0 | 添加详细接口设计 | 刘天琦，陆济贤，王加炜，杨滕超 |
| 2024.1.12 | 1.1 | 修改活动图 | 刘天琦，陆济贤，王加炜，杨滕超 |

|  |  |  |  |
| --- | --- | --- | --- |
| 评审日期 | 参与评审人员 | 评审后修改批准日期 | 参与人员 |
| 2023.12.27 | 刘天琦，陆济贤，王加炜，杨滕超 | 2024.1.12 | 刘天琦，陆济贤，王加炜，杨滕超 |

1. **引言**

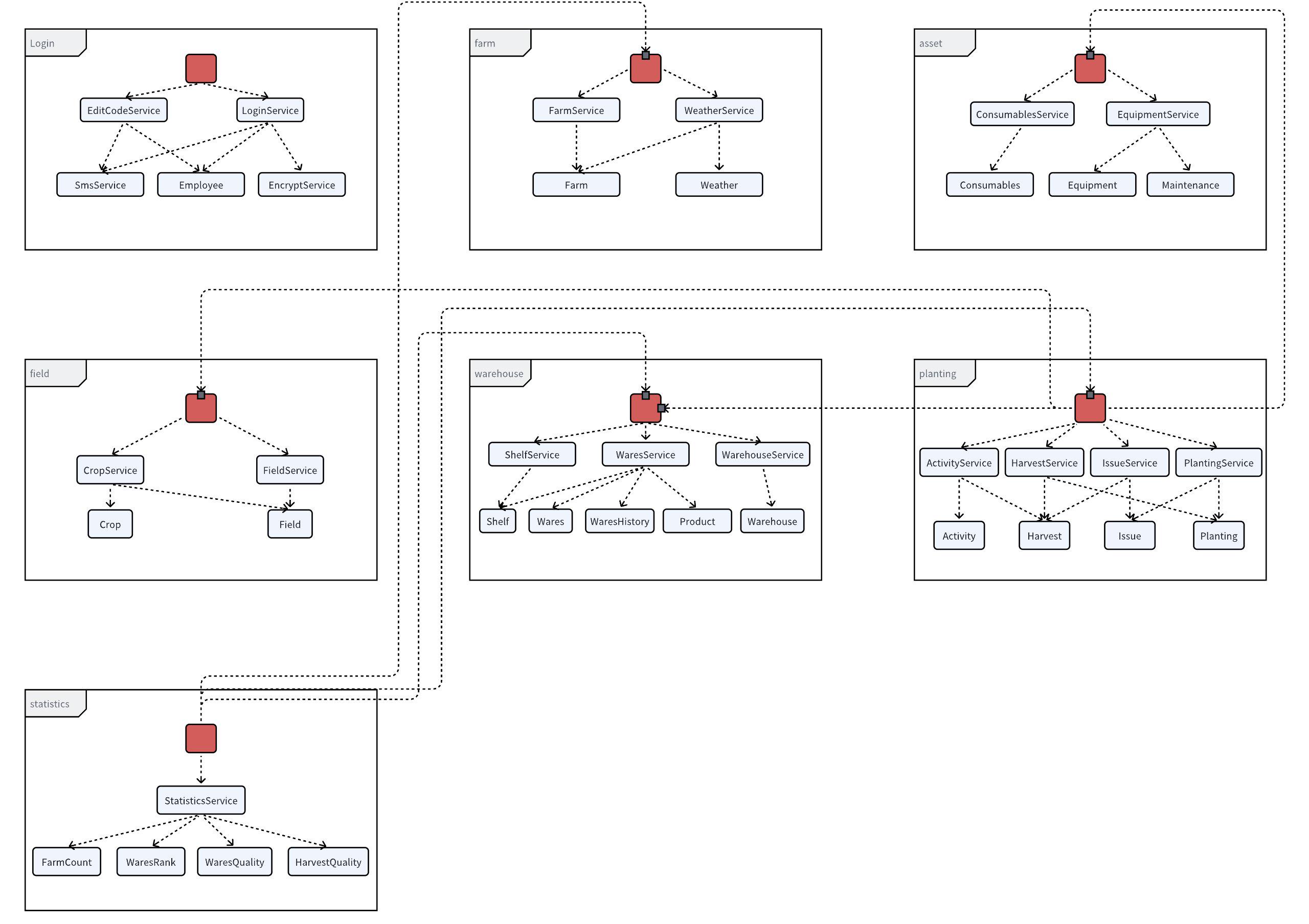
**1.1 详细设计依据**

* 阡陌——需求分析规约文档
* 阡陌——需求文档
* 阡陌——概要设计文档
* 杜庆峰老师上课所讲的详细设计基本原则

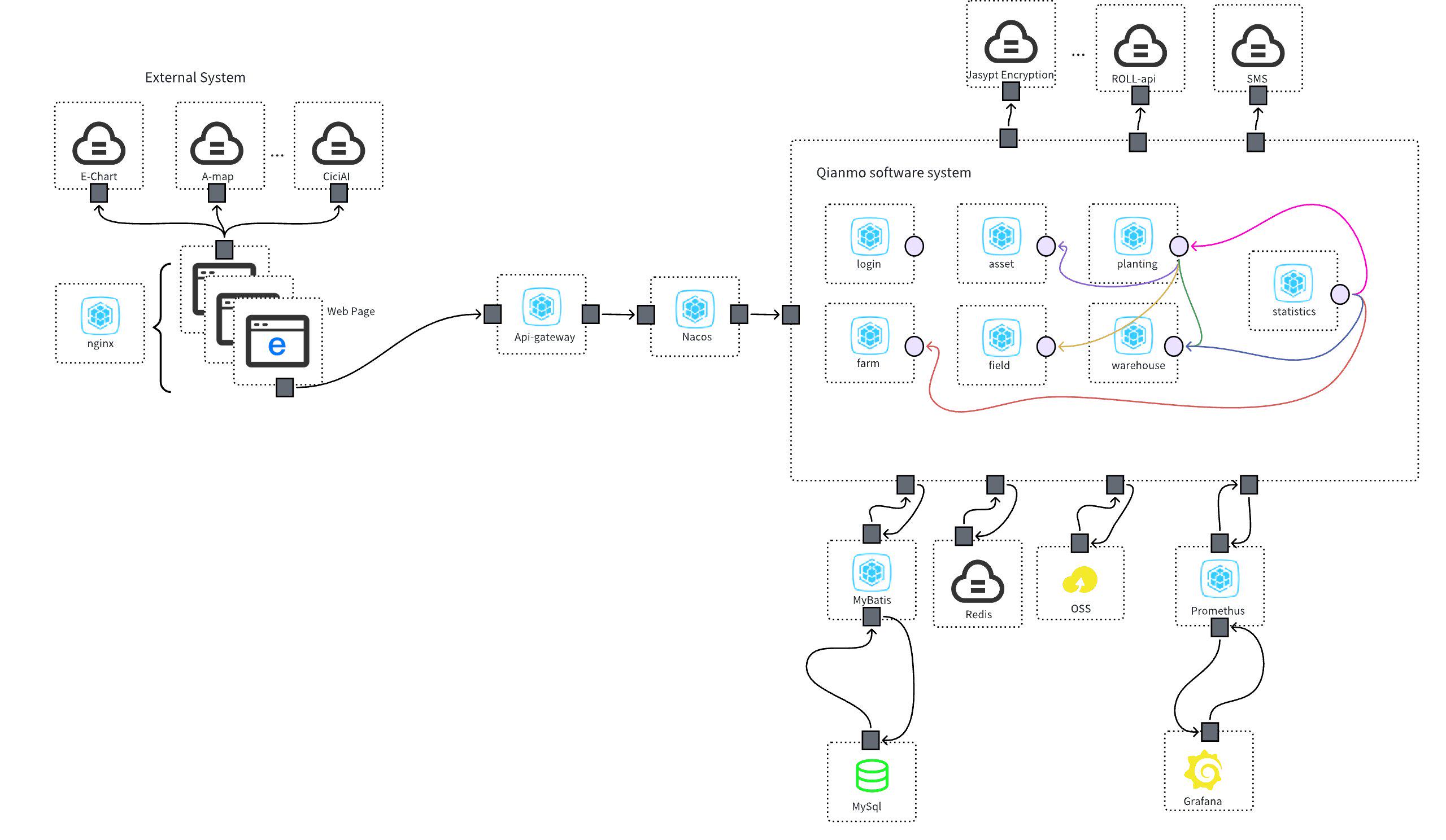
**1.2 参考资料**

* 《Spring Boot与微服务实战》
* 《Mastering Spring Boot 2.0》
* 《Design Patterns: Elements of Reusable Object-Oriented Software》
* 《Domain-Driven Design: Tackling Complexity in the Heart of Software》
* 《Software Systems Architecture: Working With Stakeholders Using Viewpoints and Perspectives》

2. **系统软件结构设计**



3. **微服务设计**



4. **类设计**

4.1 **资产微服务**

4.1.1 **Consumables类**

4.1.1.1 **功能**

主要用于追踪农场中的消耗品，如种子、化肥和农药。它包含了关于消耗品的信息，以便进行管理和追踪。

4.1.1.2 **性能**

支持并发搜索，保证每个方法的执行时间在100ms以内。

4.1.1.3 **输入项**

获取消耗物时候是对应农场id，编辑消耗物信息时是编辑后的消耗物信息，添加消耗物时是需要添加消耗物的信息。

4.1.1.4 **输出项**

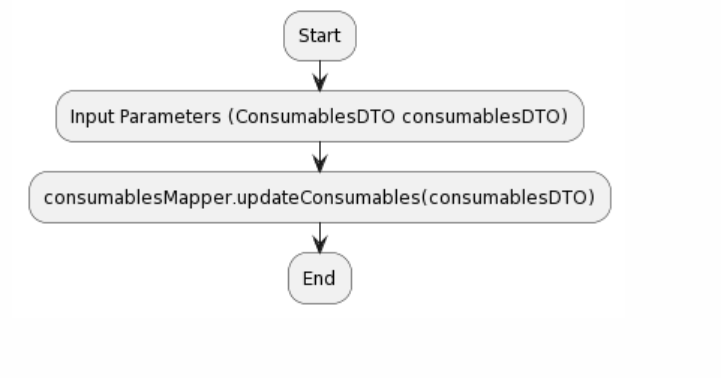
获取消耗物时是对应农场id的农场下所有消耗物。

4.1.1.5 **属性描述**

|  |  |  |
| --- | --- | --- |
| 字段名 | 数据类型 | 说明 |
| consumables\_id | int | 消耗品id |
| name | varchar | 消耗品名称 |
| type | varchar | 消耗品类型 |
| stock | int | 余量 |
| unit | varchar | 单位 |
| farm\_id | int | 农场id |

4.1.1.6 **方法详细描述**

4.1.1.6.1 **editConsumables**



**接口地址**

**POST** /api/consumables/consumables\_edit/{consumablesId}

**接口描述**

修改对应消耗物id的消耗物信息

**请求数据类型**

application/x-www-form-urlencoded,application/json

**响应数据类型**

*/*

**请求示例**

{

"consumablesId": 0,

"name": "",

"stock": 0,

"unit": ""

}

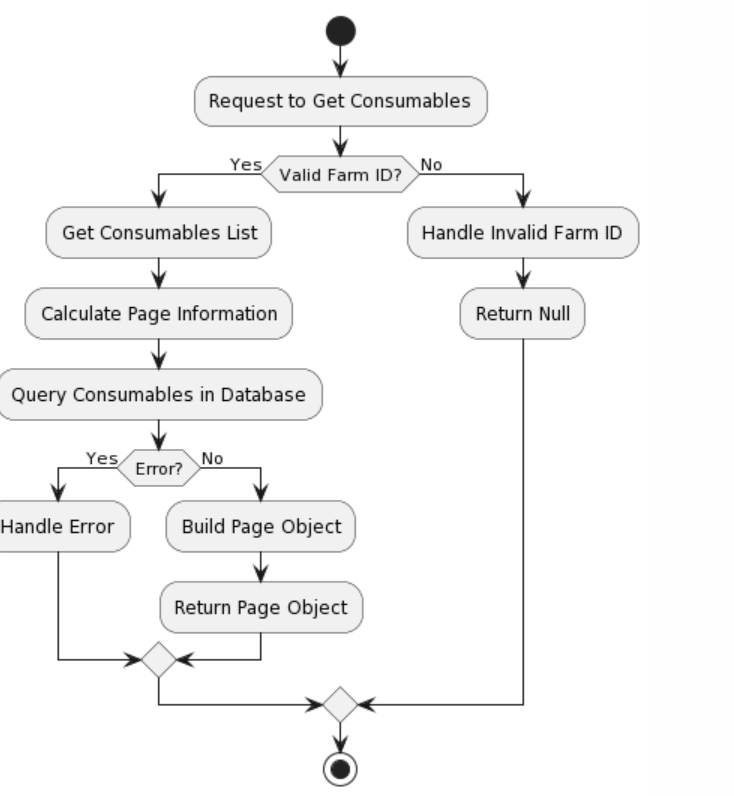
**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| consumablesId |  | path | true | integer(int32) |  |
| consumablesDTO | ConsumablesDTO | body | true | ConsumablesDTO | ConsumablesDTO |
| consumablesId |  |  | false | integer(int32) |  |
| name |  |  | false | string |  |
| stock |  |  | false | integer(int32) |  |
| unit |  |  | false | string |  |

**响应参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名称** | **参数说明** | **类型** | **schema** |
| ResponseEntity<?> |  |  |  |

4.1.1.6.2 **getConsumables**



**接口地址**

**GET** /api/consumables/farm/{farmId}

**接口描述**

获取对应所有农场id的消耗物。

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

*/*

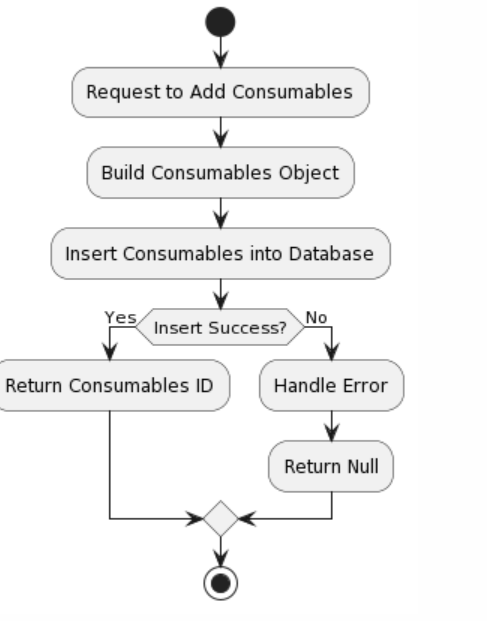
**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| farmId |  | path | true | integer(int32) |  |
| page |  | query | false | integer(int32) |  |
| size |  | query | false | integer(int32) |  |

**响应参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名称** | **参数说明** | **类型** | **schema** |
| ResponseEntity<?> |  |  |  |

4.1.1.6.3 **addConsumables**



**接口地址**

**POST** /api/consumables/farm/{farmId}

**接口描述**

为对应农场id的农场添加消耗物

**请求数据类型**

application/x-www-form-urlencoded,application/json

**响应数据类型**

*/*

**请求示例**

{

"name": "",

"type": "",

"stock": 0,

"unit": "",

"farmId": 0

}

**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| farmId |  | path | true | integer(int32) |  |
| consumablesRequest | ConsumablesRequest | body | true | ConsumablesRequest | ConsumablesRequest |
| name |  |  | false | string |  |
| type |  |  | false | string |  |
| stock |  |  | false | integer(int32) |  |
| unit |  |  | false | string |  |
| farmId |  |  | false | integer(int32) |  |

**响应参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名称** | **参数说明** | **类型** | **schema** |
| ResponseEntity<?> |  |  |  |

4.1.2 **Equipment类**

4.1.2.1 **功能**

主要追踪农场中的各类设备，记录设备的各类信息，是否空闲等。

4.1.2.2 **性能**

4.1.2.3 **输入项**

在添加设备时是需要添加设备的所有信息，废弃设备，空闲设备，使用设备时候是设备的id，获取设备时是对应农场id。

4.1.2.4 **输出项**

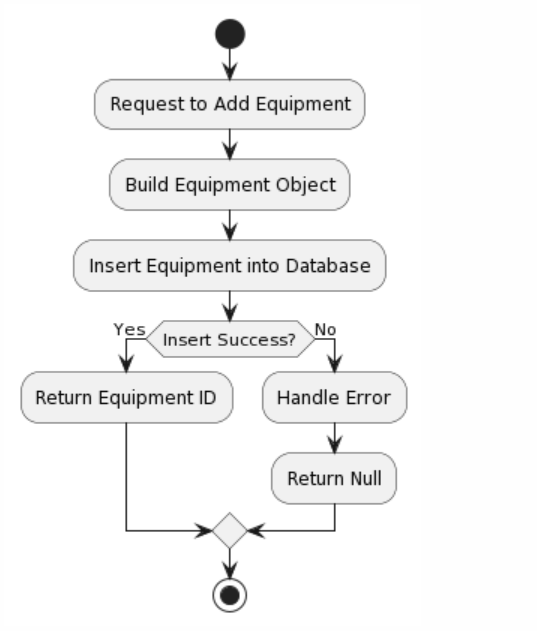
获取设备时是对应农场的所有设备的信息。

4.1.2.5 **属性描述**

|  |  |  |
| --- | --- | --- |
| 字段名 | 数据类型 | 说明 |
| equipmentId | int | 设备id |
| name | varchar | 设备名称 |
| type | varchar | 设备类型 |
| status | int | 设备状态 |
| farm\_id | int | 农场id |
| startDate | LocalDateTime | 设备使用开始日期 |
| endDate | LocalDateTime | 设备结束使用日期 |

4.1.2.6 **方法详细描述**

4.1.2.6.1 **addEquipment**



**接口地址**

**POST** /api/equipment/add\_equipment/farm/{farmId}

**接口描述**

给对应农场id的农场添加设备

**请求数据类型**

application/x-www-form-urlencoded,application/json

**响应数据类型**

*/*

**请求示例**

{

"name": "",

"type": "",

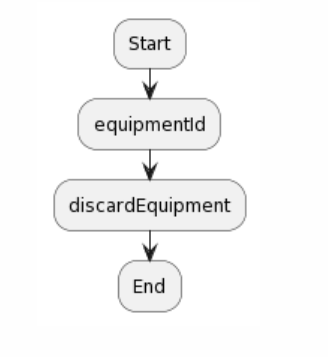
"farmId": 0

}

**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| farmId |  | path | true | integer(int32) |  |
| equipmentRequest | EquipmentRequest | body | true | EquipmentRequest | EquipmentRequest |
| name |  |  | false | string |  |
| type |  |  | false | string |  |
| farmId |  |  | false | integer(int32) |  |

4.1.2.6.2 **discardEquipment**



**接口地址**

**POST** /api/equipment/discard/{equipmentId}

**接口描述**

废弃对应equipmentId的设备。

**请求数据类型**

application/x-www-form-urlencoded

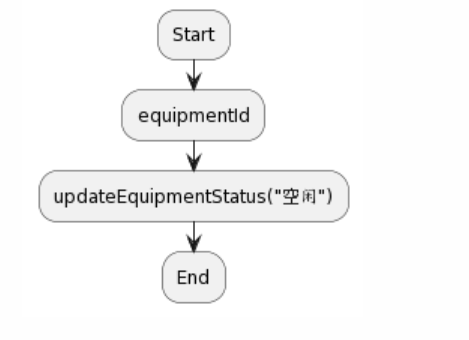
**响应数据类型**

*/*

**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| equipmentId |  | path | true | integer(int32) |  |

4.1.2.6.3 **freeEquipment**



**接口地址**

**POST** /api/equipment/free/{equipmentId}

**接口描述**

使对应设备Id的设备空闲。

**请求数据类型**

application/x-www-form-urlencoded

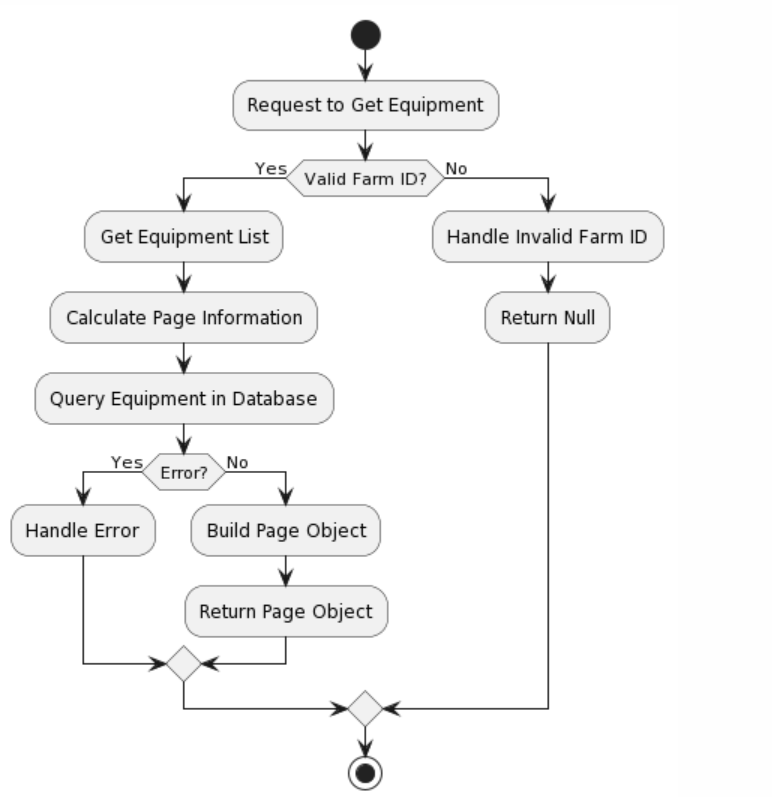
**响应数据类型**

*/*

**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| equipmentId |  | path | true | integer(int32) |  |

4.1.2.6.4 **getEquipment**



**接口地址**

**GET** /api/equipment/get\_info/farm/{farmId}

**接口描述**

获取指定农场id对应农场的所有设备

**请求数据类型**

application/x-www-form-urlencoded

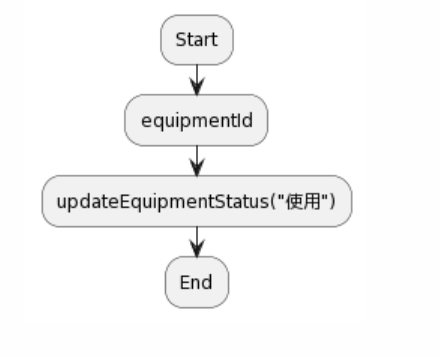
**响应数据类型**

*/*

**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| farmId |  | path | true | integer(int32) |  |
| page |  | query | false | integer(int32) |  |
| size |  | query | false | integer(int32) |  |

4.1.2.6.5 **useEquipment**



**接口地址**

**POST** /api/equipment/use/{equipmentId}

**接口描述**

使用对应设备Id的设备

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

*/*

**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| equipmentId |  | path | true | integer(int32) |  |

4.1.3 **Maintenance类**

4.1.3.1 **功能**

主要记录设备维护的历史信息。

4.1.3.2 **性能**

4.1.3.3 **输入项**

添加维护记录时是该条维护的所有信息，获取设备维护信息时是对应设备id。

4.1.3.4 **输出项**

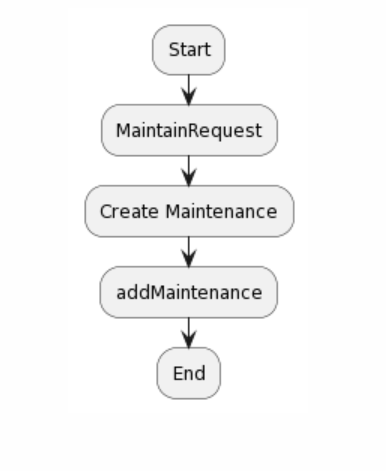
获取设备维护信息时是对应设备的所有维护信息。

4.1.3.5 **属性描述**

|  |  |  |
| --- | --- | --- |
| 字段名 | 数据类型 | 说明 |
| maintenance\_id | int | 维护id |
| employee\_id | int | 后勤部员工id |
| equipment\_id | int | 设备id |
| start\_date | date | 开始日期 |
| end\_date | date | 结束日期 |
| detail | varchar | 描述 |
| cost | decimal | 花费 |

4.1.3.6 **方法详细描述**

4.1.3.6.1 **addMaintenance**



**接口地址**

**POST** /api/equipment/add\_maintenance

**接口描述**

为指定的设备进行维护。

**请求数据类型**

application/x-www-form-urlencoded,application/json

**响应数据类型**

*/*

**请求示例**

{

"employeeId": 0,

"equipmentId": 0,

"detail": "",

"cost": 0,

"startDate": ""

}

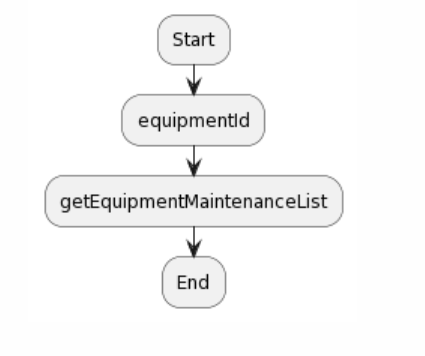
**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| maintainRequest | MaintainRequest | body | true | MaintainRequest | MaintainRequest |
| employeeId |  |  | false | integer(int32) |  |
| equipmentId |  |  | false | integer(int32) |  |
| detail |  |  | false | string |  |
| cost |  |  | false | number(double) |  |
| startDate |  |  | false | string(date-time) |  |

**响应参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名称** | **参数说明** | **类型** | **schema** |
| ResponseEntity<?> |  |  |  |

4.1.3.6.2 **getEquipmentMaintenance**



**接口地址**

**GET** /api/equipment/maintenance/{equipmentId}

**接口描述**

获取指定equipmentId对应所有设备的所有维护记录

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

*/*

**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| equipmentId |  | path | true | integer(int32) |  |

**响应参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名称** | **参数说明** | **类型** | **schema** |
| ResponseEntity<?> |  |  |  |

4.2 **农场微服务**

4.2.1 **Farm**

4.2.1.1 **功能**

记录农场的各项信息，比如名称、位置等。

4.2.1.2 **性能**

4.2.1.3 **输入项**

农场名称，添加或更新农场时是需要添加农场的各项信息。

4.2.1.4 **输出项**

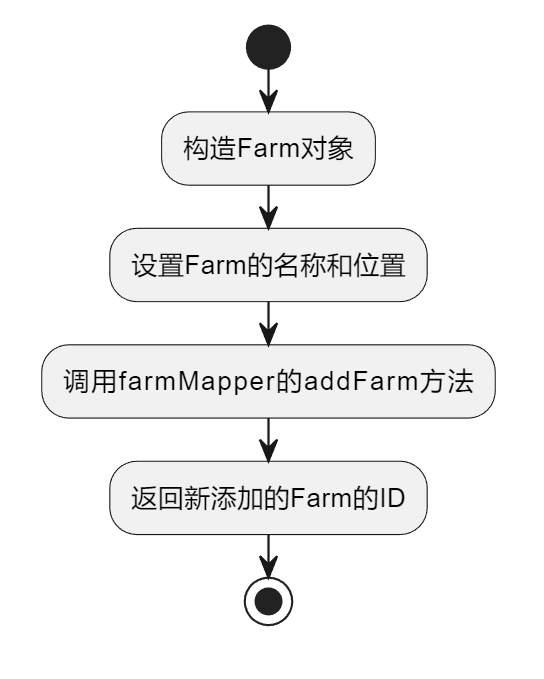
获取农场时，输出所有农场以及其对应信息，通过名称获取农场Id时是农场Id。

4.2.1.5 **属性描述**

|  |  |  |
| --- | --- | --- |
| 字段名 | 数据类型 | 说明 |
| farm\_id | int | 农场ID |
| name | varchar | 农场名称 |
| location | varchar | 位置 |

4.2.1.6 **方法详细描述**

4.2.1.6.1 **addFarm**



**接口地址**

**POST** /api/farm/addFarm

**接口描述**

新建农场

**请求数据类型**

application/x-www-form-urlencoded,application/json

**响应数据类型**

*/*

**请求示例**

{

"name": "",

"location": ""

}

**请求参数**

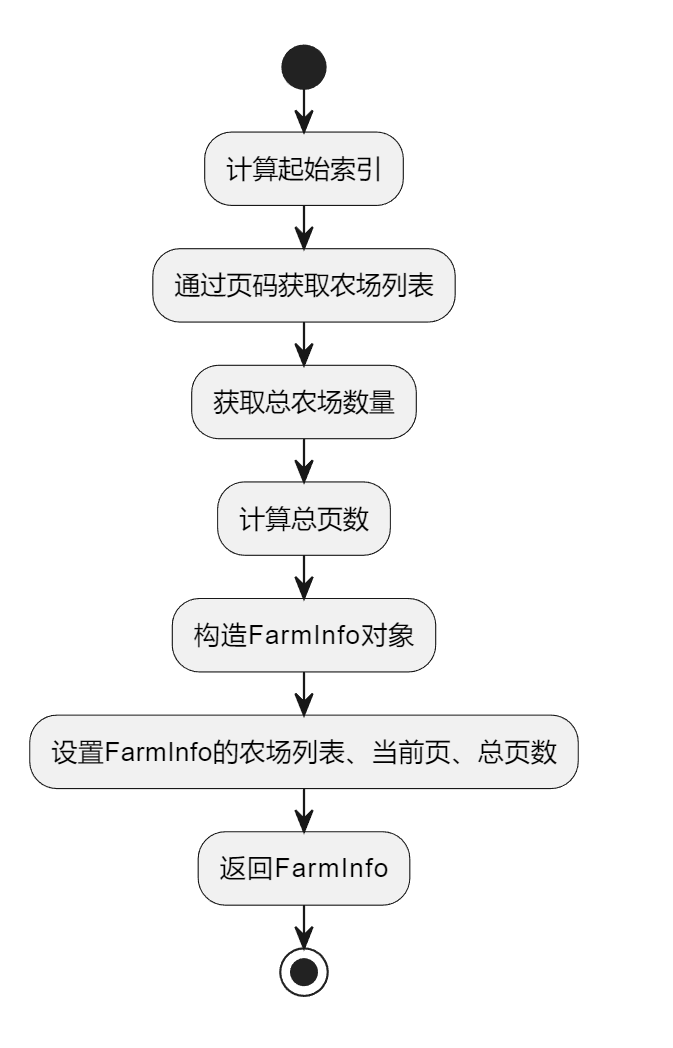
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| farmRequest | FarmRequest | body | true | FarmRequest | FarmRequest |
| name |  |  | false | string |  |
| location |  |  | false | string |  |

**响应参数**



**点击图片可查看完整电子表格**

4.2.1.6.2 **getFarm**



**接口地址**

**GET** /api/farm/getFarm/{page}

**接口描述**

获取所有农场名称

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

*/*

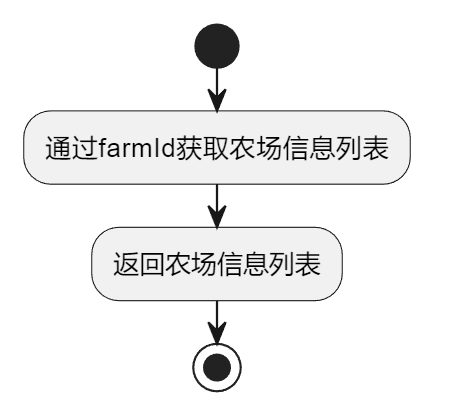
**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| page |  | path | true | integer(int32) |  |

**响应参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名称** | **参数说明** | **类型** | **schema** |
| ResponseEntity<?> |  |  |  |

4.2.1.6.3 **getFarmInfo**



**接口地址**

**GET** /api/farm/getFarmInfo/{farmId}

**接口描述**

获取农场名称以及其位置。

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

*/*

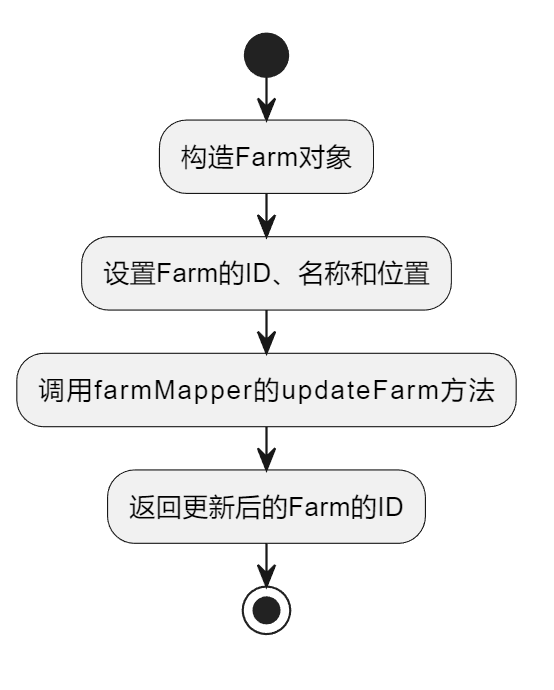
**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| farmId |  | path | true | integer(int32) |  |

**响应参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名称** | **参数说明** | **类型** | **schema** |
| farmId |  | integer(int32) | integer(int32) |
| name |  | string |  |
| location |  | string |  |

4.2.1.6.4 **updateFarm**



**接口地址**

**PUT** /api/farm/updateFarm

**接口描述**

更新农场信息

**请求数据类型**

application/x-www-form-urlencoded,application/json

**响应数据类型**

*/*

**请求示例**

{

"farm\_id": 0,

"name": "",

"location": ""

}

**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| farmDTO | FarmDTO | body | true | FarmDTO | FarmDTO |
| farm\_id |  |  | false | integer(int32) |  |
| name |  |  | false | string |  |
| location |  |  | false | string |  |

**响应参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名称** | **参数说明** | **类型** | **schema** |
| ResponseEntity<?> |  |  |  |

4.2.2 **Weather**

4.2.2.1 **功能**

记录对应农场的天气。

4.2.2.2 **性能**

4.2.2.3 **输入项**

对应农场的Id。

4.2.2.4 **输出项**

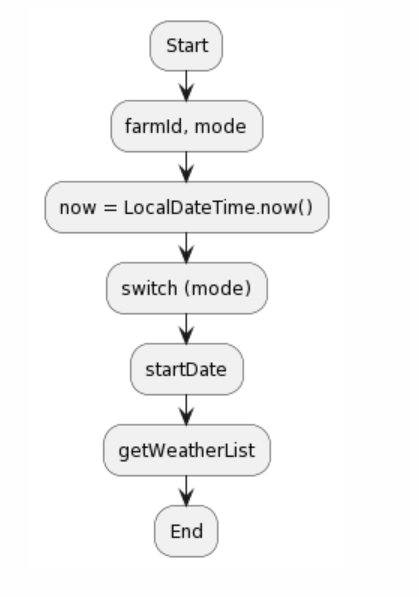
Id对应农场的天气信息。

4.2.2.5 **属性描述**

|  |  |  |
| --- | --- | --- |
| 字段名 | 数据类型 | 说明 |
| farm\_id | int | 农场id |
| record\_date | date | 记录日期 |
| high\_temp | int | 最高气温 |
| low\_temp | int | 最低气温 |
| precipitation | decimal | 降水量 |
| humidity | int | 湿度 |
| wind | int | 风速 |
| weather | varchar | 天气 |

4.2.2.6 **方法详细描述**

4.2.2.6.1 **getWeather**



**接口地址**

**GET** /api/weather/farm/{farmId}/{mode}

**接口描述**

获取农场id对应农场的天气。

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

*/*

**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| farmId |  | path | true | integer(int32) |  |
| mode |  | path | true | integer(int32) |  |

**响应参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名称** | **参数说明** | **类型** | **schema** |
| ResponseEntity<?> |  |  |  |

4.3 **田地微服务**

4.3.1 **Crop类**

4.3.1.1 **功能**

记录作物的各类信息，包括适宜生长的土壤类型，酸碱度等。

4.3.1.2 **性能**

4.3.1.3 **输入项**

添加作物时输入作物的各项信息。

4.3.1.4 **输出项**

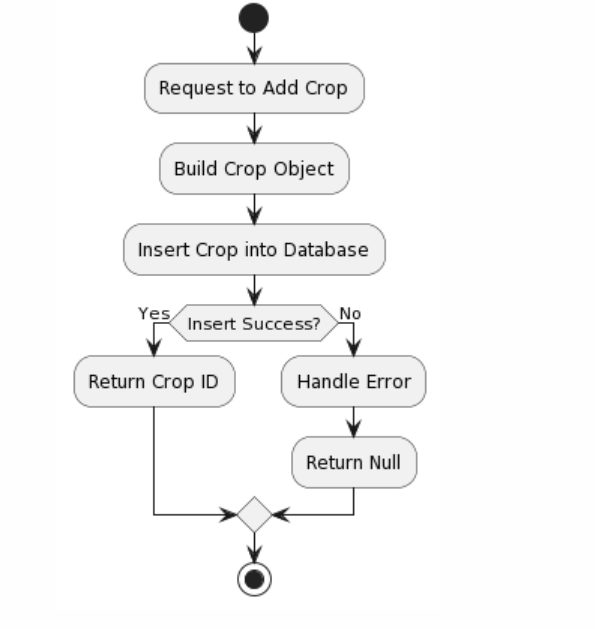
该次种植的作物或者该田地的作物历史记录。

4.3.1.5 **属性描述**

|  |  |  |
| --- | --- | --- |
| 字段名 | 数据类型 | 说明 |
| crop\_id | int | 作物id |
| name | varchar | 作物名称 |
| crop\_type | varchar | 作物类型 |
| soil\_type | varchar | 适宜土壤类型 |
| high\_ph | int | 适宜的最高ph |
| low\_ph | int | 适宜的最低ph |

4.3.1.6 **方法详细描述**

4.3.1.6.1 **addCrop**



**接口地址**

**POST** /api/crop

**接口描述**

新建作物，填写作物各种信息。

**请求数据类型**

application/x-www-form-urlencoded,application/json

**响应数据类型**

*/*

**请求示例**

{

"name": "",

"cropType": "",

"soilType": "",

"lowPh": 0,

"highPh": 0

}

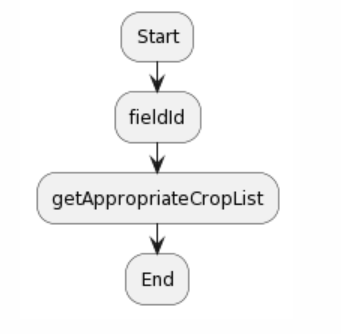
**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| cropRequest | CropRequest | body | true | CropRequest | CropRequest |
| name |  |  | false | string |  |
| cropType |  |  | false | string |  |
| soilType |  |  | false | string |  |
| lowPh |  |  | false | number(double) |  |
| highPh |  |  | false | number(double) |  |

**响应参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名称** | **参数说明** | **类型** | **schema** |
| ResponseEntity<?> |  |  |  |

4.3.1.6.2 **getApproriateCrop**



**接口地址**

**GET** /api/crop/appropriate/field/{fieldId}

**接口描述**

通过对于田地各项属性进行分析，返回对应最适宜种植的作物。

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

*/*

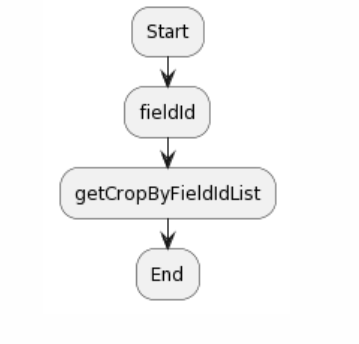
**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| fieldId |  | path | true | integer(int32) |  |

**响应参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名称** | **参数说明** | **类型** | **schema** |
| cropId |  | integer(int32) | integer(int32) |
| name |  | string |  |
| cropType |  | string |  |
| soilType |  | string |  |
| highPh |  | number(double) | number(double) |
| lowPh |  | number(double) | number(double) |

4.3.1.6.3 **getCropByFieldId**



**接口地址**

**GET** /api/crop/field/{fieldId}

**接口描述**

获取当前田地上种植的作物。

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

*/*

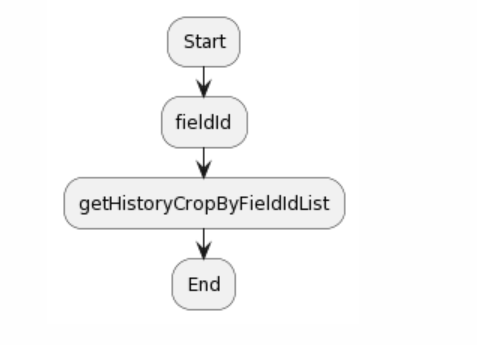
**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| fieldId |  | path | true | integer(int32) |  |

**响应参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名称** | **参数说明** | **类型** | **schema** |
| cropId |  | integer(int32) | integer(int32) |
| name |  | string |  |
| cropType |  | string |  |
| soilType |  | string |  |
| highPh |  | number(double) | number(double) |
| lowPh |  | number(double) | number(double) |

4.3.1.6.4 **getHistoryCropByFieldId**



**接口地址**

**GET** /api/crop/history/field/{fieldId}

**接口描述**

通过田地id获取当前田地历史种植的作物

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

*/*

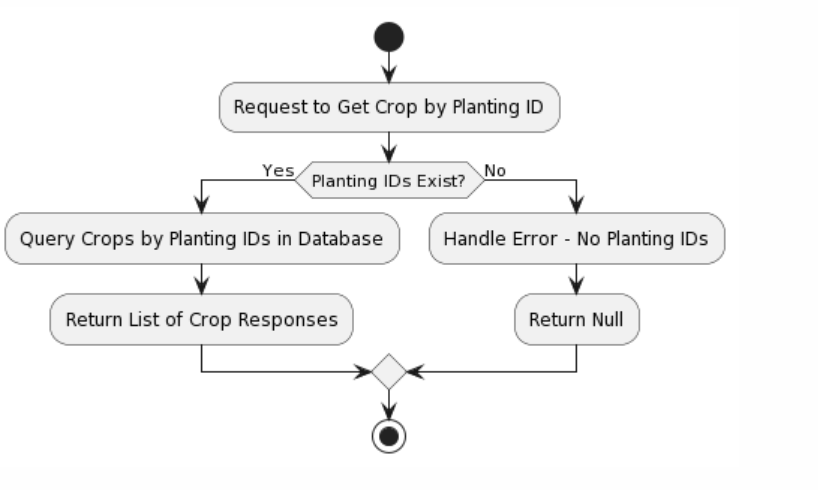
**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| fieldId |  | path | true | integer(int32) |  |

**响应参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名称** | **参数说明** | **类型** | **schema** |
| cropType |  | string |  |
| startDate |  | string(date-time) | string(date-time) |
| endDate |  | string(date-time) | string(date-time) |
| employeeName |  | string |  |

4.3.1.6.5 **getCropByPlantingId**



**接口地址**

**GET** /api/crop/name/planting

**接口描述**

通过种植id获取种植所对应的作物。

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

*/*

**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| plantingIds |  | query | true | array | integer |

**响应参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名称** | **参数说明** | **类型** | **schema** |
| cropId |  | integer(int32) | integer(int32) |
| name |  | string |  |
| plantingId |  | integer(int32) | integer(int32) |

4.3.2 **Field类**

4.3.2.1 **功能**

记录田地酸碱度，位置等等一系列属性。

4.3.2.2 **性能**

4.3.2.3 **输入项**

更新田地，筛选田地或新建田地需要田地所有属性。

4.3.2.4 **输出项**

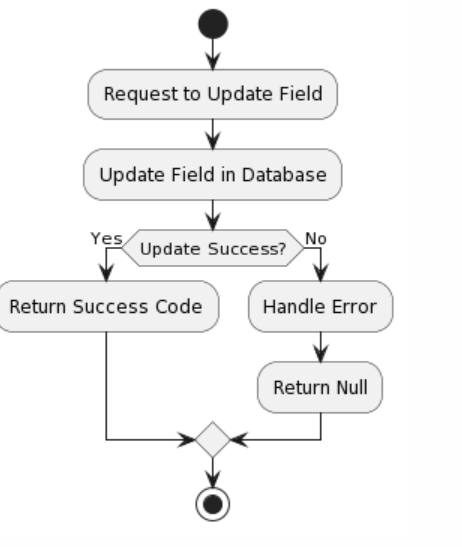
田地的所有属性

4.3.2.5 **属性描述**

|  |  |  |
| --- | --- | --- |
| 字段名 | 数据类型 | 说明 |
| field\_id | int | 田地id |
| status | varchar | 田地状态 |
| name | char | 田地名称 |
| farm\_id | int | 农场id |
| soil\_type | varchar | 土壤种类 |
| ph\_level | decimal | 酸碱度 |
| vertex1\_latitude | decimal | 第一个顶点坐标纬度 |
| vertex1\_longitude | decimal | 第一个顶点坐标经度 |
| vertex2\_latitude | decimal | 第二个顶点坐标纬度 |
| vertex2\_longitude | decimal | 第二个顶点坐标经度 |
| vertex3\_latitude | decimal | 第三个顶点坐标纬度 |
| vertex3\_longitude | decimal | 第三个顶点坐标经度 |
| vertex4\_latitude | decimal | 第四个顶点坐标纬度 |
| vertex4\_longitude | decimal | 第四个顶点坐标经度 |
| photo | string | 农田照片 |

4.3.2.6 **方法详细描述**

4.3.2.6.1 **updateField**



**接口地址**

**PUT** /api/field

**接口描述**

更新田地属性。

**请求数据类型**

application/x-www-form-urlencoded,application/json

**响应数据类型**

*/*

**请求示例**

{

"fieldId": 0,

"status": "",

"name": "",

"farmId": 0,

"soilType": "",

"phLevel": 0,

"vertex1Latitude": 0,

"vertex1Longitude": 0,

"vertex2Latitude": 0,

"vertex2Longitude": 0,

"vertex3Latitude": 0,

"vertex3Longitude": 0,

"vertex4Latitude": 0,

"vertex4Longitude": 0,

"photo": ""

}

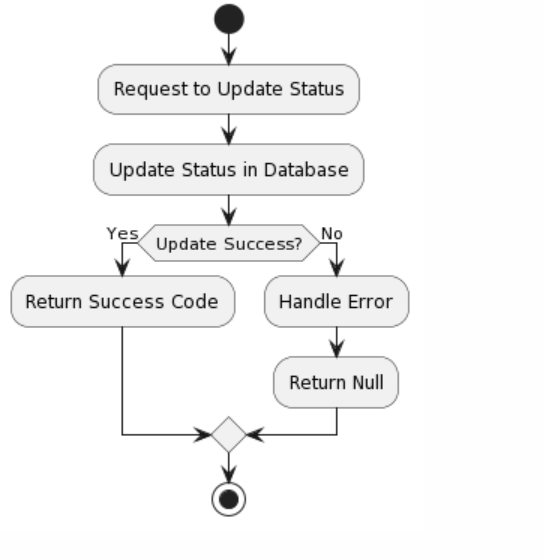
**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| field | Field | body | true | Field | Field |
| fieldId |  |  | false | integer(int32) |  |
| status |  |  | false | string |  |
| name |  |  | false | string |  |
| farmId |  |  | false | integer(int32) |  |
| soilType |  |  | false | string |  |
| phLevel |  |  | false | number(double) |  |
| vertex1Latitude |  |  | false | number(double) |  |
| vertex1Longitude |  |  | false | number(double) |  |
| vertex2Latitude |  |  | false | number(double) |  |
| vertex2Longitude |  |  | false | number(double) |  |
| vertex3Latitude |  |  | false | number(double) |  |
| vertex3Longitude |  |  | false | number(double) |  |
| vertex4Latitude |  |  | false | number(double) |  |
| vertex4Longitude |  |  | false | number(double) |  |
| photo |  |  | false | string |  |

**响应参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名称** | **参数说明** | **类型** | **schema** |
| ResponseEntity<?> |  |  |  |

4.3.2.6.2 **updateFieldStatus**



**接口地址**

**PUT** /api/field/{fieldId}/status/{status}

**接口描述**

更新田地状态。

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

*/*

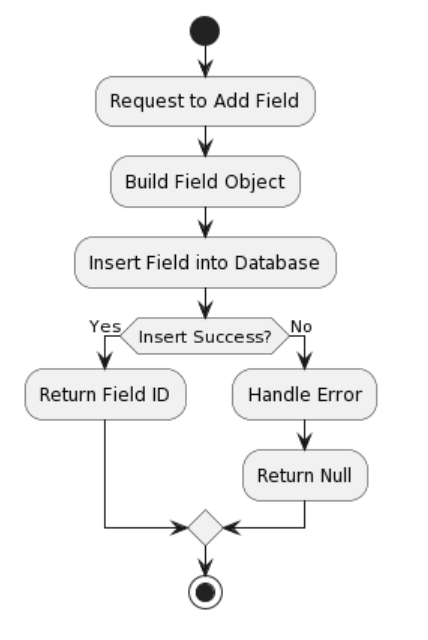
**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| fieldId |  | path | true | integer(int32) |  |
| status |  | path | true | string |  |

**响应参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名称** | **参数说明** | **类型** | **schema** |
| ResponseEntity<?> |  |  |  |

4.3.2.6.3 **addField**



**接口地址**

新建田地

**POST** /api/field/addField

**接口描述**

新建田地。

**请求数据类型**

application/x-www-form-urlencoded,application/json

**响应数据类型**

*/*

**请求示例**

{

"status": "",

"name": "",

"farmId": 0,

"soilType": "",

"phLevel": 0,

"vertex1Latitude": 0,

"vertex1Longitude": 0,

"vertex2Latitude": 0,

"vertex2Longitude": 0,

"vertex3Latitude": 0,

"vertex3Longitude": 0,

"vertex4Latitude": 0,

"vertex4Longitude": 0,

"photo": ""

}

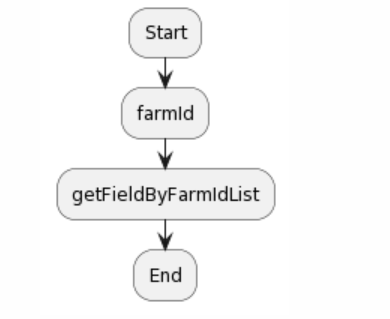
**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| fieldRequest | FieldRequest | body | true | FieldRequest | FieldRequest |
| status |  |  | false | string |  |
| name |  |  | false | string |  |
| farmId |  |  | false | integer(int32) |  |
| soilType |  |  | false | string |  |
| phLevel |  |  | false | number(double) |  |
| vertex1Latitude |  |  | false | number(double) |  |
| vertex1Longitude |  |  | false | number(double) |  |
| vertex2Latitude |  |  | false | number(double) |  |
| vertex2Longitude |  |  | false | number(double) |  |
| vertex3Latitude |  |  | false | number(double) |  |
| vertex3Longitude |  |  | false | number(double) |  |
| vertex4Latitude |  |  | false | number(double) |  |
| vertex4Longitude |  |  | false | number(double) |  |
| photo |  |  | false | string |  |

**响应参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名称** | **参数说明** | **类型** | **schema** |
| ResponseEntity<?> |  |  |  |

4.3.2.6.4 **getFieldByFarmId**



**接口地址**

**GET** /api/field/farm/{farmId}

**接口描述**

获取对应农场id的所有田地。

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

*/*

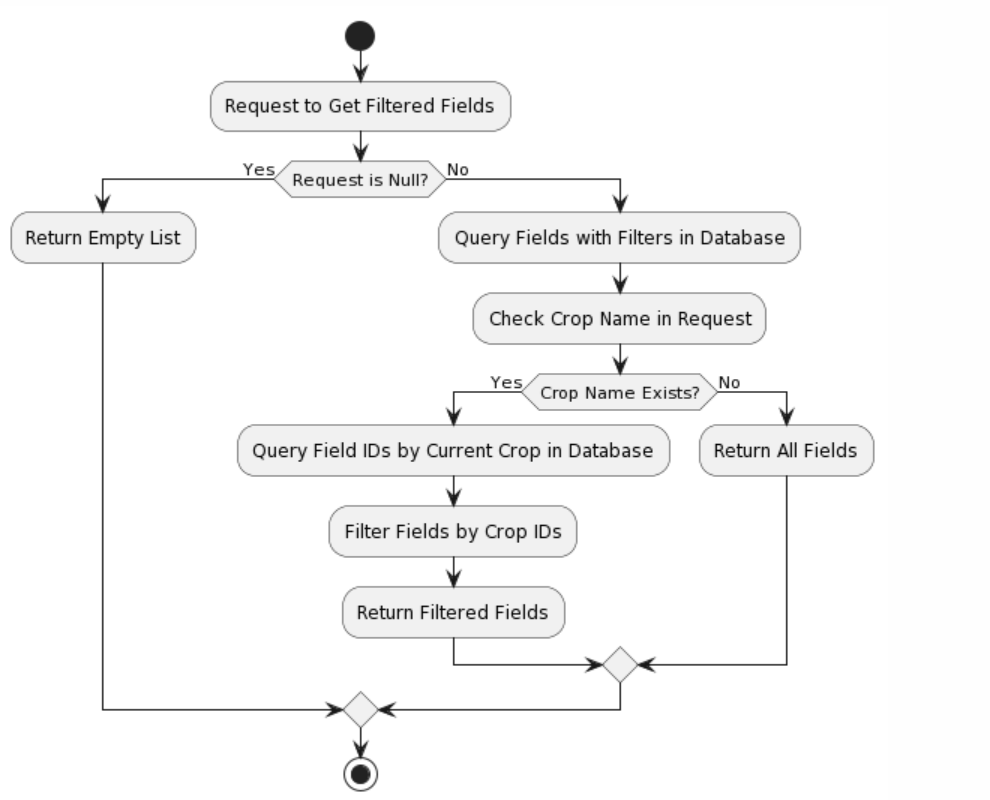
**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| farmId |  | path | true | integer(int32) |  |

**响应参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名称** | **参数说明** | **类型** | **schema** |
| fieldId |  | integer(int32) | integer(int32) |
| status |  | string |  |
| name |  | string |  |
| farmId |  | integer(int32) | integer(int32) |
| soilType |  | string |  |
| phLevel |  | number(double) | number(double) |
| vertex1Latitude |  | number(double) | number(double) |
| vertex1Longitude |  | number(double) | number(double) |
| vertex2Latitude |  | number(double) | number(double) |
| vertex2Longitude |  | number(double) | number(double) |
| vertex3Latitude |  | number(double) | number(double) |
| vertex3Longitude |  | number(double) | number(double) |
| vertex4Latitude |  | number(double) | number(double) |
| vertex4Longitude |  | number(double) | number(double) |
| photo |  | string |  |

4.3.2.6.5 **getFilteredFields**



**接口地址**

**GET** /api/field/filter

**接口描述**

获取通过条件筛选后的田地。

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

*/*

**请求参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数名称** | **参数说明** | **请求类型** | **必须** | **数据类型** | **schema** |
| request |  | query | true | FilterRequest | FilterRequest |
| farmId |  |  | false | integer(int32) |  |
| status |  |  | false | string |  |
| cropName |  |  | false | string |  |
| soilType |  |  | false | string |  |
| minPh |  |  | false | number(double) |  |
| maxPh |  |  | false | number(double) |  |

**响应参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数名称** | **参数说明** | **类型** | **schema** |
| fieldId |  | integer(int32) | integer(int32) |
| status |  | string |  |
| name |  | string |  |
| farmId |  | integer(int32) | integer(int32) |
| soilType |  | string |  |
| phLevel |  | number(double) | number(double) |
| vertex1Latitude |  | number(double) | number(double) |
| vertex1Longitude |  | number(double) | number(double) |
| vertex2Latitude |  | number(double) | number(double) |
| vertex2Longitude |  | number(double) | number(double) |
| vertex3Latitude |  | number(double) | number(double) |
| vertex3Longitude |  | number(double) | number(double) |
| vertex4Latitude |  | number(double) | number(double) |
| vertex4Longitude |  | number(double) | number(double) |
| photo |  | string |  |

4.4 **登录微服务**

4.4.1 **Employee类**

4.4.1.1 **功能**

登陆注册模块中的Employee类主要负责记录用户的登录账号（在阡陌管理系统中是手机号码）和密码。并且通过内部的方法实现检查用户登录账号和密码是否匹配，并且通过修改密码方法管理自身密码。

4.4.1.2 **性能**

无约束

4.4.1.3 **输入项**

编辑密码的时候需要提供登陆的请求信息；发送验证码的时候需要提供员工的相应信息；检查权限的时候需要提供员工的id属性。

4.4.1.4 **输出项**

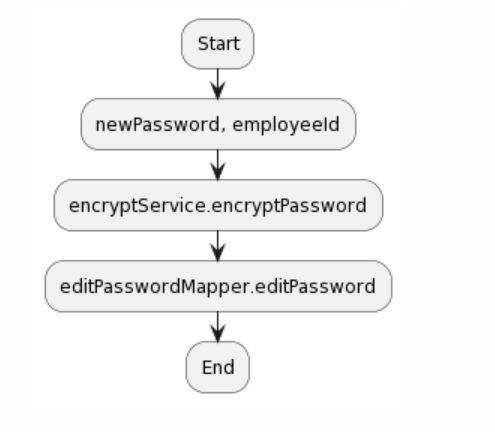
检查密码时，只会返回“正确”或“错误”进行反馈（布尔值）。其余情况下没有输出项。

4.4.1.5 **属性描述**

|  |  |  |
| --- | --- | --- |
| 字段名 | 数据类型 | 说明 |
| employee\_id | Int | 员工ID |
| name | Varchar | 员工用户名 |
| password | Varchar | 员工密码 |
| phonenumber | Varchar | 员工手机号 |
| api\_key | char | Gpt的接口密钥 |
| type | int | 员工的部门 |

4.4.1.6 **方法详细描述**

4.4.1.6.1 **editPassword**



**接口地址**

**POST** /edit/edit

**接口描述**

给相应的用户修改密码

**请求数据类型**

application/x-www-form-urlencoded,application/json

**响应数据类型**

*/*

**请求示例**

{

"employeeId": 0,

"password": ""

}

**请求参数**



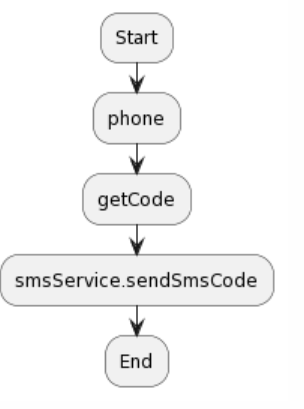
**点击图片可查看完整电子表格**

**响应参数**



**点击图片可查看完整电子表格**

4.4.1.6.2 **sendSmsCode**



**接口地址**

**POST** /edit/sms

**接口描述**

给当前用户输入的手机号码返回验证码

**请求数据类型**

application/x-www-form-urlencoded,application/json

**响应数据类型**

*/*

**请求示例**

{

"employeeId": 0

}

**请求参数**



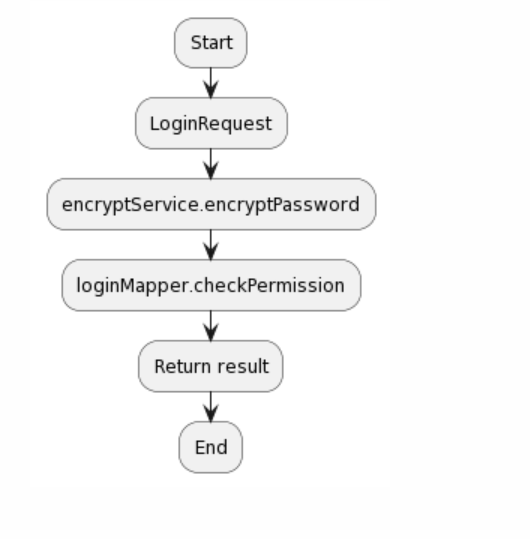
**点击图片可查看完整电子表格**

**响应参数**



**点击图片可查看完整电子表格**

4.4.1.6.3 **checkPermission**



**接口地址**

**POST** /api/login/login

**接口描述**

检查相应用户的登陆权限

**请求数据类型**

application/x-www-form-urlencoded,application/json

**响应数据类型**

*/*

**请求示例**

{

"employeeId": 0,

"password": ""

}

**请求参数**



**点击图片可查看完整电子表格**

**响应参数**



**点击图片可查看完整电子表格**

4.5 **统计微服务**

4.5.1 **功能**

统计微服务主要调用别的微服务的类，并没有自己的类，而是对其他类进行信息的整合与处理。

4.5.2 **性能**

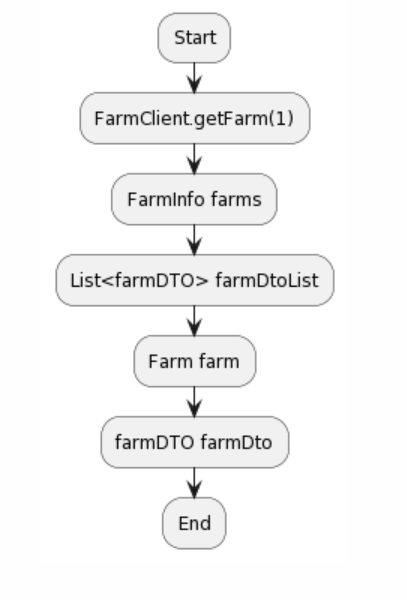
4.5.3 **输入项**

农场分配函数负责返回农场位置信息和个数信息；保质期预警榜负责返回收获的产品和产量；

4.5.4 **输出项**

4.5.5 **方法详细描述**

4.5.5.1 **farmStatistics**



**接口地址**

**GET** /api/statistics/farm\_distribution

**接口描述**

**请求数据类型**

/

**响应数据类型**

*/*

**请求参数**



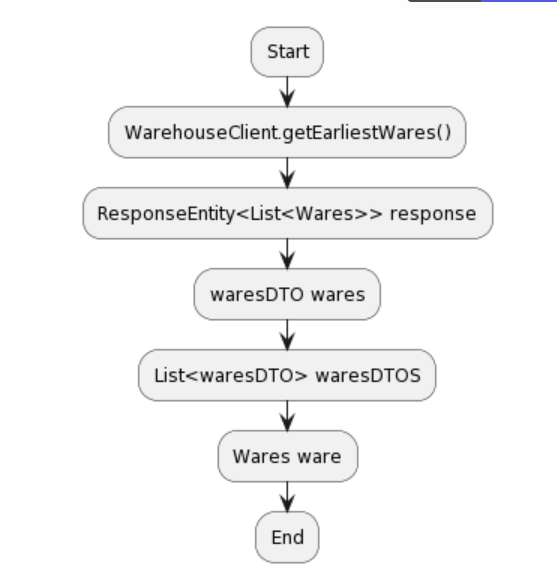
**点击图片可查看完整电子表格**

**响应参数**



**点击图片可查看完整电子表格**

4.5.5.2 **waresExpiringStatistics**



**接口地址**

**GET** /api/statistics/expiring\_rank

**接口描述**

**请求数据类型**

/

**响应数据类型**

*/*

**请求参数**



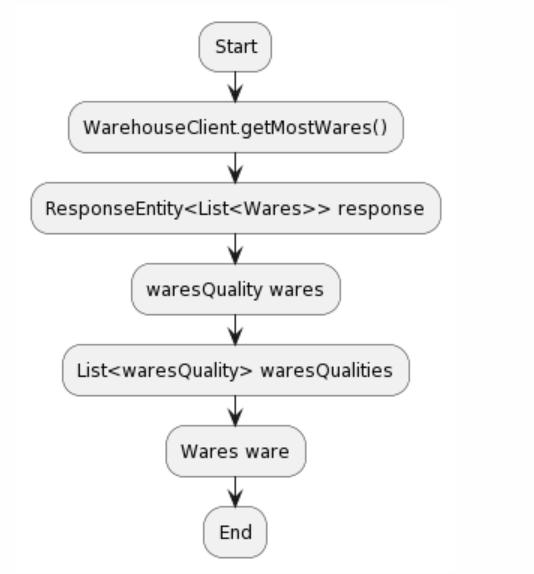
**点击图片可查看完整电子表格**

**响应参数**



**点击图片可查看完整电子表格**

4.5.5.3 **waresQualityStatistics**



**接口地址**

**GET** /api/statistics/wares\_quality

**接口描述**

**请求数据类型**

/

**响应数据类型**

*/*

**请求参数**



**点击图片可查看完整电子表格**

**响应参数**



**点击图片可查看完整电子表格**

4.5.5.4 **harvestStatistics**



**接口地址**

**GET** /api/statistics/harvest\_quality

**接口描述**

**请求数据类型**

/

**响应数据类型**

*/*

**请求参数**



**点击图片可查看完整电子表格**

**响应参数**



**点击图片可查看完整电子表格**

4.6 **种植活动微服务**

4.6.1 **Activity类**

4.6.1.1 **功能**

活动类主要负责对于种植中的农事活动进行处理。

4.6.1.2 **性能**

无约束

4.6.1.3 **输入项**

获取活动详细信息的时候的输入参数是活动Id；通过设备解决问题的输入参数是和设备有关的相应信息。

4.6.1.4 **输出项**

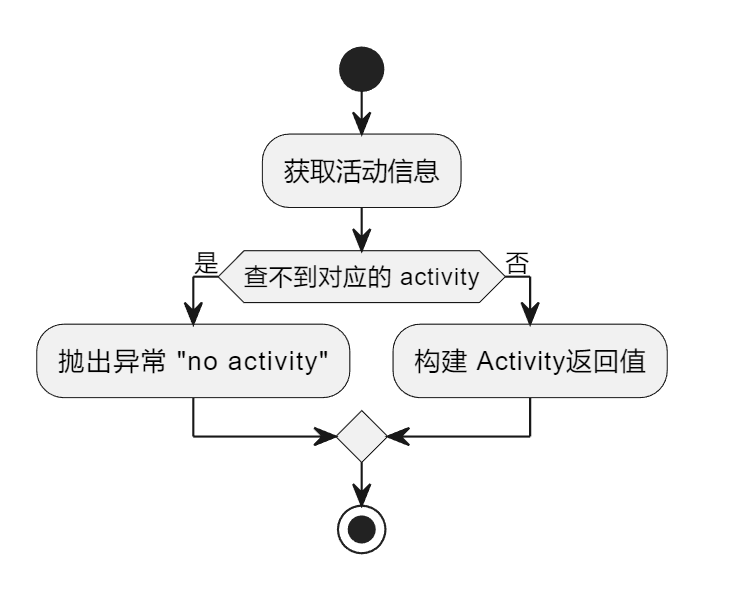
获取活动详细信息的时候的输出参数是和活动有关的详细信息；通过设备解决问题的输出参数是stack货存

4.6.1.5 **属性描述**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 说明 | 备注 |
| activity\_id | Int |  | 活动id | PK |
| issue\_id | Int |  | 事件id | FK |
| equipment\_id | int |  | 设备id | FK, NULLABLE |
| consumables\_id | int |  | 消耗品id | FK, NULLABLE |
| detail | varchar | 255 | 活动额外信息 | NULLABLE |
| amount | int |  | 消耗品使用数量 | NULLABLE |
| employee\_id | int |  | 科技部人员id | FK |

4.6.1.6 **方法详细描述**

4.6.1.6.1 **getActivityById**



**接口地址**

**GET** /api/activity/{activityId}

**接口描述**

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

*/*

**请求参数**



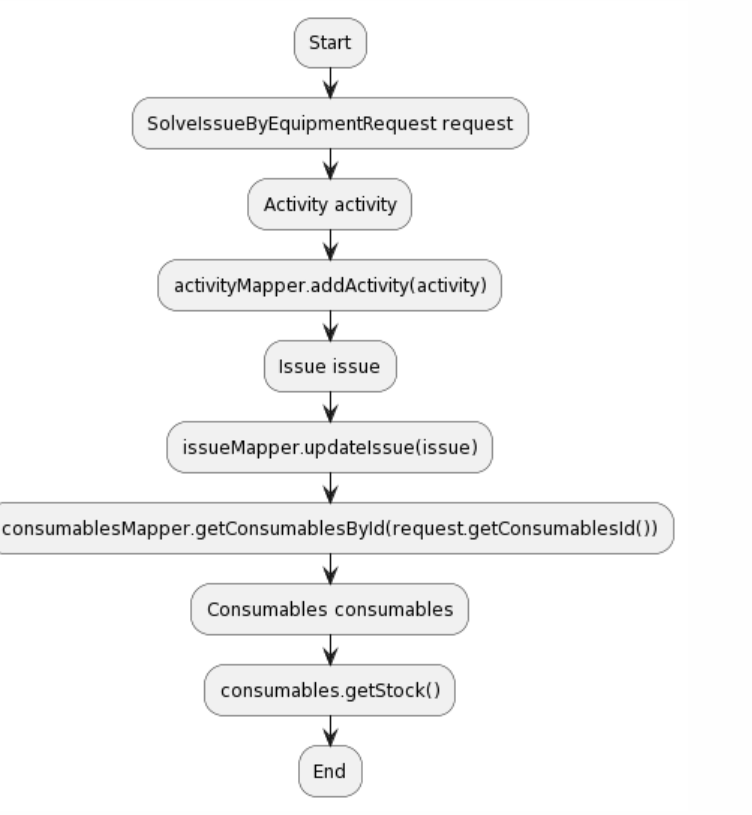
**点击图片可查看完整电子表格**

**响应参数**



**点击图片可查看完整电子表格**

4.6.1.6.2 **solveIssueByEquipment**



**接口地址**

**PUT** /api/activity/by\_equipment

**接口描述**

**请求数据类型**

application/x-www-form-urlencoded,application/json

**响应数据类型**

*/*

**请求示例**

{

"issueId": 0,

"equipmentId": 0,

"consumablesId": 0,

"time": "",

"employeeId": 0

}

**请求参数**



**点击图片可查看完整电子表格**

**响应参数**



**点击图片可查看完整电子表格**

4.6.2 **Harvest类**

4.6.2.1 **功能**

收获类主要负责对于已收获的作物进行处理。

4.6.2.2 **性能**

无约束

4.6.2.3 **输入项**

获取当前收获信息的时候的输入参数是田地Id；检查收获状态的输入参数是种植id和终止日期；获取历史收获信息的时候的输入参数是田地Id；获取收获信息的时候的输入参数是收获Id；获取收获状态的输入参数是农场Id，作物Id和timemode。

4.6.2.4 **输出项**

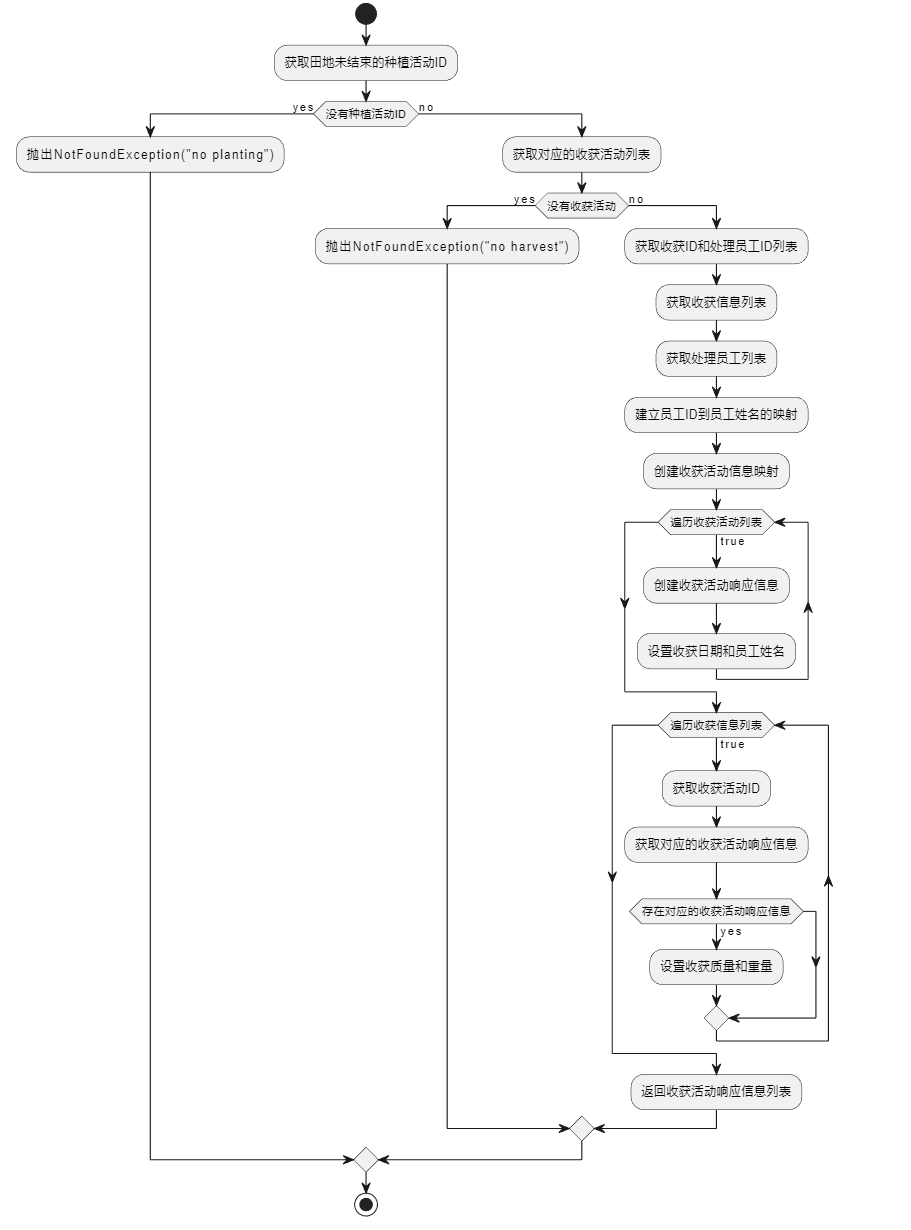
输出的是和收获有关的详细信息。

4.6.2.5 **属性描述**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 说明 | 备注 |
| harvest\_id | int |  | 收获id | PK |
| harvest\_date | date |  | 收获日期 |  |
| employee\_id | int |  | 技术人员id | FK |
| planting\_id | int |  | 种植id | FK |
| detail | varchar | 255 | 活动额外信息 | NULLABLE |
| amount | int |  | 消耗品使用数量 | NULLABLE |
| employee\_id | int |  | 科技部人员id | FK |

4.6.2.6 **方法详细描述**

4.6.2.6.1 **getCurrentHarvest**



**接口地址**

**GET** /api/harvest/current/field/{fieldId}

**接口描述**

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

*/*

**请求参数**



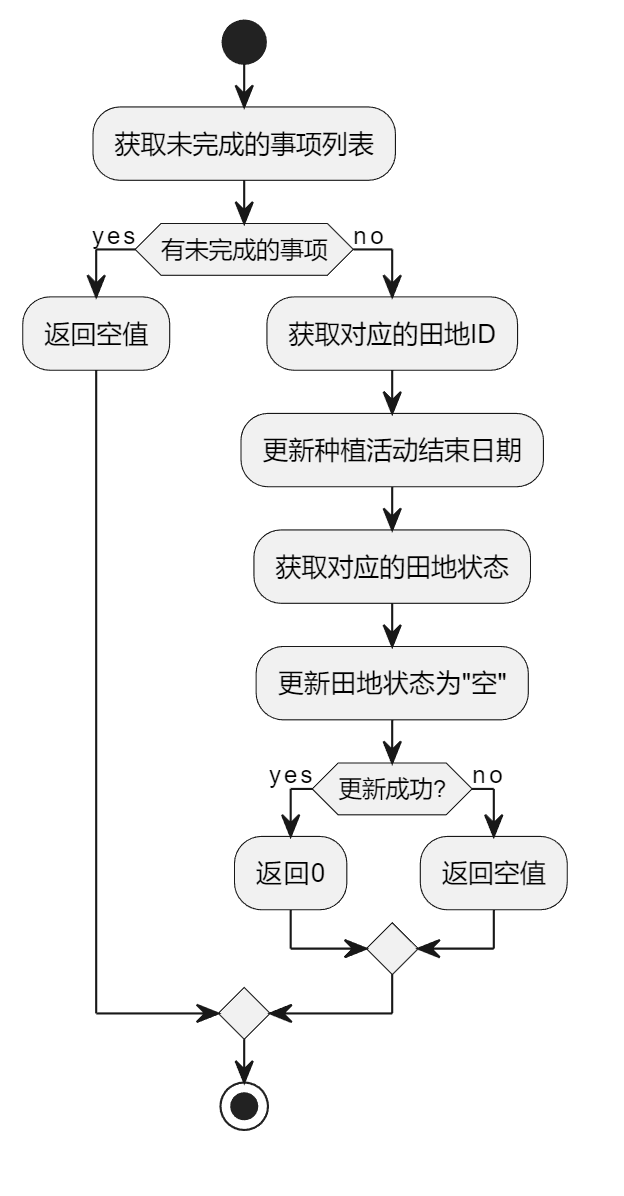
**点击图片可查看完整电子表格**

**响应参数**



**点击图片可查看完整电子表格**

4.6.2.6.2 **checkHarvestStatus**



**接口地址**

**PUT** /api/harvest/finish

**接口描述**

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

*/*

**请求参数**



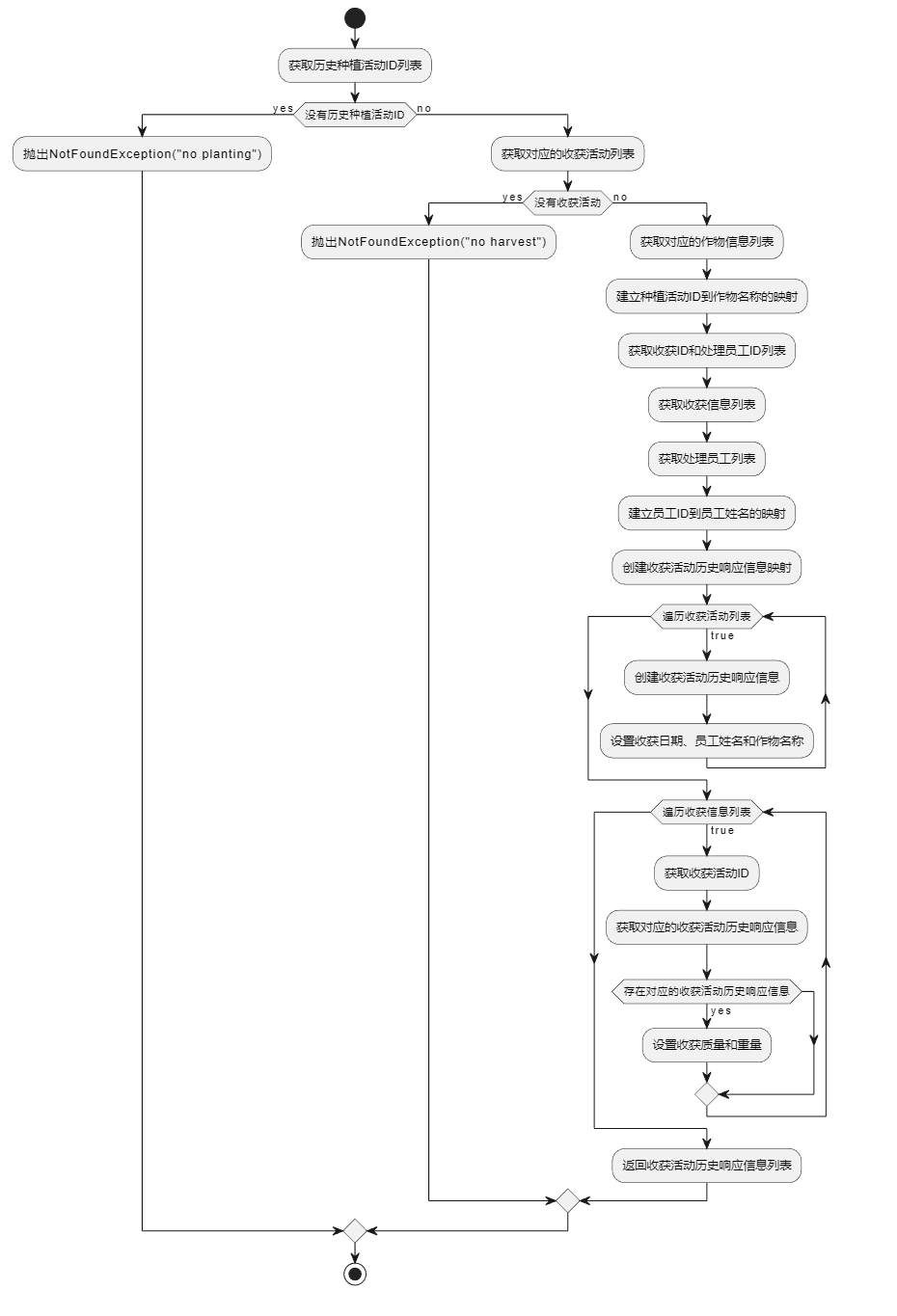
**点击图片可查看完整电子表格**

**响应参数**



**点击图片可查看完整电子表格**

4.6.2.6.3 **getHistoryHavest**



**接口地址**

**GET** /api/harvest/history/field/{fieldId}

**接口描述**

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

*/*

**请求参数**



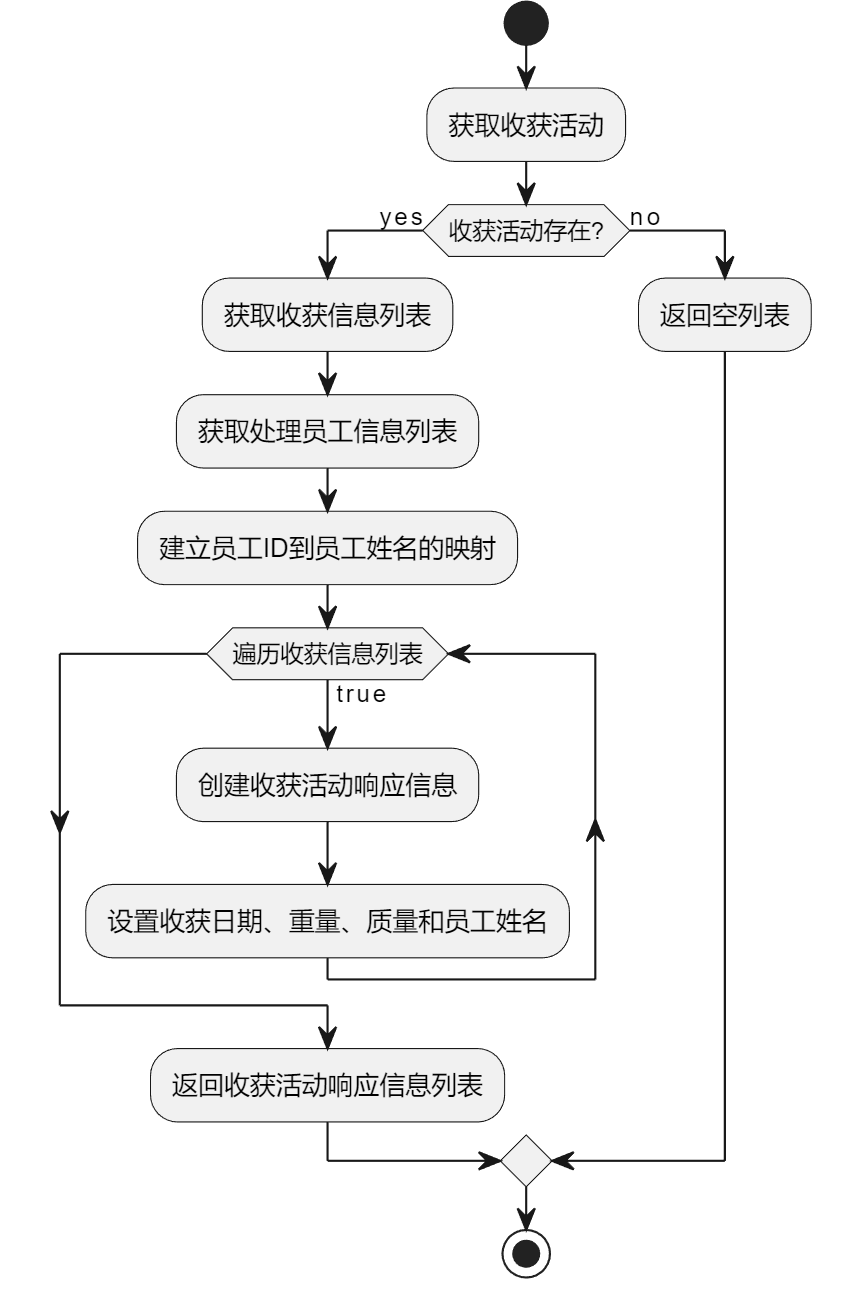
**点击图片可查看完整电子表格**

**响应参数**



**点击图片可查看完整电子表格**

4.6.2.6.4 **getHarvestInfo**



**接口地址**

**GET** /api/harvest/info/{harvestId}

**接口描述**

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

*/*

**请求参数**



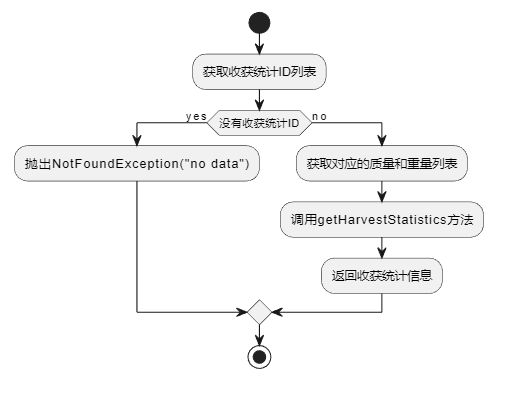
**点击图片可查看完整电子表格**

**响应参数**



**点击图片可查看完整电子表格**

4.6.2.6.5 **getHarvestStatistics**



**接口地址**

**GET** /api/harvest/statistics/farm/crop

**接口描述**

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

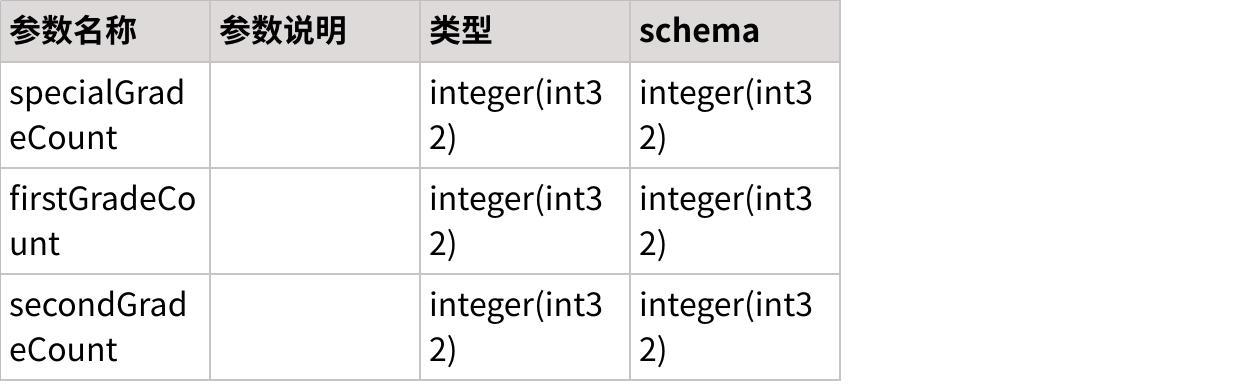
*/*

**请求参数**



**点击图片可查看完整电子表格**

**响应参数**



**点击图片可查看完整电子表格**

4.6.3 **Issue类**

4.6.3.1 **功能**

问题类主要负责对于种植过程中产生的问题进行处理和信息的展示。

4.6.3.2 **性能**

无约束

4.6.3.3 **输入项**

获取问题的函数的输入项是田地Id，页码和大小；添加问题的函数的输入项是问题的请求信息。

4.6.3.4 **输出项**

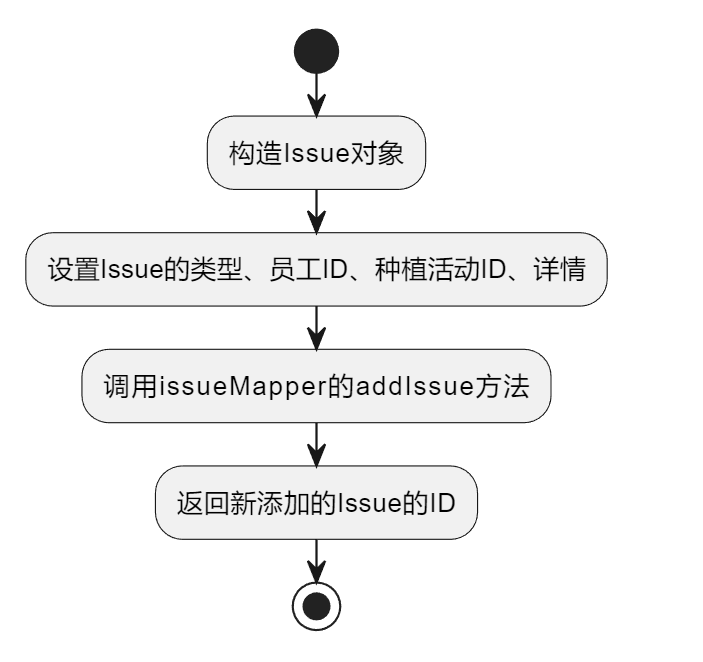
输出的是和问题有关的详细信息。

4.6.3.5 **属性描述**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 说明 | 备注 |
| issue\_id | int |  | 事件id | PK |
| type | varchar | 20 | 事件类型 | e.g., Drought, Insect Attack, Disease, Flood, etc. |
| start\_date | date |  | 事件开始日期 |  |
| address\_date | date |  | 事件开始处理日期 | NULLABLE |
| end\_date | date |  | 事件结束日期 | NULLABLE |
| planting\_id | int |  | 种植id | FK |
| detail | varchar | 255 | 细节 | e.g., Crop Loss, Reduced Yield, Delayed Harvest, etc. |
| employee\_id | int |  | 技术人员id | FK |

4.6.3.6 **方法详细描述**

4.6.3.6.1 **addIssue**



**接口地址**

**POST** /api/issue

**接口描述**

**请求数据类型**

application/x-www-form-urlencoded,application/json

**响应数据类型**

*/*

**请求示例**

{

"type": "",

"employeeId": 0,

"plantingId": 0,

"detail": ""

}

**请求参数**



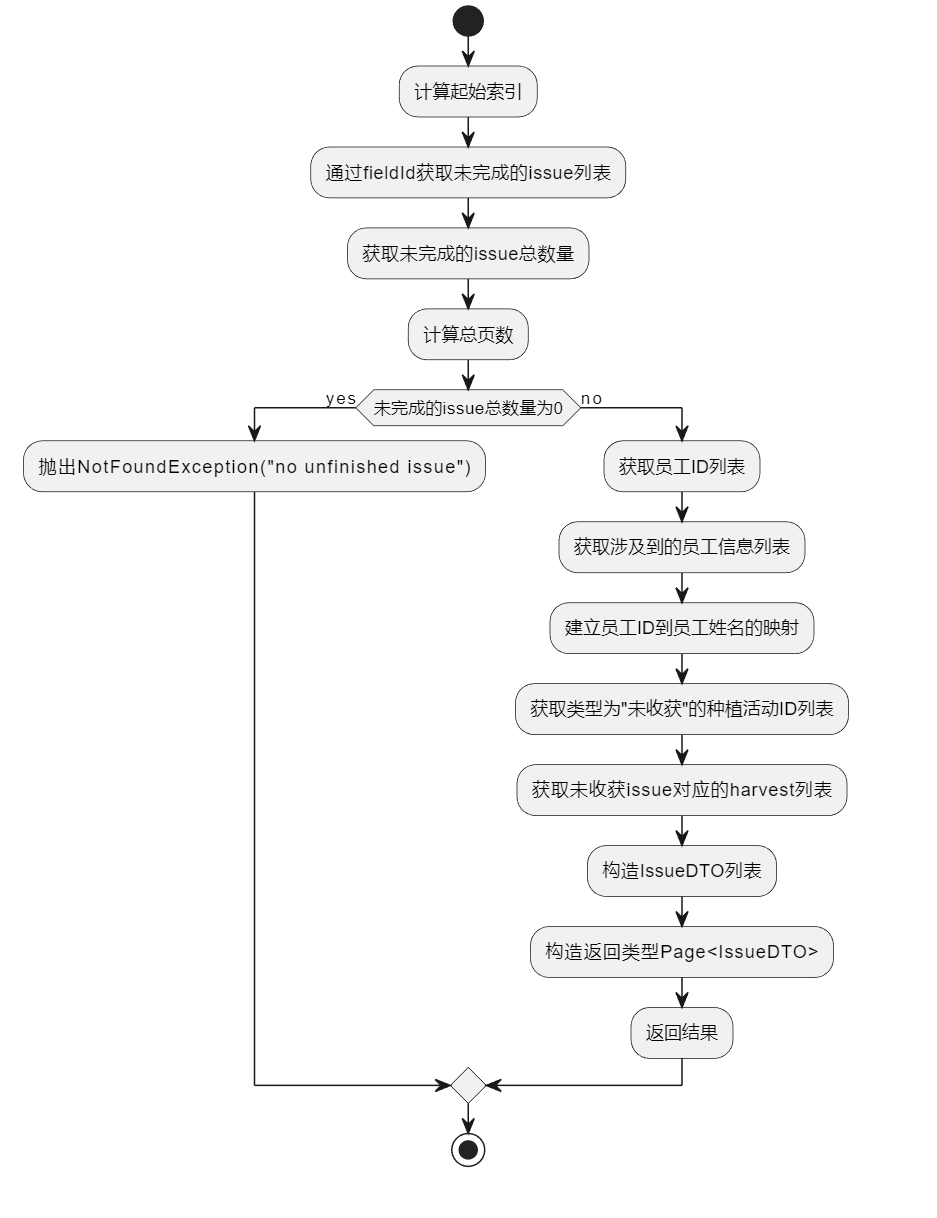
**点击图片可查看完整电子表格**

**响应参数**



**点击图片可查看完整电子表格**

4.6.3.6.2 **getIssuesByFieldId**



**接口地址**

**GET** /api/issue/field/{filedId}

**接口描述**

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

*/*

**请求参数**



**点击图片可查看完整电子表格**

**响应参数**



**点击图片可查看完整电子表格**

4.6.4 **Planting类**

4.6.4.1 **功能**

种植类主要负责对于种植中的作物进行处理。

4.6.4.2 **性能**

无约束

4.6.4.3 **输入项**

添加种植和问题的函数的输入是种植请求的相关信息；获取农场Id和作物Id的输入参数是种植Id。

4.6.4.4 **输出项**

输出的是和planting相关联的表的详细信息。

4.6.4.5 **属性描述**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 说明 | 备注 |
| planting\_id | int |  | 种植id | PK |
| field\_id | int |  | 田地id | FK |
| crop\_id | int |  | 作物id | FK |
| start\_date | date |  | 田地开始占用日期 | 开始日期早于结束日期 |
| end\_date | date |  | 田地结束占用日期 |  |
| employee\_id | int |  | 技术人员id | FK |
| employee\_id | int |  | 科技部人员id | FK |

4.6.4.6 **方法详细描述**

4.6.4.6.1 **initPlanting**



**接口地址**

**PUT** /api/planting

**接口描述**

**请求数据类型**

application/x-www-form-urlencoded,application/json

**响应数据类型**

*/*

**请求示例**

{

"fieldId": 0,

"cropId": 0,

"employeeId": 0

}

**请求参数**



**点击图片可查看完整电子表格**

**响应参数**



**点击图片可查看完整电子表格**

4.7 **粮仓微服务**

4.7.1 **Warehouse类**

4.7.1.1 **功能**

粮仓类主要负责对于粮仓中产生的问题进行处理和信息的展示。

4.7.1.2 **性能**

无约束

4.7.1.3 **输入项**

添加粮仓的函数的输入项是和粮仓有关的请求信息；获取所有粮仓的输入项是空。

4.7.1.4 **输出项**

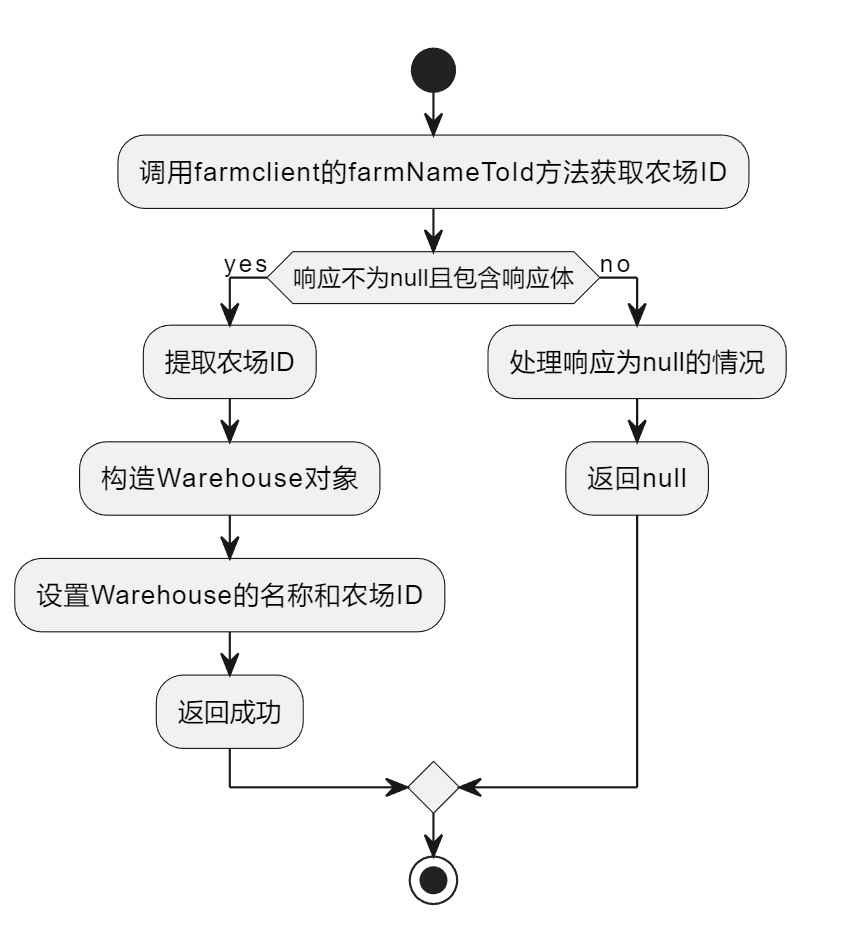
输出的是和粮仓有关的详细信息。

4.7.1.5 **属性描述**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 实体名称 | 粮仓 | 长度 | 说明 | 备注 |
| 标识符 | warehouse |  | 事件id | PK |
| 描述 | 记录粮仓的信息 | 20 | 事件类型 | e.g., Drought, Insect Attack, Disease, Flood, etc. |
| 主键 | 仓库ID |  | 事件开始日期 |  |
| 数据项 | 仓库ID, name, farm\_id |  | 事件开始处理日期 | NULLABLE |
| end\_date | date |  | 事件结束日期 | NULLABLE |
| planting\_id | int |  | 种植id | FK |
| detail | varchar | 255 | 细节 | e.g., Crop Loss, Reduced Yield, Delayed Harvest, etc. |
| employee\_id | int |  | 技术人员id | FK |

4.7.1.6 **方法详细描述**

4.7.1.6.1 **addWarehouse**



**接口地址**

**PUT** /api/warehouse/add\_warehouse

**接口描述**

**请求数据类型**

application/x-www-form-urlencoded,application/json

**响应数据类型**

*/*

**请求示例**

{

"farmName": "",

"warehouseName": ""

}

**请求参数**



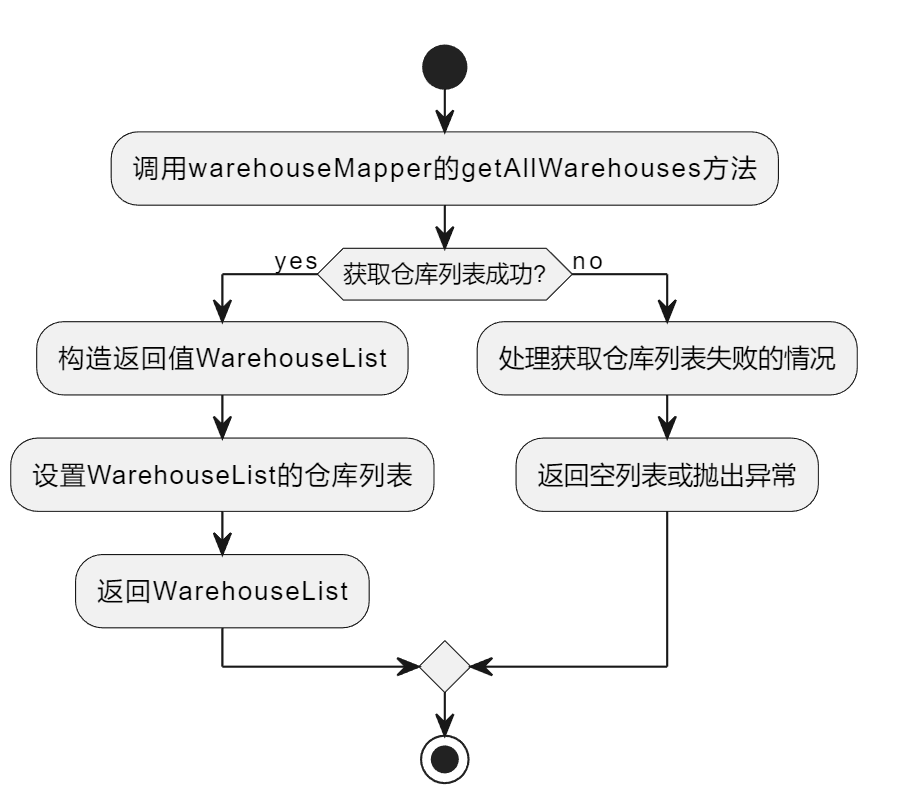
**点击图片可查看完整电子表格**

**响应参数**



**点击图片可查看完整电子表格**

4.7.1.6.2 **getAllWarehouse**



**接口地址**

**GET** /api/warehouse/get\_all\_warehouse

**接口描述**

**请求数据类型**

application/x-www-form-urlencoded

**响应数据类型**

*/*

**请求参数**



**点击图片可查看完整电子表格**

**响应参数**



**点击图片可查看完整电子表格**