

# 根瘤菌条件化固氮评价系统

## V1.0

### —设计说明书

# 第一章 前言

## 1.1 安装与初始化及软件主要参数设置

软件环境：

JDK1.8

Maven: 3.3.9

Mysql:5.7

软件编译：

```
cd /strainInfo/
```

```
mvn clean install
```

软件部署：

```
./start.sh
```

## 1.2 软件故障排除

查看服务进程：

```
ps -ef | grep strainInfo
```

查看进程是否存在：

如果不存在，进入/strainInfo/

启动服务：./start.sh

查看日志：

```
cd /logs
```

```
tail -f logs.log
```

## 1.3 运行环境

硬件环境(主机):

主机配置: i7 9 代, 内存 8G, 硬盘 256G

软件环境(主机):

服务器环境: 操作系统: CentOS7.5

CPU: 4cores

内存: 32GB

DISK: 500GB

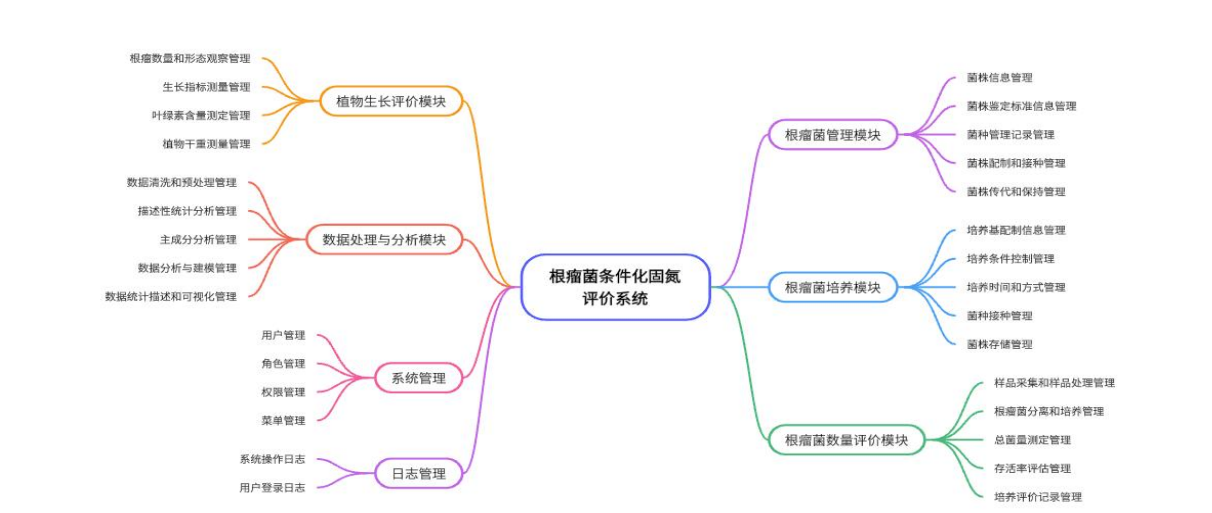
客户端: 终端

## 1.4 系统概述

根瘤菌条件化固氮评价系统是一种评价根瘤菌在不同环境条件下固氮效果的系统。该系统可以用于评价根瘤菌菌株的固氮能力, 优选高效的根瘤菌菌株, 并优化根瘤菌的培养条件, 以提高其固氮效果。同时, 该系统还可以用于研究不同环境因素对根瘤菌固氮效果的影响, 为优化植物固氮技术提供科学依据。

## 第二章 总体设计

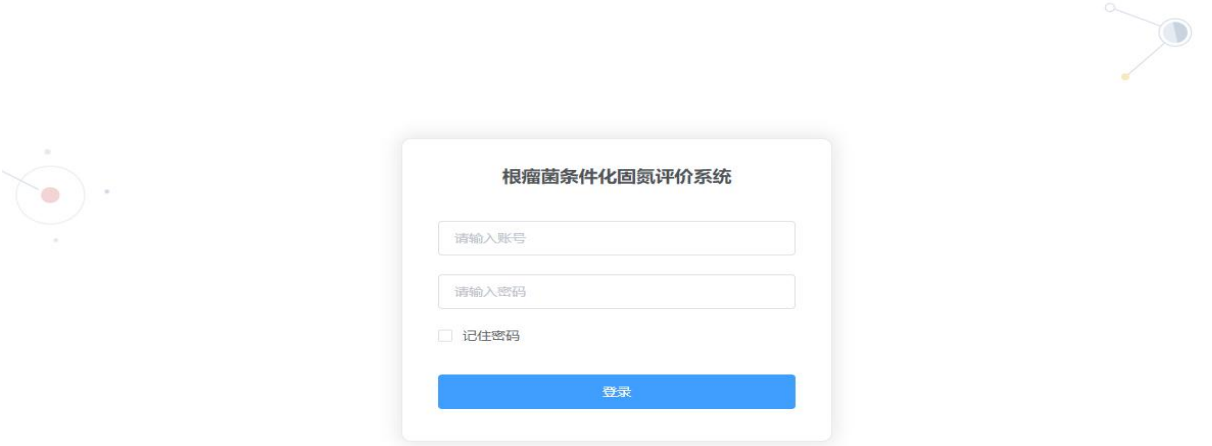
### 2.1 功能结构图



### 2.2 界面效果图

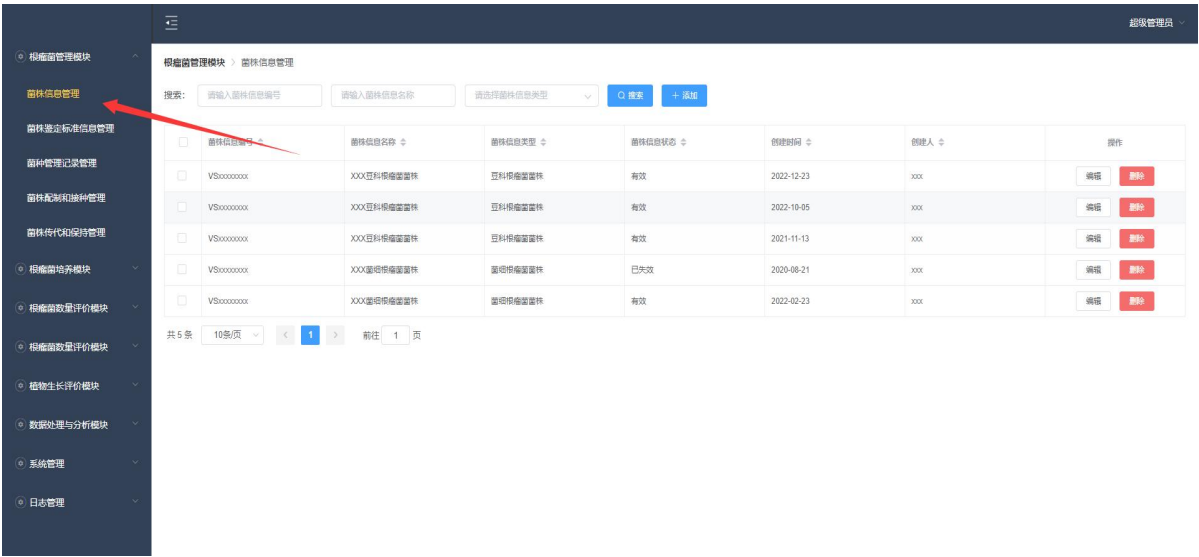
#### 2.2.1 登录系统

在浏览器输入系统地址，点击【enter】键，进入系统登录页面，用户依次输入自己的账号、密码后，点击【登录】按钮，进入根瘤菌条件化固氮评价系统。



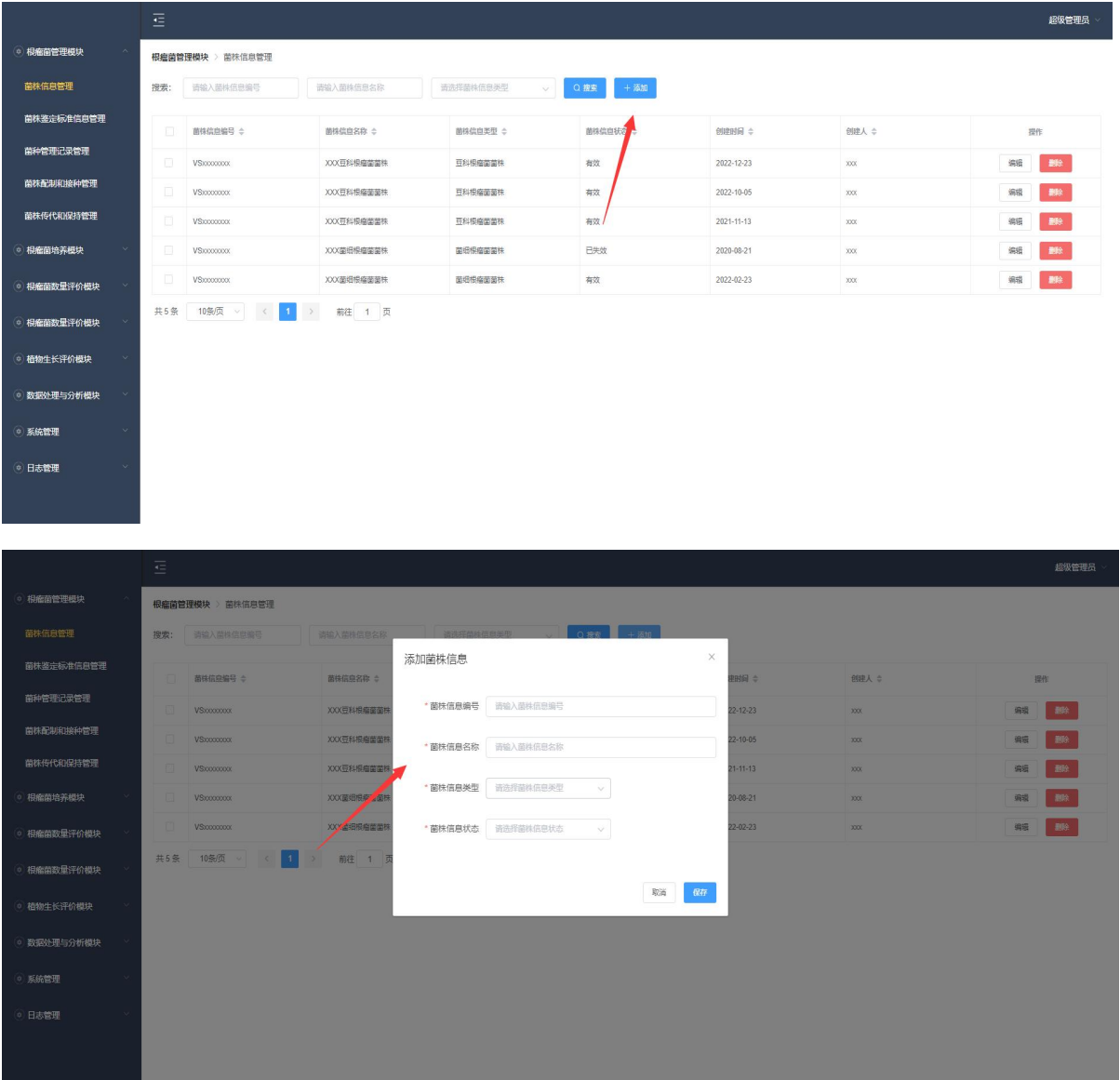
2.2.2 菌株信息管理

点击【根瘤菌管理模块】菜单，再点击菌株信息管理，展示菌株信息管理列表，展示信息：菌株信息编号、菌株信息名称、菌株信息类型、菌株信息状态、创建时间、创建人。



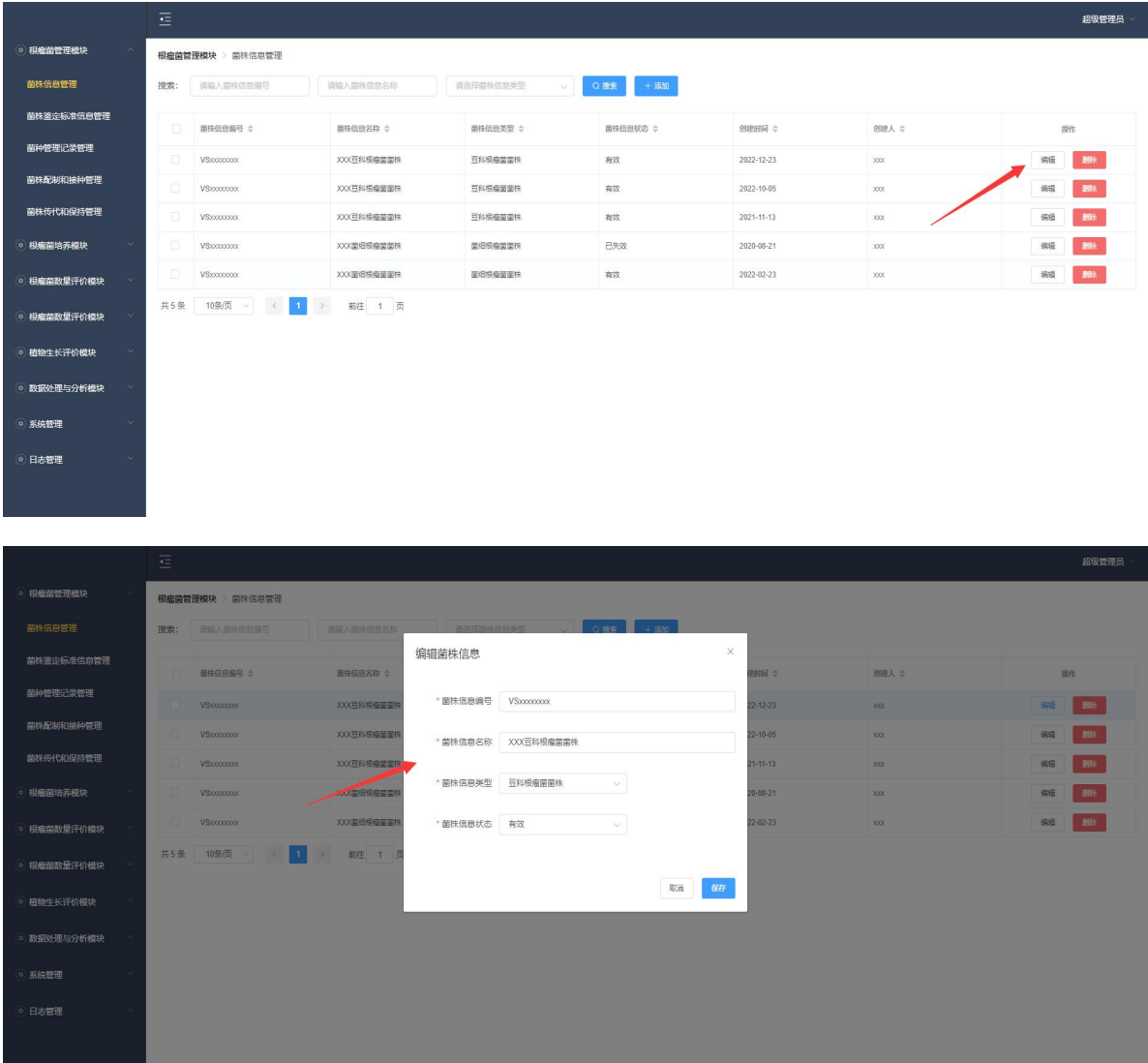
2.2.3 菌株信息添加

点击添加按钮，弹出菌株信息添加弹窗，填写信息添加即可。



2.3.4 菌株信息编辑

点击右侧编辑按钮，弹出菌株信息编辑弹窗，填写信息编辑即可。



## 2.2.5 培养基配制信息管理

点击【根瘤菌培养模块】菜单，再点击培养基配制信息管理进入列表页。展示信息：

培养基配制信息编号、培养基配制信息名称、培养基配制信息类型、培养基配制信息状态、创建时间。

培养基配制信息管理

搜索:  请输入培养基配制信息编号  请输入培养基配制信息名称  请选择培养基配制信息类型

	培养基配制信息编号	培养基配制信息名称	培养基配制信息类型	培养基配制信息状态	创建时间	操作
1	VC000000	xxx基本成分信息	基本成分信息	有效	2020-08-12	<input type="button" value="编辑"/> <input type="button" value="删除"/>
2	VC000000	xxx基本成分信息	基本成分信息	有效	2022-10-24	<input type="button" value="编辑"/> <input type="button" value="删除"/>
3	VC000000	xxx基本成分信息	基本成分信息	有效	2022-01-12	<input type="button" value="编辑"/> <input type="button" value="删除"/>
4	VC000000	xxx氧气含量信息	氧气含量信息	已失效	2022-05-17	<input type="button" value="编辑"/> <input type="button" value="删除"/>
5	VC000000	xxx氧气含量信息	氧气含量信息	已失效	2021-09-12	<input type="button" value="编辑"/> <input type="button" value="删除"/>

共 5 条 10 条/页 < 1 > 前往 1 页

## 2.2.6 培养基配制信息添加

点击添加按钮，弹出培养基配制信息添加弹窗，填写信息添加即可。

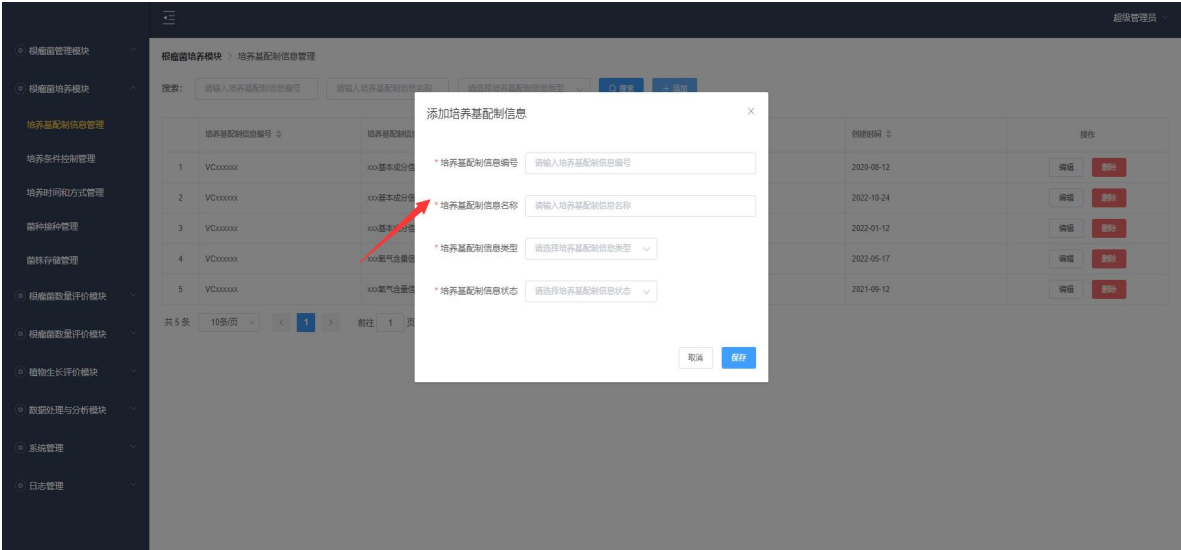
培养基配制信息管理

搜索:  请输入培养基配制信息编号  请输入培养基配制信息名称  请选择培养基配制信息类型

	培养基配制信息编号	培养基配制信息名称	培养基配制信息类型	培养基配制信息状态	创建时间	操作
1	VC000000	xxx基本成分信息	基本成分信息	有效	2020-08-12	<input type="button" value="编辑"/> <input type="button" value="删除"/>
2	VC000000	xxx基本成分信息	基本成分信息	有效	2022-10-24	<input type="button" value="编辑"/> <input type="button" value="删除"/>
3	VC000000	xxx基本成分信息	基本成分信息	有效	2022-01-12	<input type="button" value="编辑"/> <input type="button" value="删除"/>
4	VC000000	xxx氧气含量信息	氧气含量信息	已失效	2022-05-17	<input type="button" value="编辑"/> <input type="button" value="删除"/>
5	VC000000	xxx氧气含量信息	氧气含量信息	已失效	2021-09-12	<input type="button" value="编辑"/> <input type="button" value="删除"/>

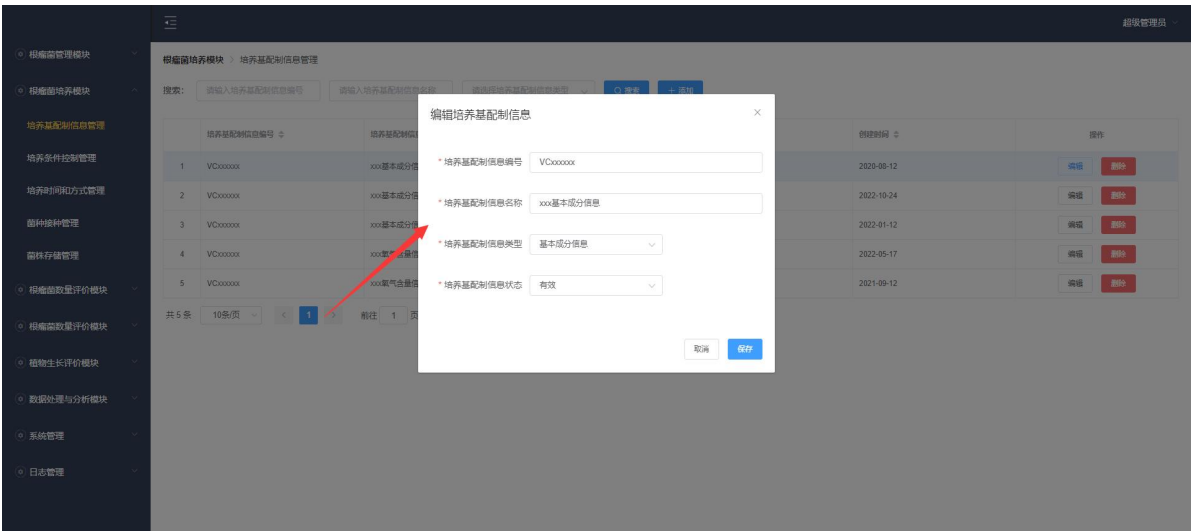
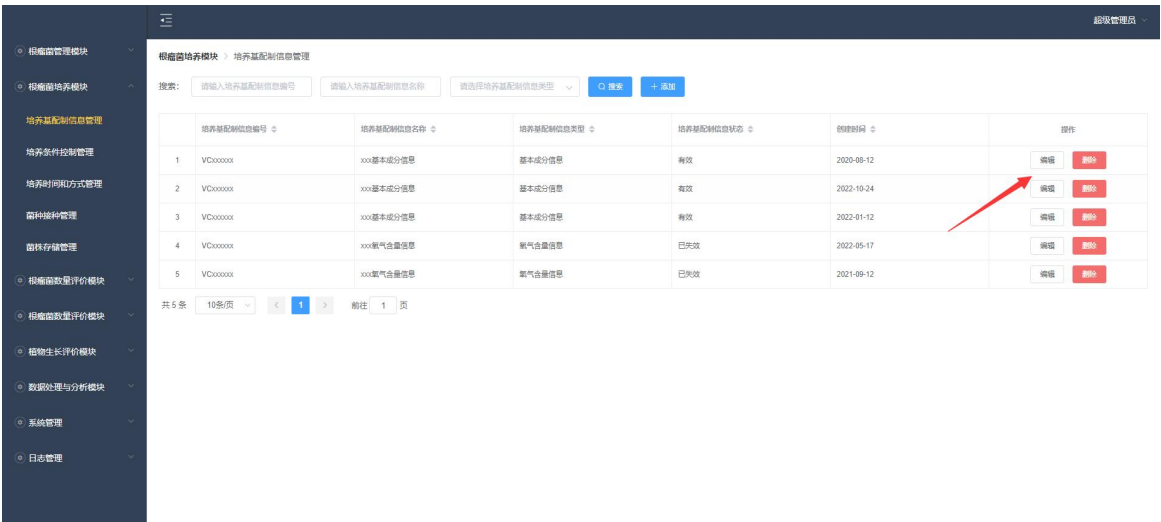
共 5 条 10 条/页 < 1 > 前往 1 页





2.2.7 培养基配制信息编辑

点击右侧编辑按钮，弹出培养基配制信息编辑弹窗，填写信息编辑即可。



### 第三章 接口设计

#### 3.1 菌株信息和培养基配制信息管理

##### 3.1.1 菌株信息管理接口

请求路径

/common/strainInfoList

请求方式 POST

json 参数

```
{
  "pageNo": 1,
  "pageSize": 10,
  "strainInfoNo": "xxxx",
  "strainInfoName": "xxxx",
  "strainInfoType": "xxxx",
  "createTime": "xxxx",
  "creator": "xxxx",
}
```

json 参数字段说明

参数名	类型	必选	说明
pageNo	String	是	分页号
pageSize	String	是	分页数量
strainInfoNo	String	是	菌株信息编号
strainInfoName	String	是	菌株信息名称
strainInfoType	String	是	菌株信息类型
createTime	Date	是	创建时间

creator	String	是	创建人
---------	--------	---	-----

返回示例

```
{
  "code": 0,
  "data": {},
  "flag": true,
  "msg": "",
}
```

返回参数说明

字段名称	字段类型	注释说明
flag	boolean	异常标识
code	int	错误码
msg	string	错误信息
data	Object	返回数据

3.1.2 菌株信息创建接口

请求路径

/common/addStrainInfo

请求方式 POST

json 参数

```
{
  "strainInfoNo": "***",
  "strainInfoName": "***",
  "strainInfoType": "***",
  "strainInfoStatus": "***",
}
```

```
        "createTime": "***",
        "creator": "***",
    }
}
```

json 参数字段说明

参数名	类型	必选	说明
strainInfoNo	String	是	菌株信息编号
strainInfoName	String	是	菌株信息名称
strainInfoType	String	是	菌株信息类型
strainInfoStatus	String	是	菌株信息状态
createTime	Date	是	创建时间
creator	String	是	创建人信息

返回示例

```
{
    "code": 0,
    "data": {},
    "flag": true,
    "msg": " ",
}
```

返回参数说明

字段名称	字段类型	注释说明
flag	boolean	异常标识
code	int	错误码
msg	string	错误信息
data	Object	返回数据

3.1.3 培养基配制信息管理接口

请求路径

/common/mediaPreparationList      请求方式 POST

json 参数

```
{
  "pageNo": 1,
  "pageSize": 10,
  "mediaPreparationNo": xxx,
  "mediaPreparationName": "xxx",
  "mediaPreparationType": "xxxx",
  "createTime": "xxxx",
}
```

json 参数字段说明

参数名	类型	必选	说明
pageNo	String	是	分页号
pageSize	String	是	分页数量
mediaPreparationNo	String	是	培养基配制信息编号
mediaPreparationName	String	是	培养基配制信息名称
mediaPreparationType	String	是	培养基配制信息类型
createTime	Date	是	创建时间

返回示例

```
{
  "code": 0,
  "data": {},
}
```

```
    "flag": true,
    "msg": " ",
  }
```

返回参数说明

字段名称	字段类型	注释说明
flag	boolean	异常标识
code	int	错误码
msg	string	错误信息
data	Object	返回数据

3. 1. 4 培养基配制信息创建接口

请求路径

/common/addMediaPreparation

请求方式 POST

json 参数

```
{
  "mediaPreparationNo": "***",
  "mediaPreparationName": "***",
  "mediaPreparationType": "***",
  "mediaPreparationStatus": "***",
  "createTime": "***",
}
```

json 参数字段说明

参数名	类型	必选	说明
-----	----	----	----

mediaPreparationNo	String	是	培养基配制信息编号
mediaPreparationName	String	是	培养基配制信息名称
mediaPreparationType	String	是	培养基配制信息类型
mediaPreparationStatus	String	是	培养基配制信息状态
createTime	String	是	创建时间

返回示例

```
{
  "code": 0,
  "data": {},
  "flag": true,
  "msg": "",
}
```

返回参数说明

字段名称	字段类型	注释说明
flag	boolean	异常标识
code	int	错误码
msg	string	错误信息
data	Object	返回数据

3.2 系统用户管理

3.2.1 登录接口

请求路径

/common/user/login

请求方式 POST

json 参数

```
{
  "username": 1,
  "password": "123456",
  "address": "xxxx"
}
```

json 参数字段说明

参数名	类型	必选	说明
password	String	是	密码
username	String	是	用户名
address	String	是	用户地址

返回示例

```
{
  "code": 0,
  "data": {},
  "flag": true,
  "msg": " ",
}
```

返回参数说明

字段名称	字段类型	注释说明
flag	boolean	异常标识
code	int	错误码
msg	string	错误信息
data	Object	返回数据



3.2.2 注销接口

请求路径

/common/user/layout

请求方式 POST

json 参数

```
{
  "username": 1,
  "password": "123456"
}
```

json 参数字段说明

参数名	类型	必选	说明
password	String	是	密码
username	String	是	用户名

返回示例

```
{
  "code": 0,
  "data": {},
  "flag": true,
  "msg": " ",
}
```

返回参数说明

字段名称	字段类型	注释说明
------	------	------

flag	boolean	异常标识
code	int	错误码
msg	string	错误信息
data	Object	返回数据

3.2.3 注册接口

请求路径

/common/user/register

请求方式 POST

json 参数

```
{
  "username": 1,
  "password": "123456",
  "mobile": "xxxx",
  "address": "xxxx",
  "Role": "xxxx",
  "Picture": "xxxx",
  "File": "xxxx",
}
```

json 参数字段说明

参数名	类型	必选	说明
password	String	是	密码
username	String	是	用户名
mobile	String	是	手机号

Role	String	是	角色
Picture	String	是	照片
address	String	是	区域
File	String	是	附件

返回示例

```
{
  "code": 0,
  "data": {},
  "flag": true,
  "msg": " ",
}
```

返回参数说明

字段名称	字段类型	注释说明
flag	boolean	异常标识
code	int	错误码
msg	string	错误信息
data	Object	返回数据

3. 2. 4 信息修改接口

请求路径

/common/user/changeUserInfo

请求方式 POST

json 参数

```
{
```

```
    "username": 1,
    "password": "123456",
    "mobile": "xxxx",
    "address": "xxxx",
    "Role": "xxxx",
    "Picture": "xxxx",
    "File": "xxxx",
  }
```

json 参数字段说明

参数名	类型	必选	说明
password	String	是	密码
username	String	是	用户名
Sex	String	否	性别
Role	String	否	角色
Picture	String	否	照片
password	String	否	密码
address	String	否	区域
File	String	否	附件
Department	String	否	部门

返回示例

```
{
  "code": 0,
  "data": {},
  "flag": true,
  "msg": " ",
}
```

返回参数说明

字段名称	字段类型	注释说明
flag	boolean	异常标识
code	int	错误码
msg	string	错误信息
data	Object	返回数据

3.3 权限管理

3.3.1 权限新增接口

请求路径

/common/security/addPermission

请求方式 POST

json 参数

```
{
  "username": 1,
  "Wechatnumber": "123456",
  "Sex": "xxxx",
  "Role": "xxxx"
}
```

json 参数字段说明

参数名	类型	必选	说明
username	String	是	用户名
Wechatnumber	String	是	微信号

Sex	String	是	性别
Role	String	是	角色

返回示例

```
{
  "code": 0,
  "data": {},
  "flag": true,
  "msg": "",
}
```

返回参数说明

字段名称	字段类型	注释说明
flag	boolean	异常标识
code	int	错误码
msg	string	错误信息
data	Object	返回数据

3.3.2 权限删除接口

请求路径

/common/security/deletePermission

请求方式 POST

json 参数

```
{
  "username": 1,
  "Wechatnumber": "123456",
}
```

```
    "Sex": "xxxx",
    "Role": "xxxx"
  }
```

json 参数字段说明

参数名	类型	必选	说明
username	String	是	用户名
Wechatnumber	String	是	微信号
Sex	String	是	性别
Role	String	是	角色

返回示例

```
{
  "code": 0,
  "data": {},
  "flag": true,
  "msg": " ",
}
```

返回参数说明

字段名称	字段类型	注释说明
flag	boolean	异常标识
code	int	错误码
msg	string	错误信息
data	Object	返回数据

### 3.3.3 权限列表查询接口

请求路径

/common/security/queryPermissionList

请求方式 POST

json 参数

```
{
  "username": 1,
  "Wechatnumber": "123456",
  "Sex": "XXXX",
  "Role": "XXXX",
  "isActivate": "XXXX"
  "Permissionlist": "XXXX"
}
```

json 参数字段说明

参数名	类型	必选	说明
username	String	是	用户名
Wechatnumber	String	是	微信号
Sex	String	是	性别
Role	String	是	角色
isActivate	String	是	是否激活
Permissionlist	String	是	权限列表

返回示例

```
{
  "code": 0,
  "data": {},
  "flag": true,
```



```
    "msg": "",
  }
```

返回参数说明

字段名称	字段类型	注释说明
flag	boolean	异常标识
code	int	错误码
msg	string	错误信息
data	Object	返回数据

3.3.4 管理员权限赋予接口

请求路径

/common/security/adminPermission

请求方式 POST

json 参数

```
{
  "username": 1,
  "Wechatnumber": "123456",
  "Sex": "xxxx",
  "Role": "xxxx"
}
```

json 参数字段说明

参数名	类型	必选	说明
username	String	是	用户名

Wechatnumber	String	是	微信号
Sex	String	是	性别
Role	String	是	角色

返回示例

```
{
  "code": 0,
  "data": {},
  "flag": true,
  "msg": " ",
}
```

返回参数说明

字段名称	字段类型	注释说明
flag	boolean	异常标识
code	int	错误码
msg	string	错误信息
data	Object	返回数据

## 第四章 表设计

### 4.1 菌株信息表

简要描述：

- 菌株信息

字段	类型	空	默认	注释
Id	String	true	Null	主键 id
strainInfoNo	String	true	Null	菌株信息编号
strainInfoName	String	true	Null	菌株信息名称
strainInfoType	String	true	Null	菌株信息类型
strainInfoStatus	String	true	Null	菌株信息状态
createTime	String	true	Null	创建时间
creator	String	true	Null	创建人

- 备注：无

### 4.2 培养基配制信息表

简要描述：

- 培养基配制信息

字段	类型	空	默认	注释
Id	String	false	Null	主键 id

mediaPreparationNo	String	true	Null	培养基配制信息编号
mediaPreparationName	String	true	Null	培养基配制信息名称
mediaPreparationType	String	false	Null	培养基配制信息类型
mediaPreparationStatus	String	true	Null	培养基配制信息状态
createTime	String	false	Null	创建时间

备注：无

4.3 系统用户表

简要描述：

- 描述储存系统用户信息

字段	类型	空	默认	注释
password	String	false	Null	密码
username	String	false	Null	用户名
mobile	String	false	Null	手机号
gender	String	true	Null	性别
role	String	true	Null	角色
picture	String	true	Null	照片
isActivate	String	true	Null	是否激活
file	String	false	Null	附件

备注：无

## 第五章 非功能性设计

### 5.1 性能设计

数据库采用主从方式；

合理的应用程序架构：MVC

数据缓存：redis

搭建两套应用服务器，应用 SL[改成 SLB]B 负载均衡，提高系统性能，应对多用户、高并发业务场景；

平均响应时间 小于 100ms；最大响应时间不超过 500ms

资源利用率：cpu 平均利用率 30%，最高不超过 80%； 内存利用率平均 45%，最高不超过 80%；

请求并发量 $\approx$ 90

### 5.2 安全性设计

定义基于角色的权限管理机制，不同角色培养基配制信息访问权限内数据

服务器只开放需要的端口

数据库操作使用预编译预计，防止 SQL 注入[增加 XSS 注入、CSRF 注入]、

非登录用户不允许进入系统

接口添加签名校验

### 5.3 其他非功能性设计

易用性、界面友好性：

用户交互界面绘图考虑不同层面用户对信息使用的不同需求，要求页面美观大方，操作简便，界面术语名称一致。

### **兼容性**

**操作系统：**系统服务端及选用的数据库、中间件，应兼容 Linux 不同发行版或 Windows 不同版本；

**产品版本：**系统升级可兼容以前版本所有数据，不会造成数据丢失；

**浏览器：**应兼容主流浏览器&版本，系统监测报告中应明确体现浏览器兼容列表。

### **可移植性**

提高代码可移植性，降低代码重复开发工作量。