

友禾车间制造执行系统 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 /service/
```

```
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 service
```

查看进程是否存在：

如果不存在，进入/service/

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

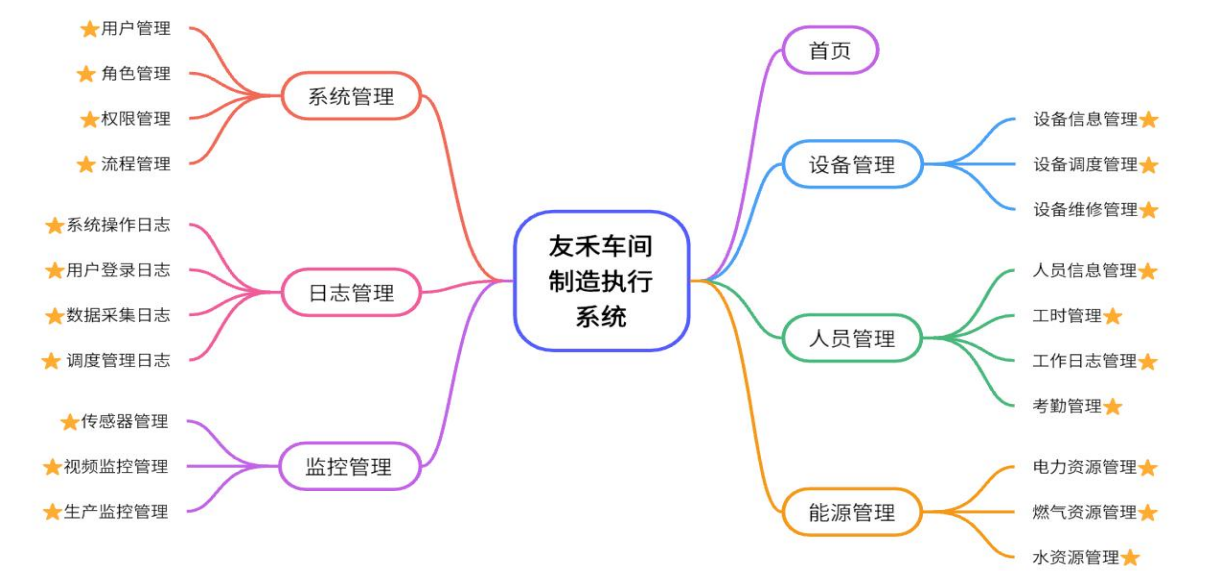
客户端: 终端

1.4 系统概述

友禾车间制造执行系统是一种在车间中应用的 ERP 软件, 旨在支持生产流程的执行和控制, 以及对生产流程进行监测和优化。友禾车间制造执行系统通过实时数据采集和信息化技术支持, 实现了生产的自动化、信息化和智能化, 从而提高了生产效率、优化了生产流程、降低了生产成本。

第二章 总体设计

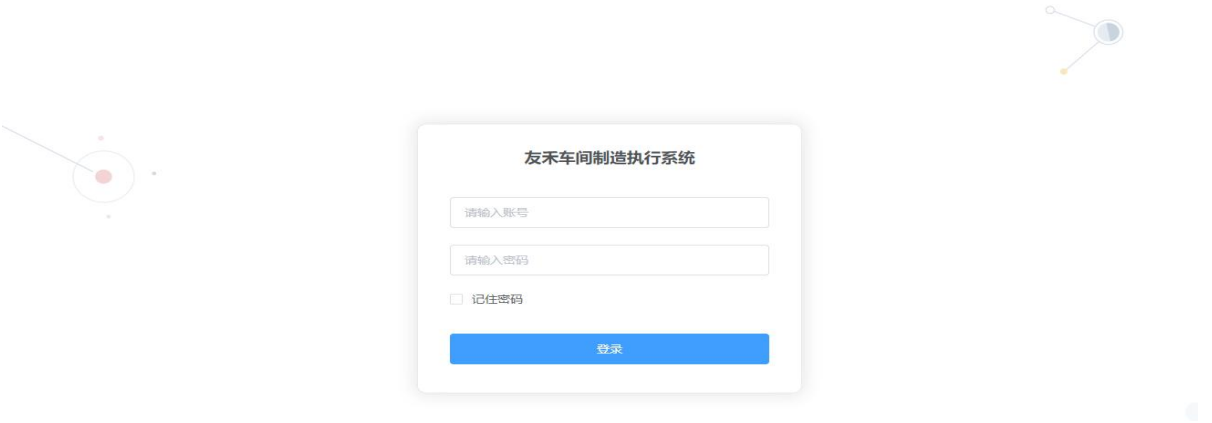
2.1 功能结构图



2.2 界面效果图

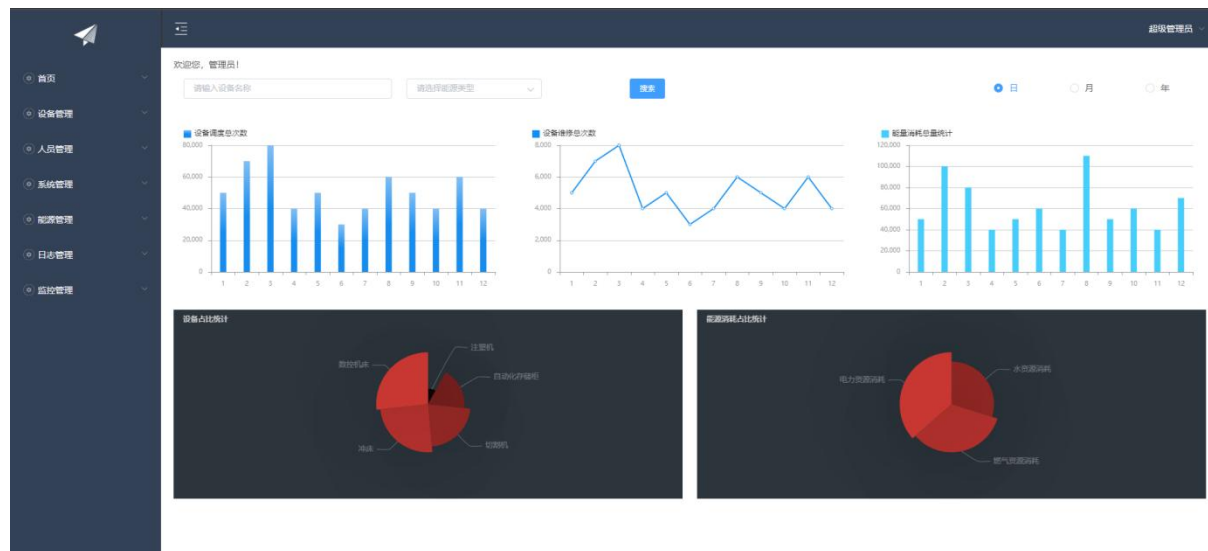
2.2.1 登录系统

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，进入友禾车间制造执行系统。



2.2.2 系统首页

点击【登录】按钮，进入系统首页，可以查看数据总览：设备调度总次数、设备维修总次数、能源消耗总量统计、设备占比统计、能源消耗占比统计。



2.2.3 设备管理

点击【设备管理】菜单，再点击设备信息管理，展示设备信息管理列表，展示信息：设备名称、设备类型、设备状态、创建时间、创建人。

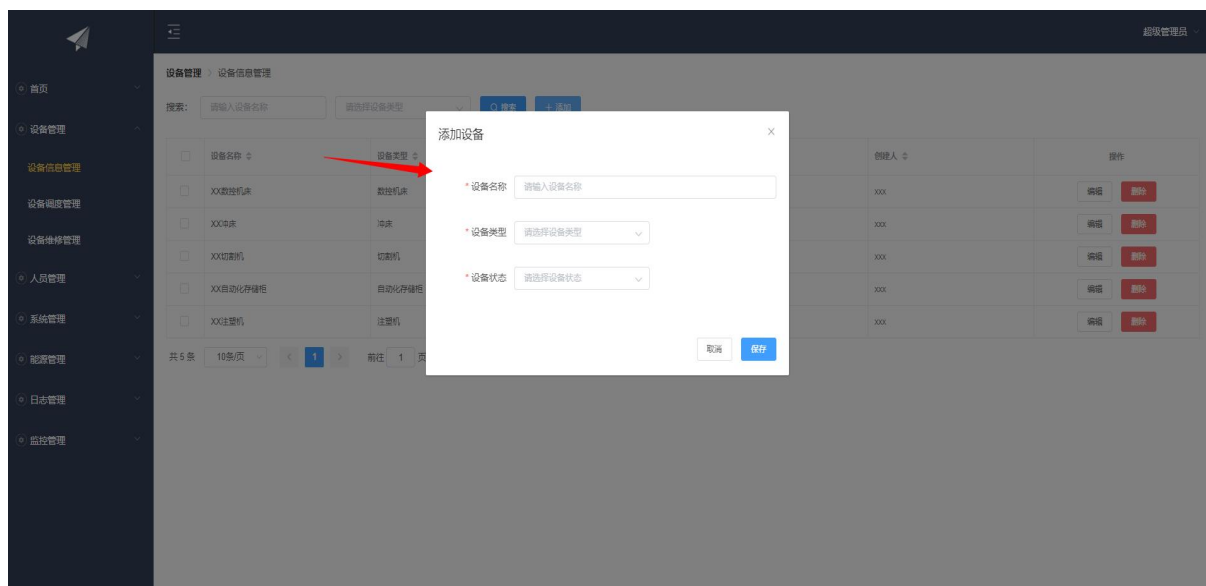
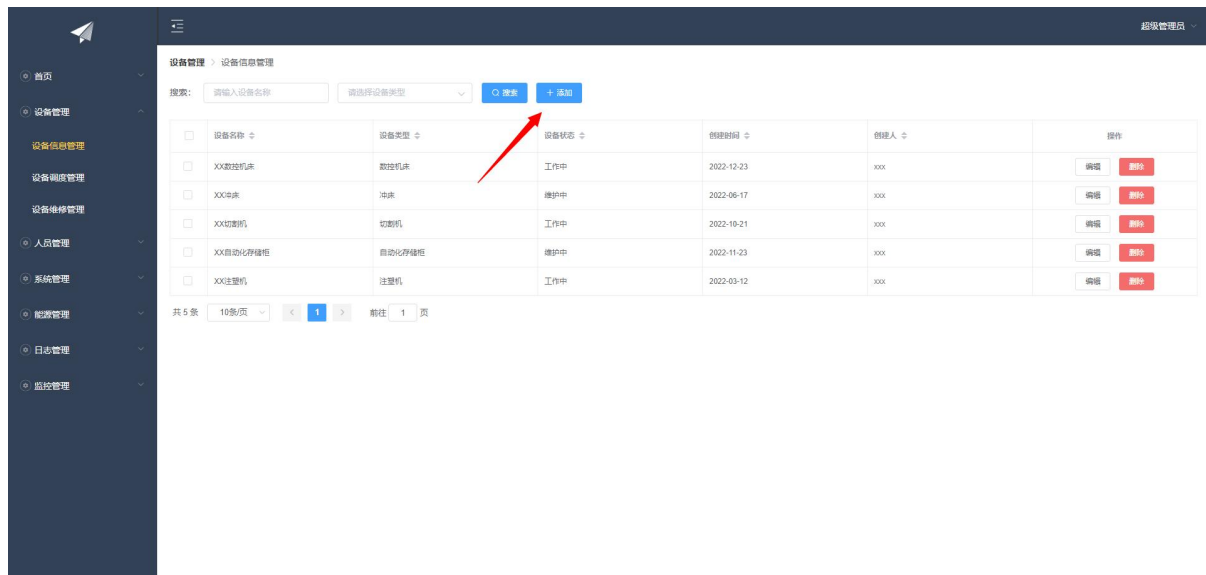
The screenshot shows the '设备信息管理' (Equipment Information Management) page. The left sidebar has a red arrow pointing to the '设备信息管理' menu item. The main content area displays a table of equipment information.

<input type="checkbox"/>	设备名称	设备类型	设备状态	创建时间	创建人	操作
<input type="checkbox"/>	XX数控机床	数控机床	工作中	2022-12-23	xxx	<button>编辑</button> <button>删除</button>
<input type="checkbox"/>	XX冲床	冲床	维护中	2022-06-17	xxx	<button>编辑</button> <button>删除</button>
<input type="checkbox"/>	XX切磨床	切磨床	工作中	2022-10-21	xxx	<button>编辑</button> <button>删除</button>
<input type="checkbox"/>	XX自动化存储柜	自动化存储柜	维护中	2022-11-23	xxx	<button>编辑</button> <button>删除</button>
<input type="checkbox"/>	XX注塑机	注塑机	工作中	2022-03-12	xxx	<button>编辑</button> <button>删除</button>

共 5 条 | 10条/页 | < 1 > | 前往 1 页

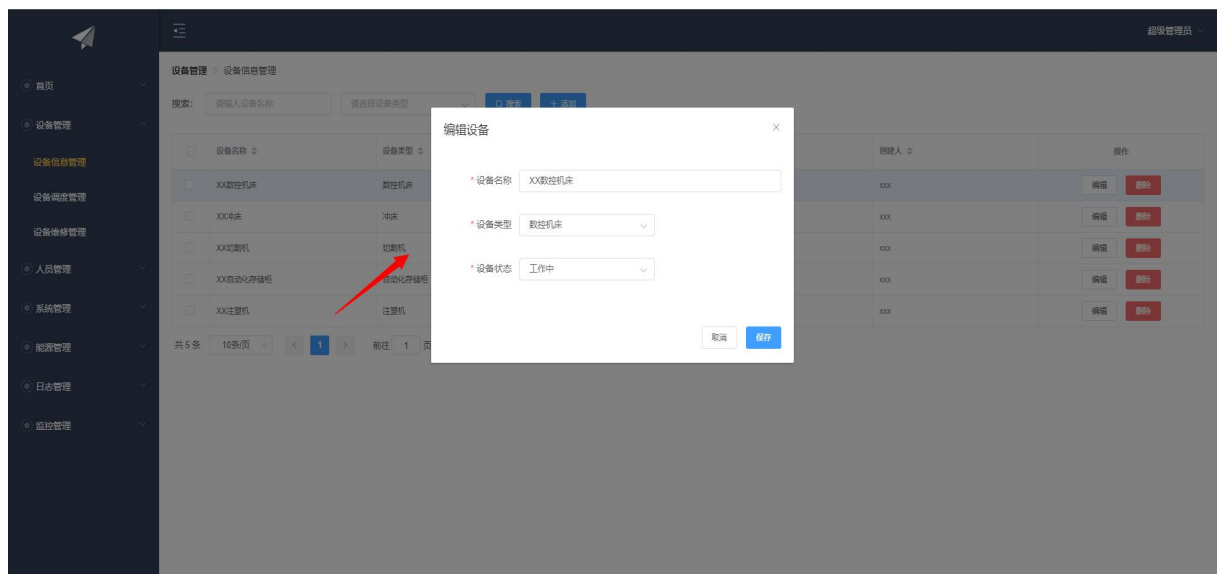
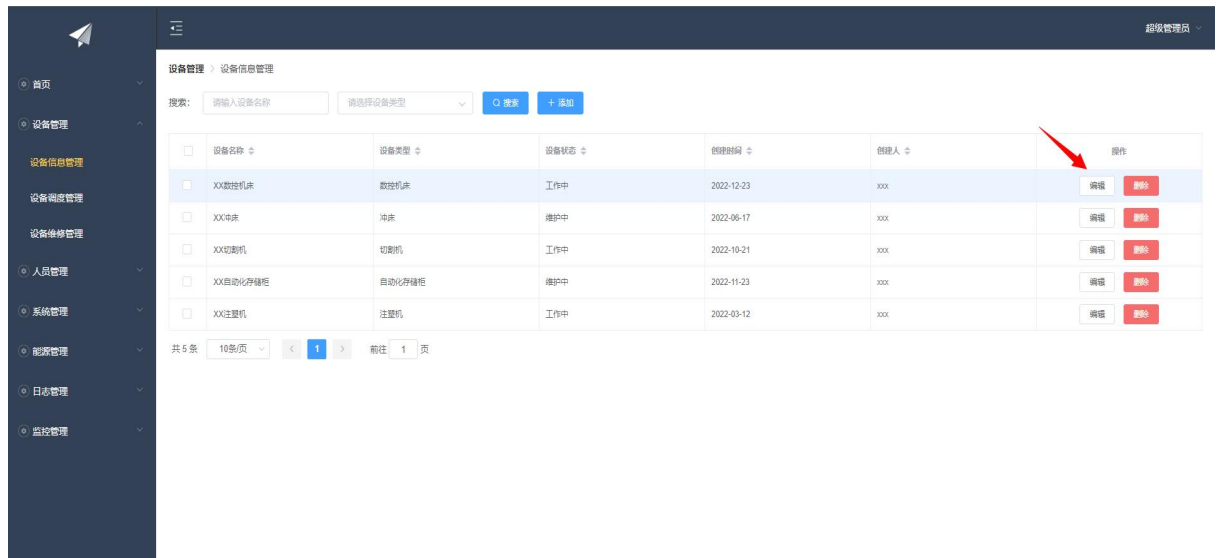
2.2.4 设备添加

点击添加按钮，弹出设备添加弹窗，填写信息添加即可。



2.3.5 设备编辑

点击右侧编辑按钮，弹出设备编辑弹窗，填写信息编辑即可。



2.2.6 人员管理

点击【人员管理】菜单，再点击人员信息管理进入列表页。展示信息：姓名、地址、性别、年龄、工作状态、创建时间。

人员管理 > 人员信息管理

搜索: 请输入姓名 请选择性别 请选择工作状态 搜索 添加

	姓名	地址	性别	年龄	工作状态	创建时间	操作
1	xxx	安徽合肥xxx	男	50	休息中	2020-08-12	编辑 删除
2	xxx	安徽芜湖xxx	男	50	休息中	2022-10-12	编辑 删除
3	xxx	安徽合肥xxx	男	55	休息中	2022-01-12	编辑 删除
4	xxx	安徽合肥xxx	女	40	工作中	2022-10-12	编辑 删除
5	xxx	安徽滁州xxx	男	50	休息中	2021-10-12	编辑 删除
6	xxx	安徽合肥xxx	男	44	休息中	2022-10-12	编辑 删除
7	xxx	安徽合肥xxx	男	50	工作中	2022-10-12	编辑 删除
8	xxx	安徽合肥xxx	男	50	工作中	2022-10-12	编辑 删除
9	xxx	安徽合肥xxx	男	50	工作中	2022-10-12	编辑 删除
10	xxx	安徽合肥xxx	男	50	工作中	2022-10-12	编辑 删除

共 23 条 10条/页 < 1 2 3 > 前往 1 页

2.2.7 人员添加

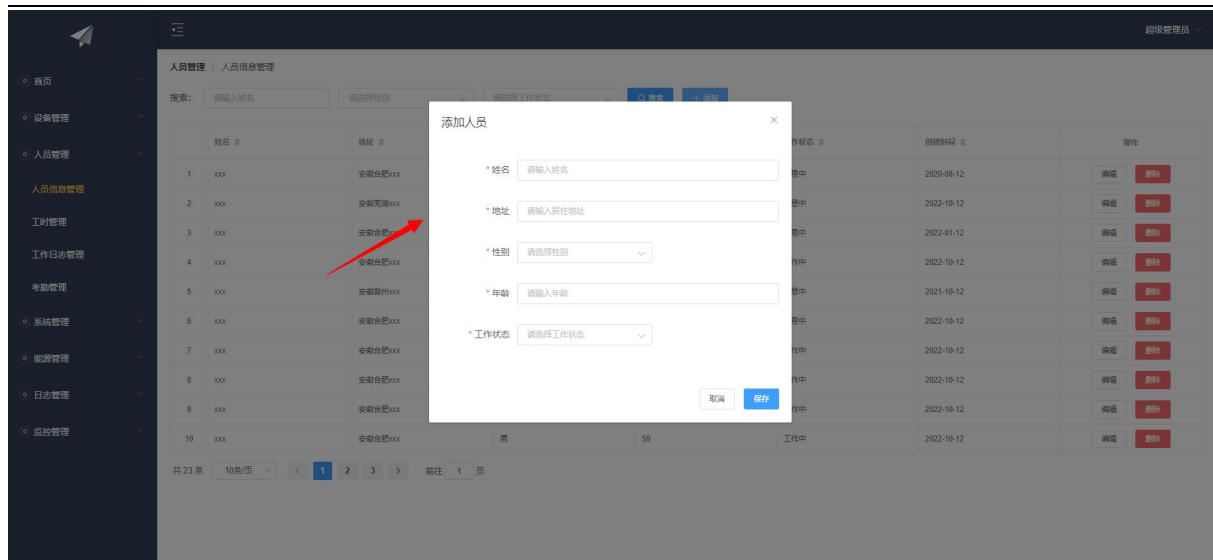
点击添加按钮，弹出人员添加弹窗，填写信息添加即可。

人员管理 > 人员信息管理

搜索: 请输入姓名 请选择性别 请选择工作状态 搜索 添加

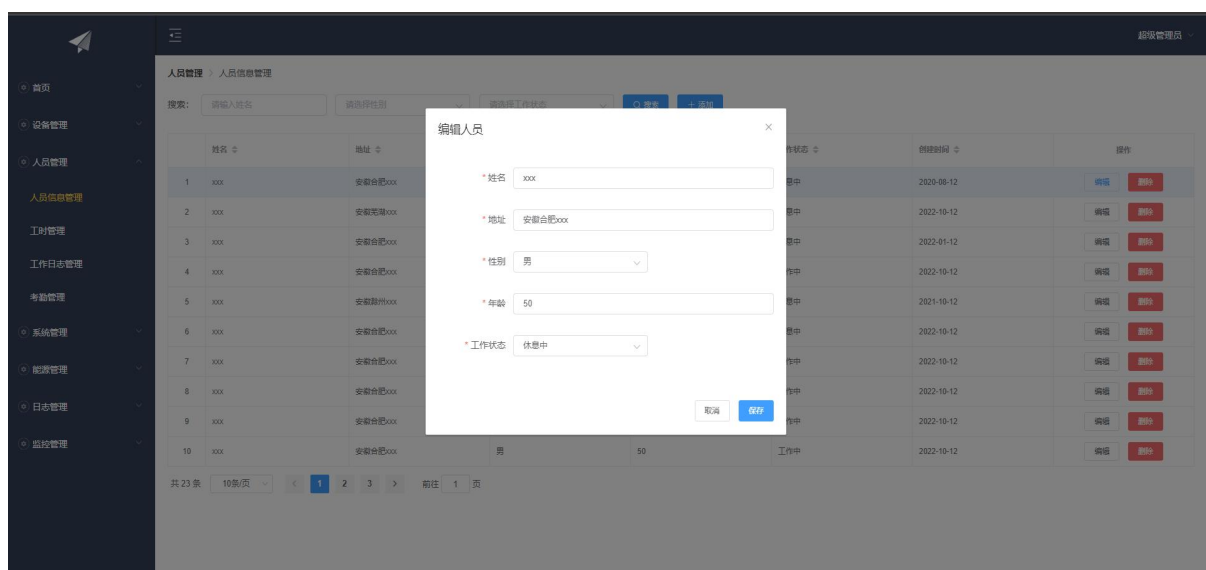
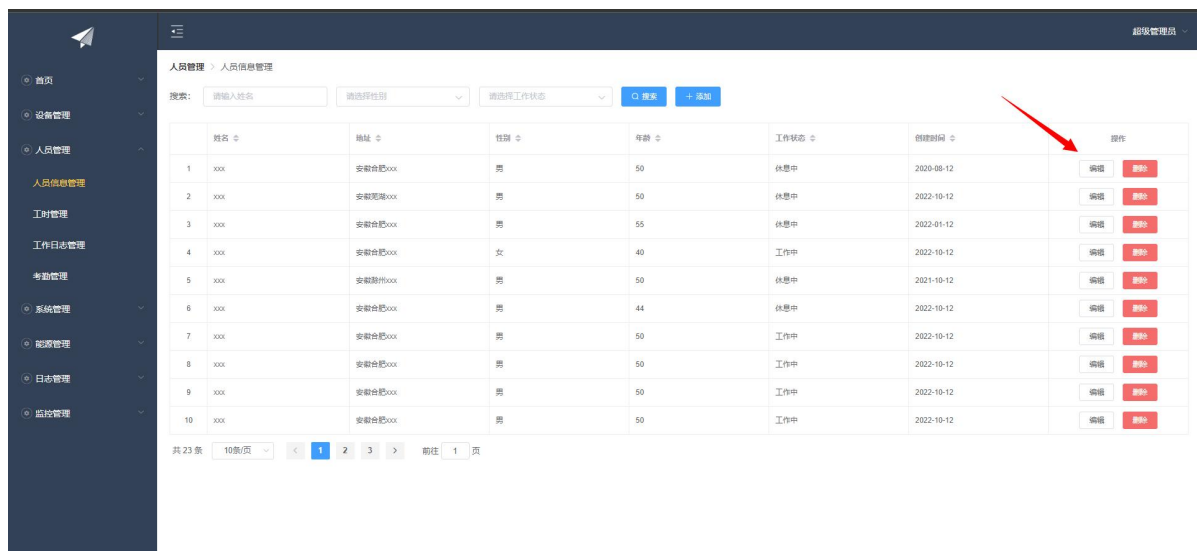
	姓名	地址	性别	年龄	工作状态	创建时间	操作
1	xxx	安徽合肥xxx	男	50	休息中	2020-08-12	编辑 删除
2	xxx	安徽芜湖xxx	男	50	休息中	2022-10-12	编辑 删除
3	xxx	安徽合肥xxx	男	55	休息中	2022-01-12	编辑 删除
4	xxx	安徽合肥xxx	女	40	工作中	2022-10-12	编辑 删除
5	xxx	安徽滁州xxx	男	50	休息中	2021-10-12	编辑 删除
6	xxx	安徽合肥xxx	男	44	休息中	2022-10-12	编辑 删除
7	xxx	安徽合肥xxx	男	50	工作中	2022-10-12	编辑 删除
8	xxx	安徽合肥xxx	男	50	工作中	2022-10-12	编辑 删除
9	xxx	安徽合肥xxx	男	50	工作中	2022-10-12	编辑 删除
10	xxx	安徽合肥xxx	男	50	工作中	2022-10-12	编辑 删除

共 23 条 10条/页 < 1 2 3 > 前往 1 页



2.2.8 人员编辑

点击右侧编辑按钮，弹出人员编辑弹窗，填写信息编辑即可。



第三章 接口设计

3.1 设备和人员管理

3.1.1 设备信息管理接口

请求路径

/workshop/deviceManagementList

请求方式 POST

json 参数

```
{
  "pageNo": 1,
  "pageSize": 10,
  "deviceType": "xxxx",
  "createTime": "xxxx",
  "deviceStatus": "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
deviceType	String	是	设备类型
createTime	String	是	创建时间
deviceStatus	Date	是	设备状态
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 人员信息管理接口

请求路径

/workshop/workshopWorkerList

请求方式 POST

json 参数

```
{
  "pageNo": 1,
  "pageSize": 10,
  "name": "xxx",
  "address": "xxx",
  "gender": "xxxx",
  "age": "xxxx",
}
```

```
        "workStatus": "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
name	String	是	姓名
address	String	是	居住地址
gender	String	是	性别
age	Date	是	年龄
workStatus	String	是	工作状态
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

3.1.3 设备创建接口

请求路径

/workshop/addDevice

请求方式 POST

json 参数

```
{
  "deviceName": "***",
  "deviceType": "***",
  "createTime": "***",
  "deviceStatus": "***",
  "creator": "***",
}
```

json 参数字段说明

参数名	类型	必选	说明
deviceName	String	是	设备名称
deviceType	String	是	设备类型
createTime	String	是	新建时间
deviceStatus	String	是	设备状态
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

3.1.4 人员创建接口

请求路径

/workshop/addWorkshopWorker

请求方式 POST

json 参数

```
{
  "name": "***",
  "address": "***",
  "gender": "***",
  "age": "***",
  "workStatus": "***",
  "createTime": "***",
}
```

json 参数字段说明

参数名	类型	必选	说明
name	String	是	姓名
addres	String	是	地址
gender	String	是	性别
age	String	是	年龄
workStatus	String	是	工作状态
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 用户管理

3. 2. 1 登录接口

请求路径

/workshop/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

/workshop/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

/workshop/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

/workshop/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

/workshop/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

/workshop/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

/workshop/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

/workshop/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 设备信息表

简要描述:

- 设备信息

字段	类型	空	默认	注释
Id	String	true	Null	设备 id
deviceName	String	true	Null	设备名称
deviceType	String	true	Null	设备类型
deviceStatus	String	true	Null	设备状态
createTime	String	true	Null	创建时间
creator	String	true	Null	创建人

- 备注: 无

4.2 人员信息表

简要描述:

- 人员信息

字段	类型	空	默认	注释
Id	String	true	Null	人员 id
name	String	true	Null	人员姓名

address	String	true	Null	人员居住地址
gender	String	true	Null	人员性别
age	String	true	Null	人员年龄
workStatus	String	true	Null	人员工作状态
createTime	String	true	Null	创建时间

备注：无

4.3 用户管理表

简要描述：

- 描述储存用户信息

字段	类型	空	默认	注释
password	String	true	Null	密码
username	String	true	Null	用户名
mobile	String	true	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	true	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 ≈ 90

5.2 安全性设计

定义基于角色的权限管理机制，不同角色人员访问权限内数据

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。