

财务共享统计系统 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 /account/
```

```
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 account
```

查看进程是否存在：

如果不存在，进入/account/

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

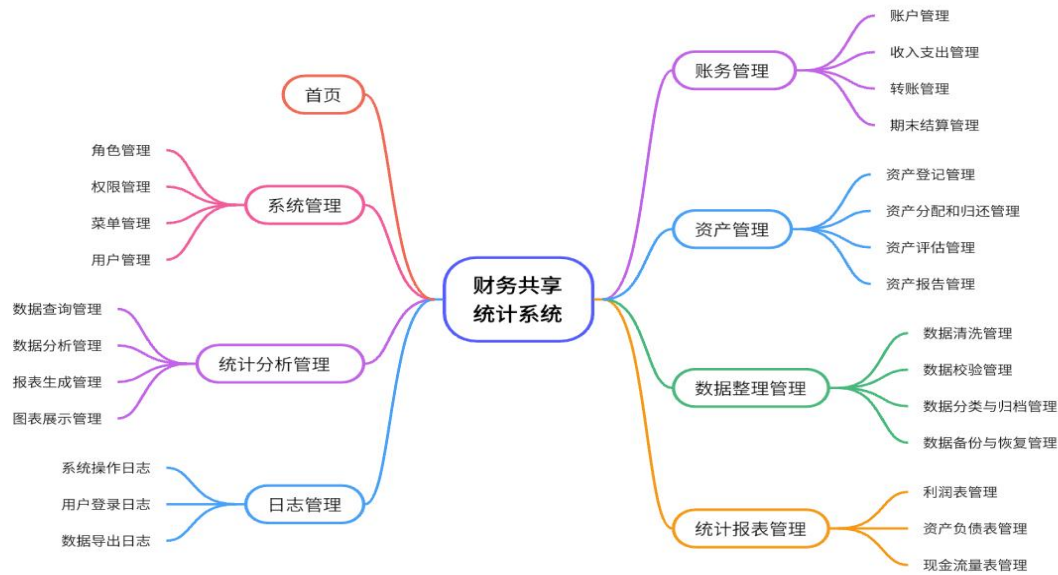
客户端: 终端

1.4 系统概述

财务共享统计系统可以帮助企业对财务数据进行统计和分析的系统。通过该系统,企业可以方便地收集、整理和分析财务数据,以便更好地了解企业的财务状况和经营表现。

第二章 总体设计

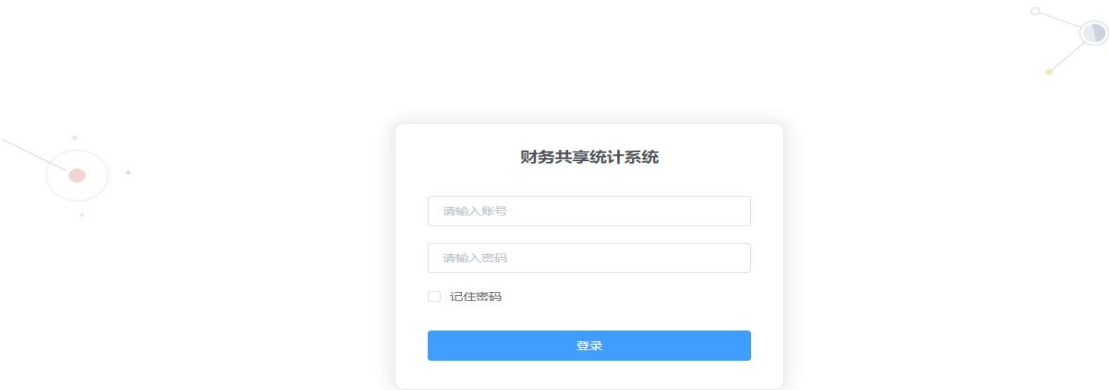
2.1 功能结构图



2.2 界面效果图

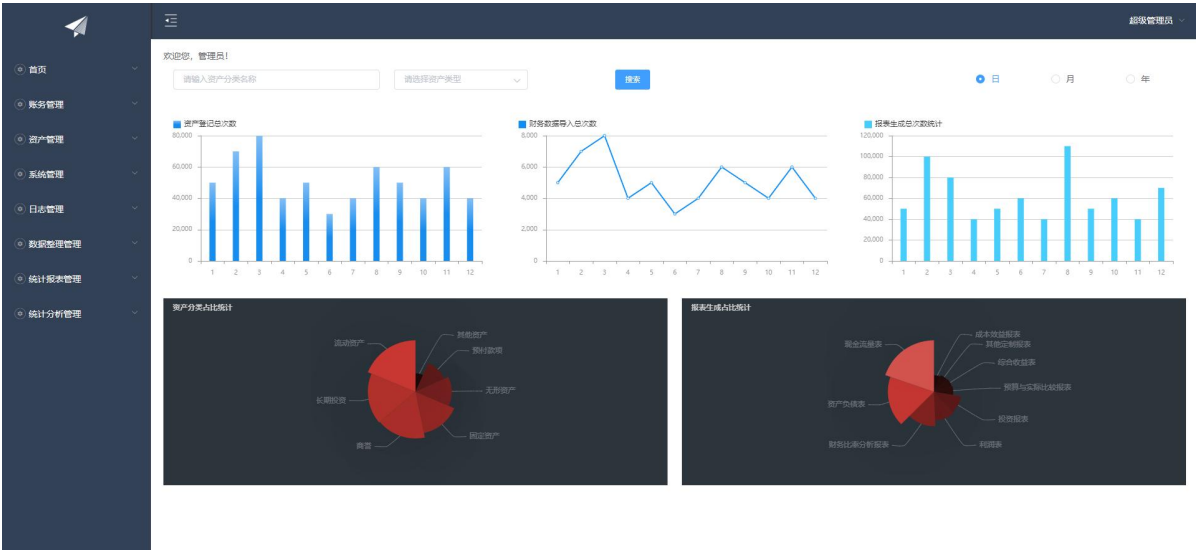
2.2.1 登录系统

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，进入财务共享统计系统。



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：政务服务总次数统计、政务服务流程处理总次数统计、支付结算统计、财务账户占比统计、能源消耗占比统计。



2.2.3 财务账户管理

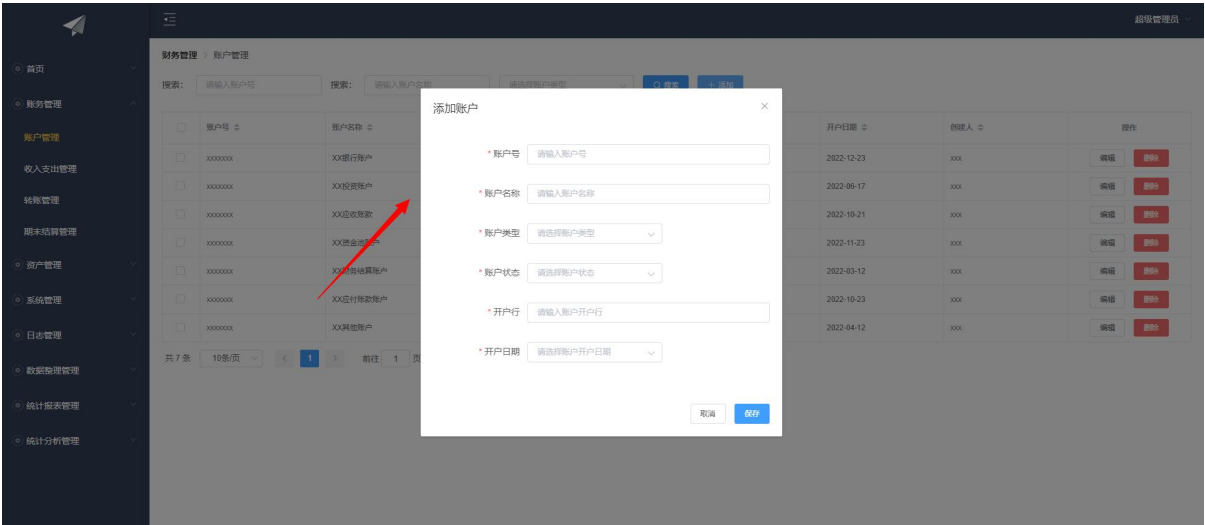
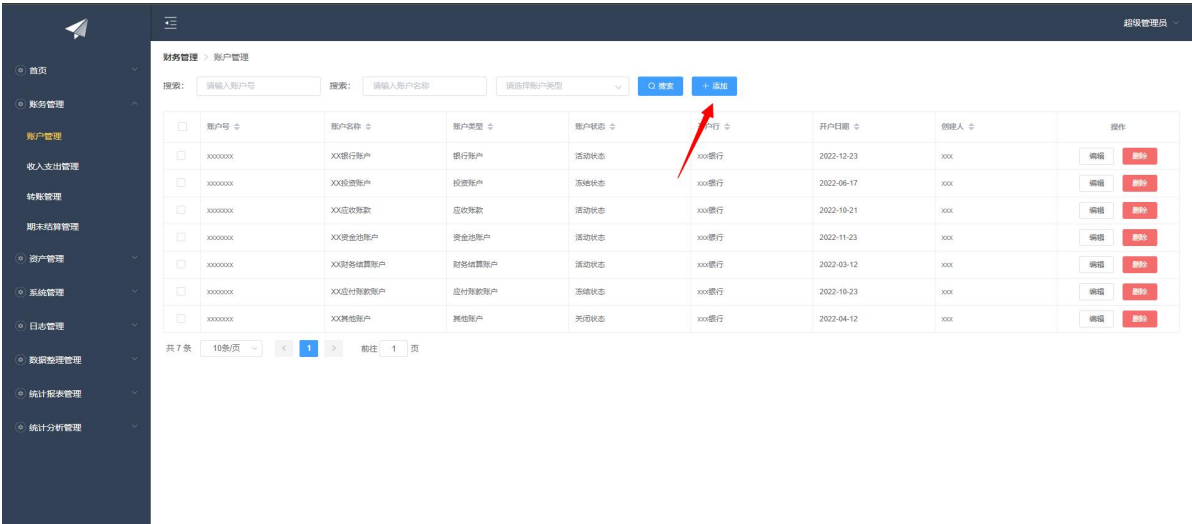
点击【财务管理】菜单，再点击账户管理，展示财务账户信息管理列表，展示信息：财务账户号、财务账户名称、财务账户类型、财务账户状态、开户行、开户时间、创建人。

The interface shows a list of financial accounts with the following data:

| 账户号 | 账户名称 | 账户类型 | 账户状态 | 开户行 | 开户日期 | 创建人 | 操作 |
|---------|----------|--------|------|-------|------------|-----|-------|
| XXXXXXX | XX银行账户 | 银行账户 | 活跃状态 | xxx银行 | 2022-12-23 | xxx | 编辑 删除 |
| XXXXXXX | XX投资账户 | 投资账户 | 冻结状态 | xxx银行 | 2022-06-17 | xxx | 编辑 删除 |
| XXXXXXX | XX应收账款 | 应收账款 | 活跃状态 | xxx银行 | 2022-10-21 | xxx | 编辑 删除 |
| XXXXXXX | XX资金池账户 | 资金池账户 | 活跃状态 | xxx银行 | 2022-11-23 | xxx | 编辑 删除 |
| XXXXXXX | XX财务结算账户 | 财务结算账户 | 活跃状态 | xxx银行 | 2022-03-12 | xxx | 编辑 删除 |
| XXXXXXX | XX应付账款账户 | 应付账款账户 | 冻结状态 | xxx银行 | 2022-10-23 | xxx | 编辑 删除 |
| XXXXXXX | XX其他账户 | 其他账户 | 关闭状态 | xxx银行 | 2022-04-12 | xxx | 编辑 删除 |

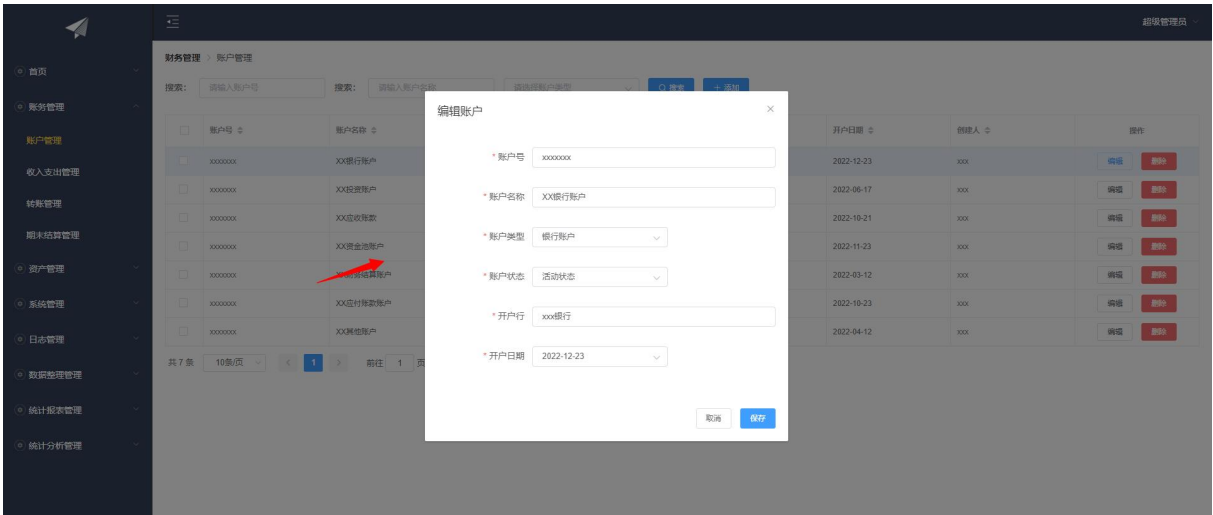
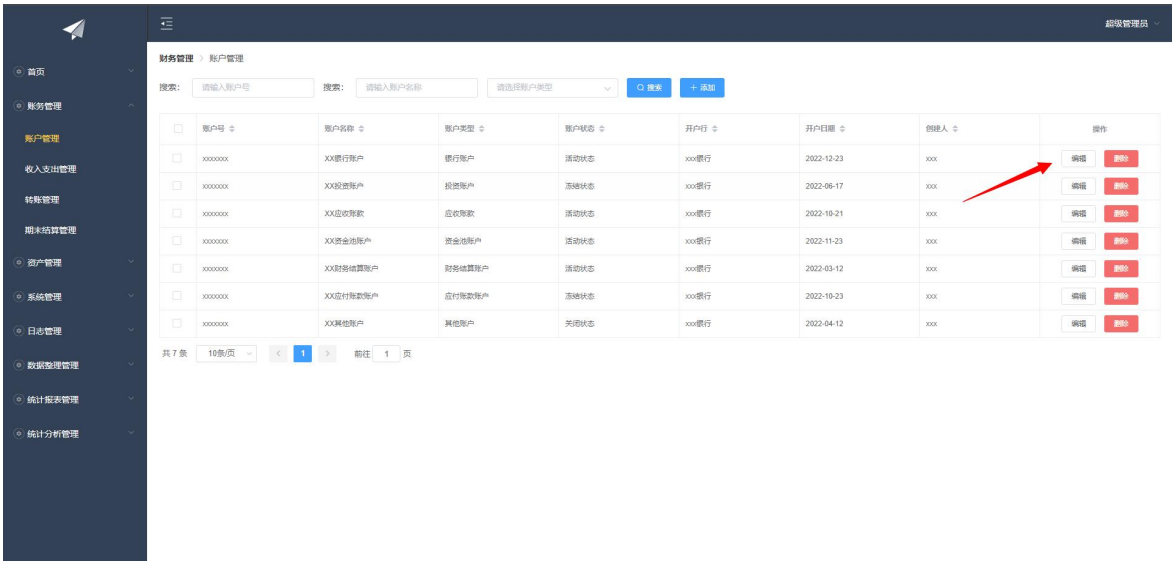
2.2.4 财务账户添加

点击添加按钮，弹出财务账户添加弹窗，填写信息添加即可。



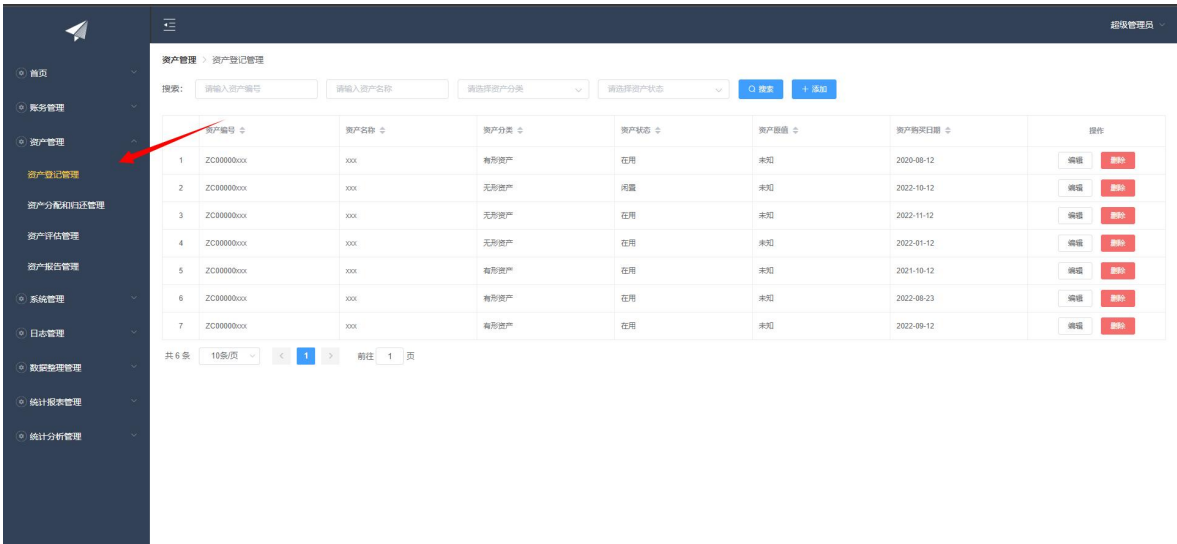
2.3.5 财务账户编辑

点击右侧编辑按钮，弹出财务账户编辑弹窗，填写信息编辑即可。



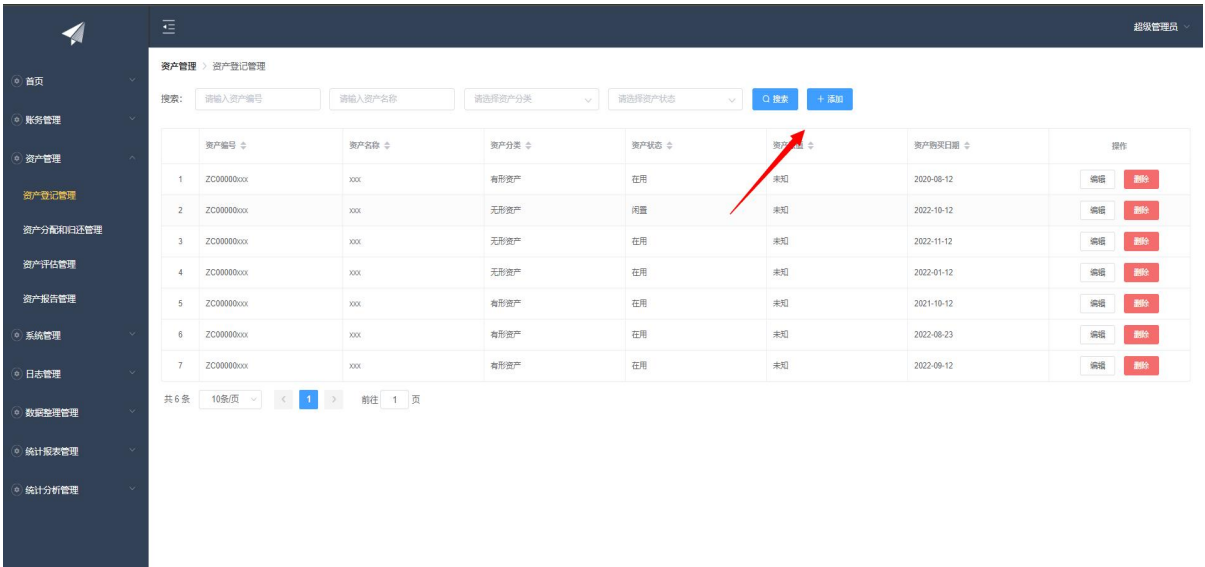
2.2.6 资产信息管理

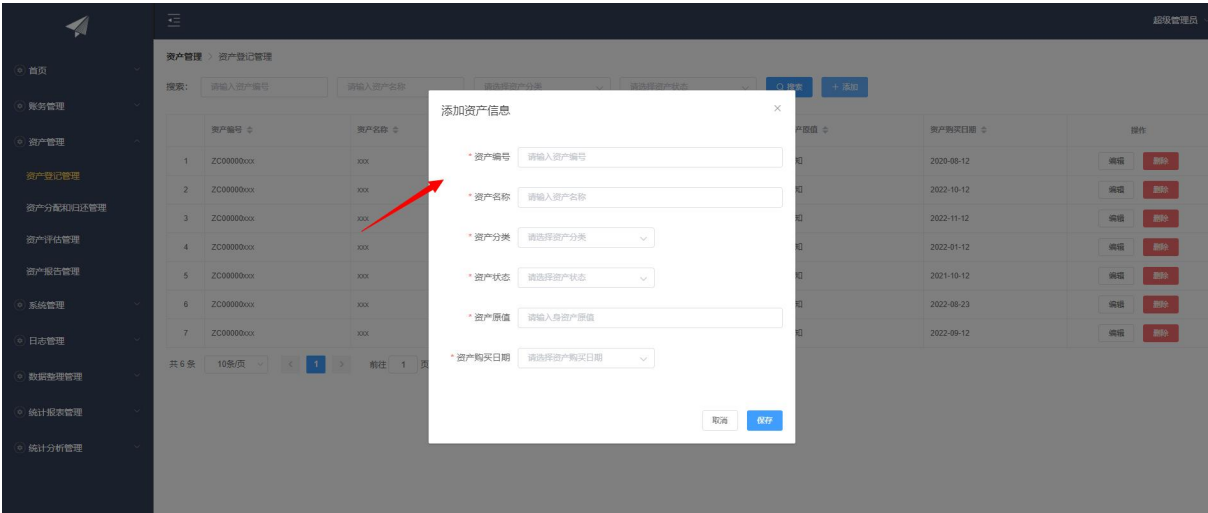
点击【资产管理】菜单，再点击资产登记管理进入列表页。展示信息：资产编号、资产名称、资产分类、资产状态、资产原值、资产购买日期。



2.2.7 资产信息添加

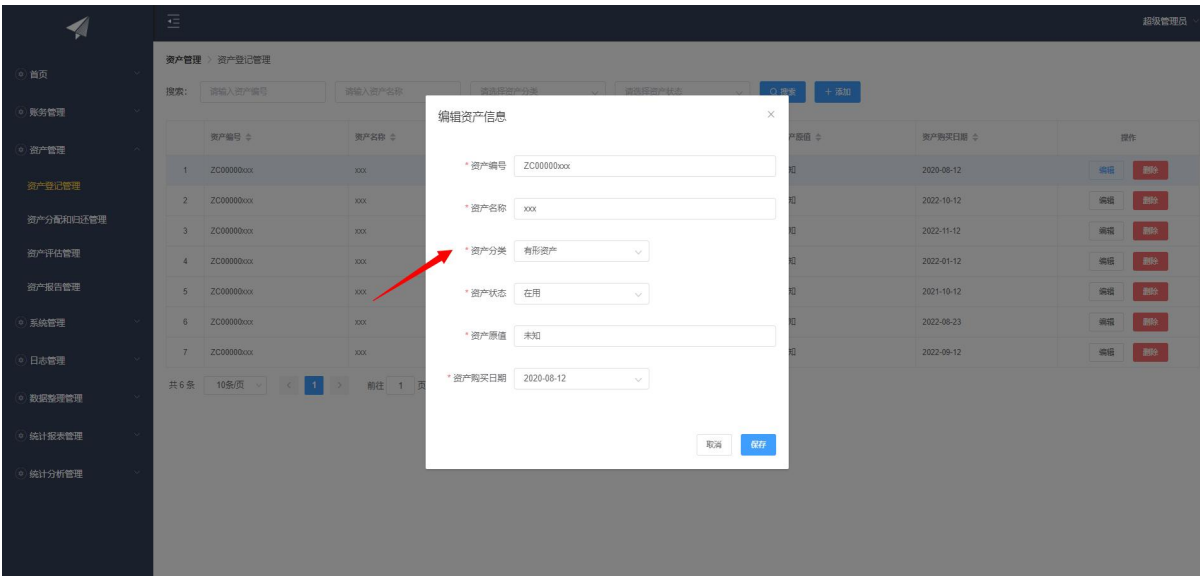
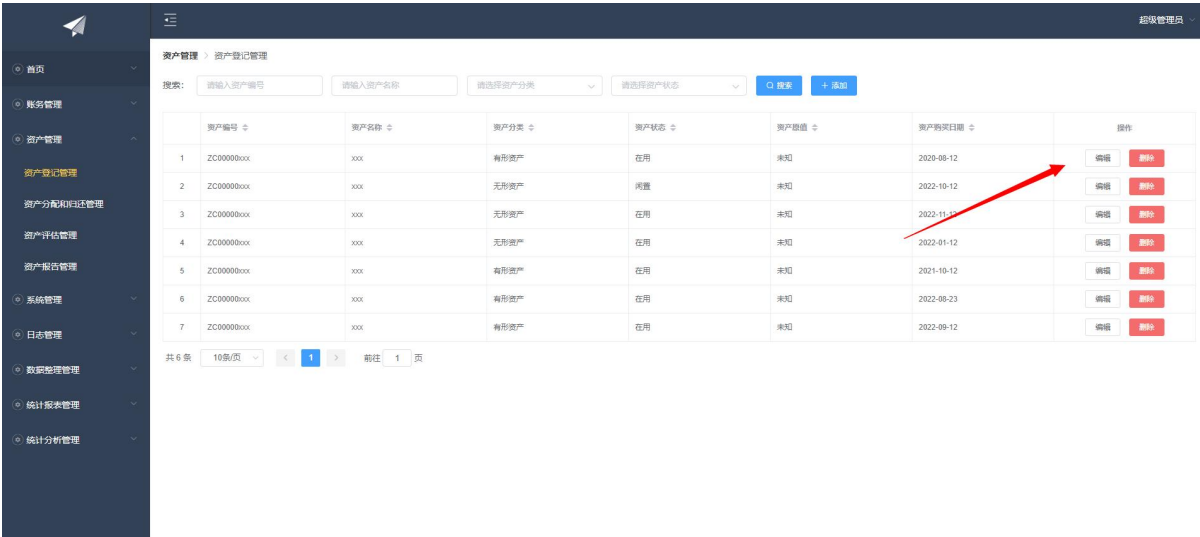
点击添加按钮，弹出资产信息添加弹窗，填写信息添加即可。





2.2.8 资产信息编辑

点击右侧编辑按钮，弹出资产信息编辑弹窗，填写信息编辑即可。



第三章 接口设计

3.1 财务账户和资产信息管理

3.1.1 财务账户信息管理接口

请求路径

/finance/accountManagementList

请求方式 POST

json 参数

```
{
  "pageNo": 1,
  "pageSize": 10,
  "accountNo": "xxxx",
  "accountName": "xxxx",
  "accountType": "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 |
| accountType | String | 是 | 财务账户类型 |
| createTime | String | 是 | 开户日期 |
| accountName | Date | 是 | 财务账户名称 |
| accountType | String | 是 | 财务账户类型 |

| | | | |
|---------|--------|---|-----|
| 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 财务账户创建接口

请求路径

/finance/addaccount

请求方式 POST

json 参数

```
{
  "accountNo": "xxxx",
  "accountName": "***",
  "accountBank": "xxxx",
  "accountType": "***",
}
```

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

json 参数字段说明

| 参数名 | 类型 | 必选 | 说明 |
|---------------|--------|----|--------|
| accountNo | String | 是 | 财务账户号 |
| accountName | String | 是 | 财务账户名称 |
| accountType | String | 是 | 财务账户类型 |
| createTime | String | 是 | 开户时间 |
| accountBank | String | 是 | 财务开户银行 |
| accountStatus | String | 是 | 财务账户状态 |
| 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 |

3.1.3 资产信息管理接口

请求路径

/finance/citizenList

请求方式 POST

json 参数

```
{
  "pageNo": 1,
  "pageSize": 10,
  "propertyNo": xxx,
  "propertyName": xxx,
  "propertyType": xxx,
  "propertyStatus": "xxx",
}
```

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 |
| propertyNo | String | 是 | 资产编号 |
| perpertyName | String | 是 | 资产名称 |
| propertyType | String | 是 | 资产分类 |
| propertyStatus | String | 是 | 资产状态 |

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 资产信息创建接口

请求路径

/finance/addCitizen

请求方式 POST

json 参数

```
{
  "propertyNo": xxx,
  "propertyName": xxx,
  "propertyType": xxx,
  "propertyStatus": "xxx",
  "propertyValue": "xxx",
}
```

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

json 参数字段说明

| 参数名 | 类型 | 必选 | 说明 |
|----------------|--------|----|--------|
| propertyNo | String | 是 | 资产编号 |
| propertyName | String | 是 | 资产名字 |
| propertyType | String | 是 | 资产类型 |
| propertyStatus | String | 是 | 资产状态 |
| propertyValue | String | 是 | 资产原值 |
| createTime | String | 是 | 资产购买日期 |

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

/finance/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

/finance/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

/finance/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

/finance/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

/finance/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

/finance/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

/finance/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

/finance/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 财务账户信息表

简要描述:

- 财务账户信息

| 字段 | 类型 | 空 | 默认 | 注释 |
|---------------|--------|------|------|----------|
| Id | String | true | Null | 主键 id |
| accountNo | String | true | Null | 财务账户号 |
| accountName | String | true | Null | 财务账户名称 |
| accountType | String | true | Null | 财务账户类型 |
| accountStatus | String | true | Null | 财务账户状态 |
| accountBank | String | true | Null | 财务账户开户银行 |
| createTime | String | true | Null | 开户日期 |
| creator | String | true | Null | 创建人 |

- 备注：无

4.2 资产信息表

简要描述:

- 资产信息

| 字段 | 类型 | 空 | 默认 | 注释 |
|----|----|---|----|----|
|----|----|---|----|----|

| | | | | |
|----------------|--------|------|------|-------|
| Id | String | true | Null | 主键 id |
| propertyNo | String | true | Null | 资产编号 |
| propertyName | String | true | Null | 资产姓名 |
| propertyType | String | true | Null | 资产类型 |
| propertyStatus | String | true | Null | 资产状态 |
| propertyValue | String | true | Null | 资产原值 |
| createTime | String | true | Null | 创建时间 |

备注：无

4.3 系统用户表

简要描述：

- 描述储存系统员工信息

| 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，不同角色资产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。