

智慧展厅增强现实设计展示 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 /virtuallyProduct/
```

```
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 virtuallyProduct
```

查看进程是否存在：

如果不存在，进入/virtuallyProduct/

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

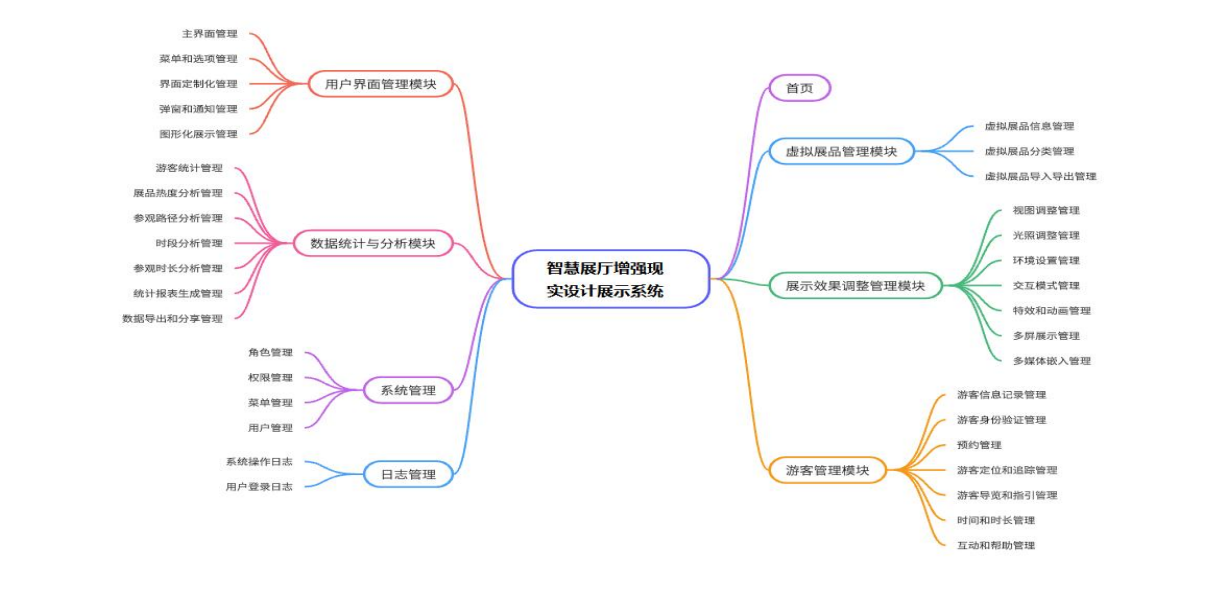
客户端: 终端

1.4 系统概述

智慧展厅增强现实设计展示系统是一种通过增强现实技术将虚拟的设计展示内容叠加在现实场景中的展示系统。它可以将物理展品与虚拟模型相结合, 为参观者提供一个全新的展览体验。

第二章 总体设计

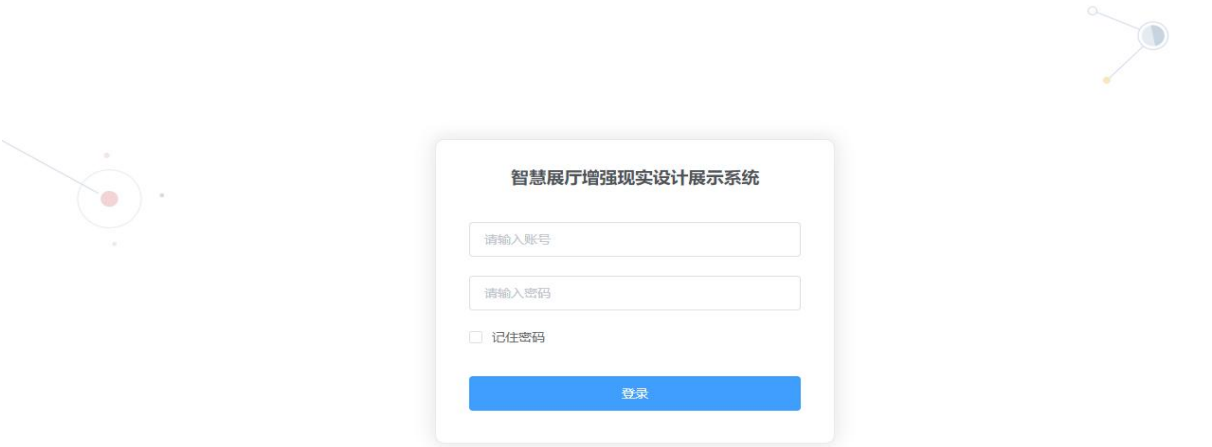
2.1 功能结构图



2.2 界面效果图

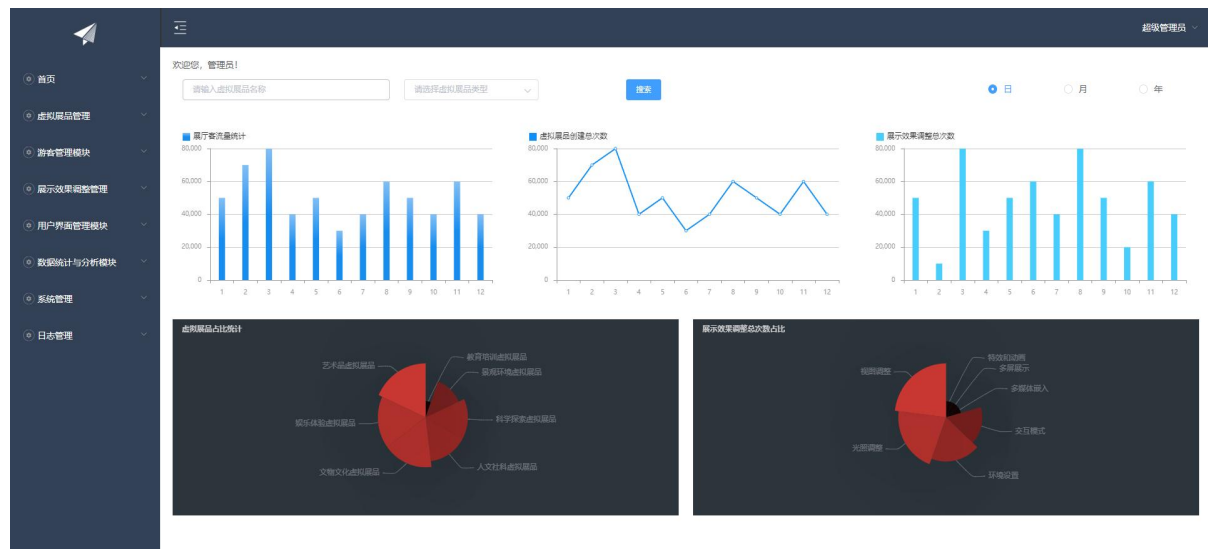
2.2.1 登录系统

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，进入智慧展厅增强现实设计展示系统。



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：展厅客流量统计、虚拟展品创建总次数、展示效果调整总次数、虚拟展品占比统计、展示效果调整总次数占比。



2.2.3 虚拟展品信息管理

点击【虚拟展品管理】菜单，再点击虚拟展品信息管理，展示虚拟展品信息管理列表，展示信息：虚拟展品信息编号、虚拟展品信息名称、虚拟展品信息类型、虚拟展品信息状态、创建时间、创建人。

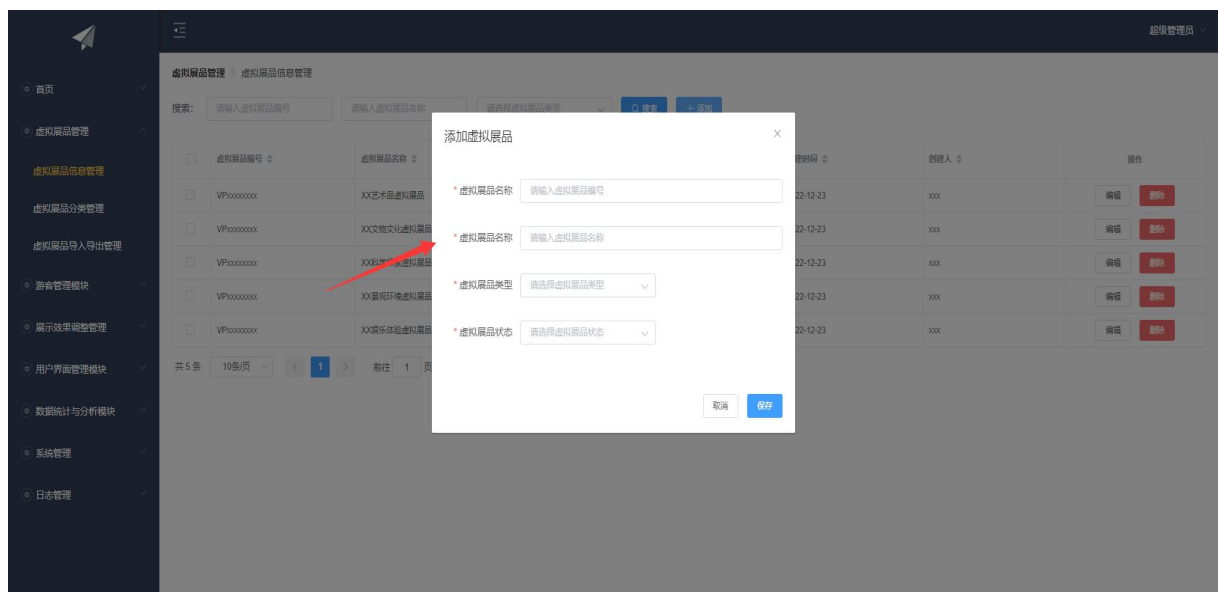
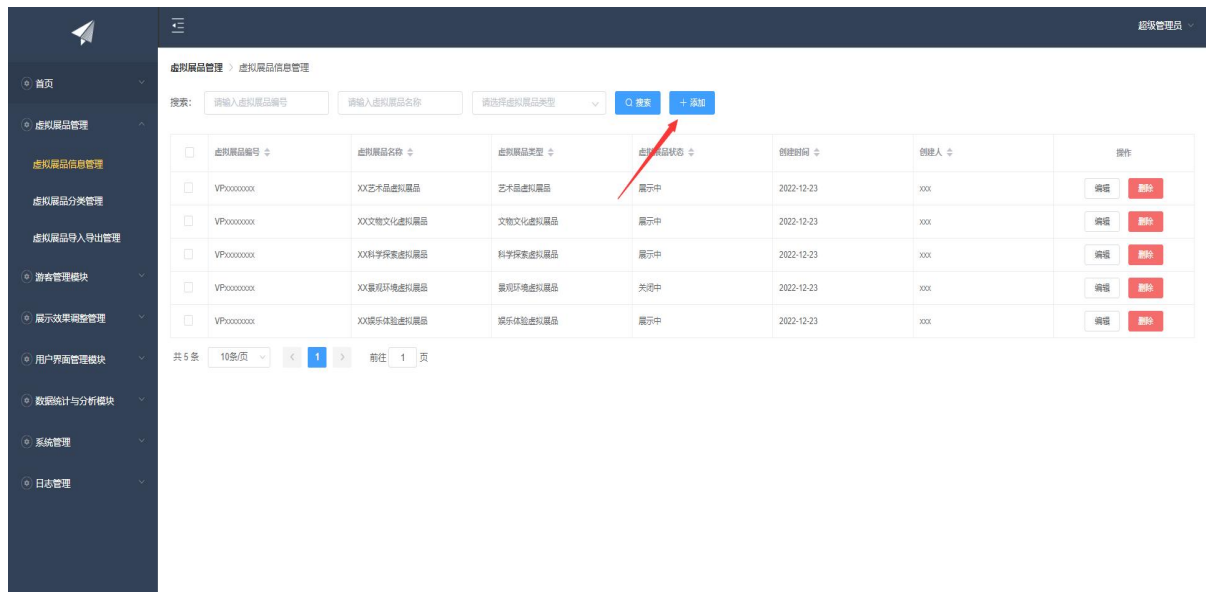
The interface shows the 'Virtual Exhibit Management' section with a table of exhibit information. A red arrow points to the 'Virtual Exhibit Information Management' menu item in the sidebar.

| 虚拟展品编号 | 虚拟展品名称 | 虚拟展品类型 | 虚拟展品状态 | 创建时间 | 创建人 | 操作 |
|------------|------------|----------|--------|------------|-----|---|
| VP00000001 | XX艺术品虚拟展品 | 艺术品虚拟展品 | 展示中 | 2022-12-23 | xxx | <button>编辑</button> <button>删除</button> |
| VP00000002 | XX文物文化虚拟展品 | 文物文化虚拟展品 | 展示中 | 2022-12-23 | xxx | <button>编辑</button> <button>删除</button> |
| VP00000003 | XX科学探索虚拟展品 | 科学探索虚拟展品 | 展示中 | 2022-12-23 | xxx | <button>编辑</button> <button>删除</button> |
| VP00000004 | XX虚拟环境虚拟展品 | 虚拟环境虚拟展品 | 关闭中 | 2022-12-23 | xxx | <button>编辑</button> <button>删除</button> |
| VP00000005 | XX娱乐体验虚拟展品 | 娱乐体验虚拟展品 | 展示中 | 2022-12-23 | xxx | <button>编辑</button> <button>删除</button> |

共 5 条 | 10条/页 | 1 | 前往 1 页

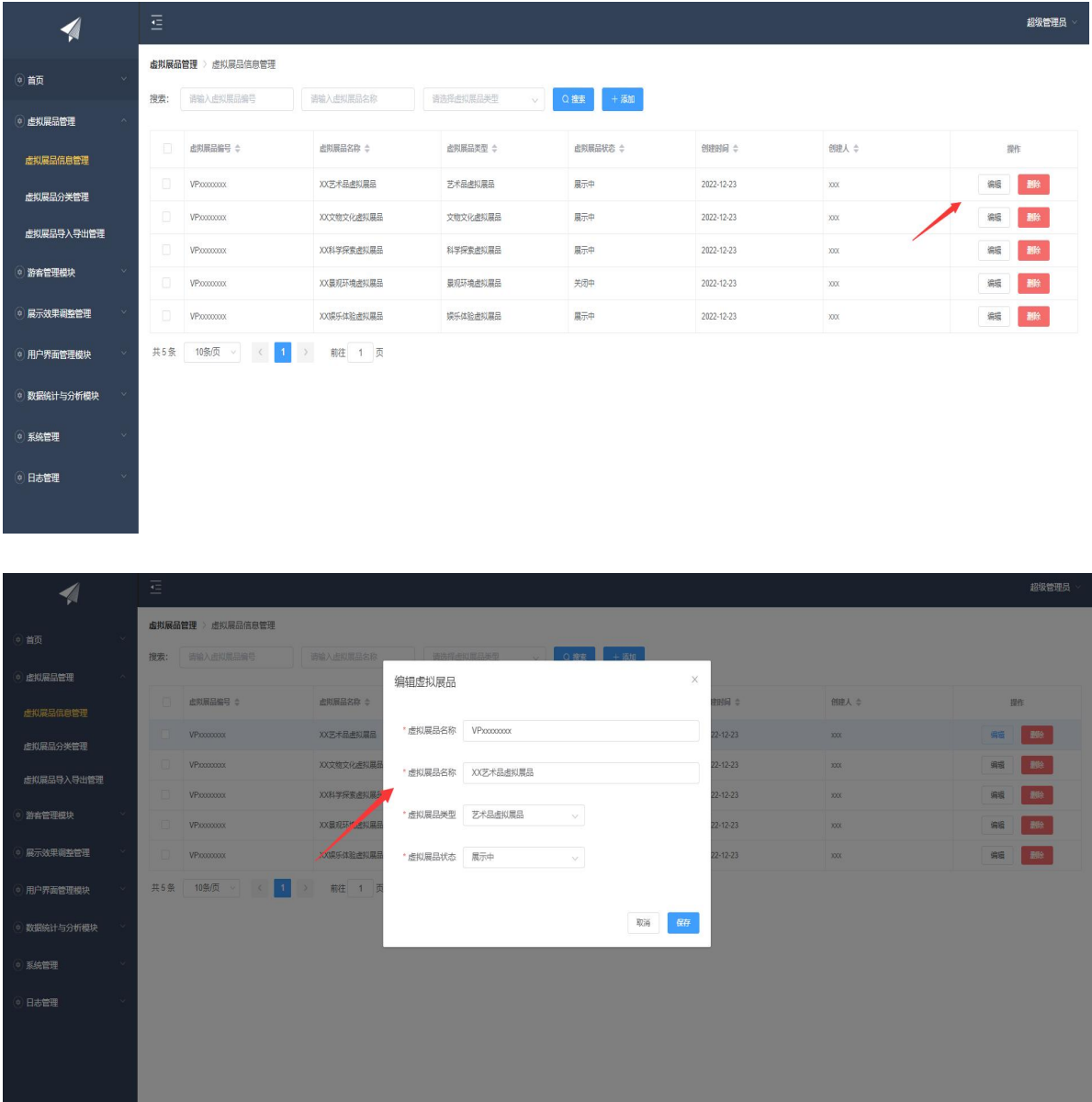
2.2.4 虚拟展品信息添加

点击添加按钮，弹出虚拟展品信息添加弹窗，填写信息添加即可。



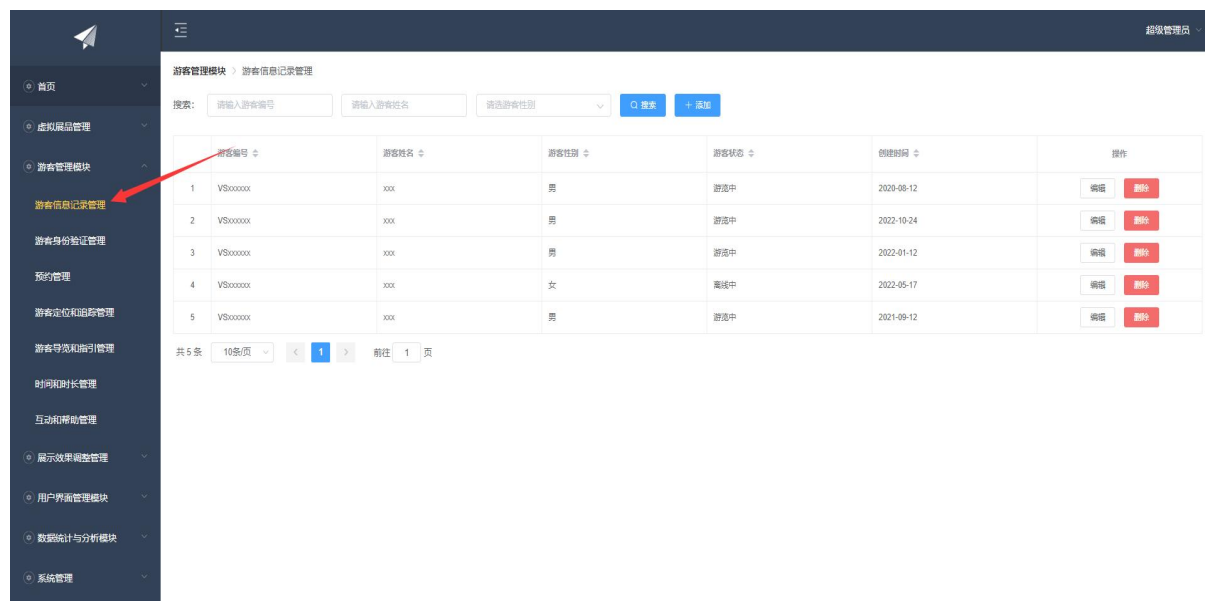
2.3.5 虚拟展品信息编辑

点击右侧编辑按钮，弹出虚拟展品信息编辑弹窗，填写信息编辑即可。



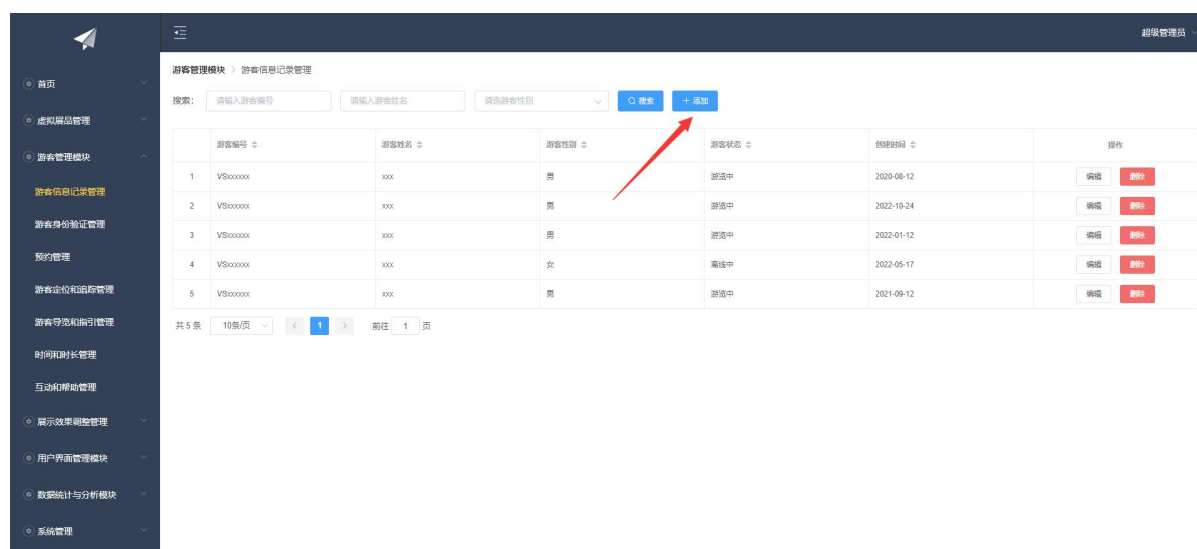
2.2.6 游客信息管理

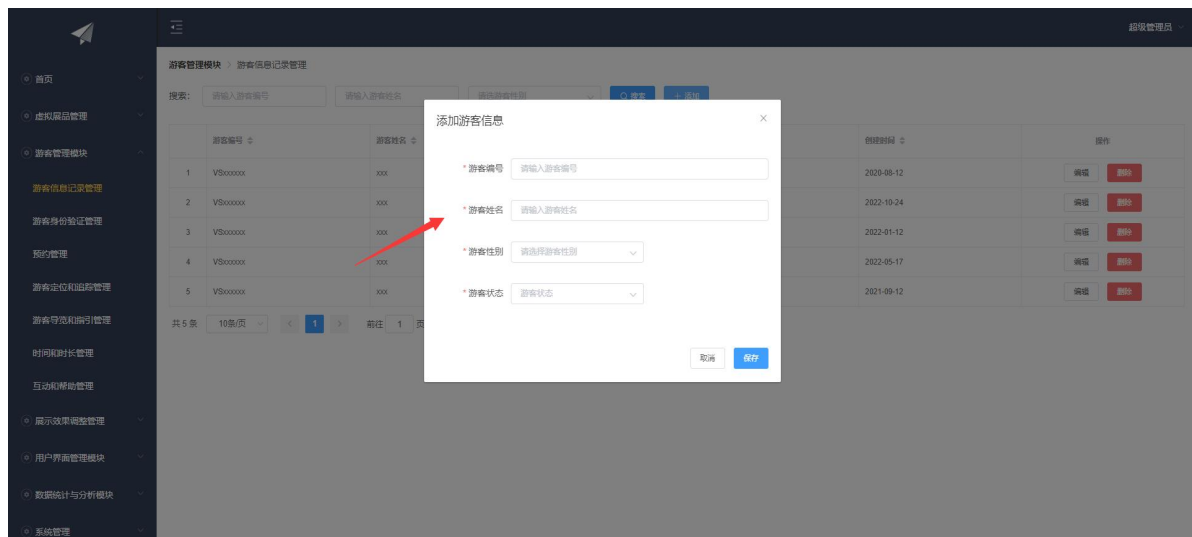
点击【游客管理模块】菜单，再点击游客信息管理进入列表页。展示信息：游客编号、游客名称、游客性别、游客状态、创建时间。



2.2.7 游客信息添加

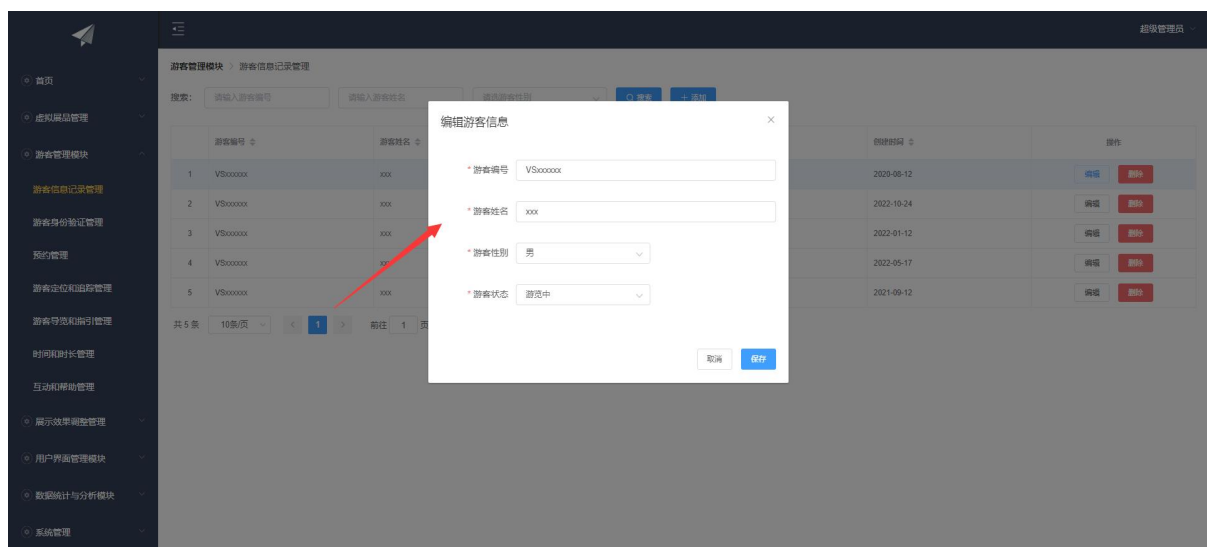
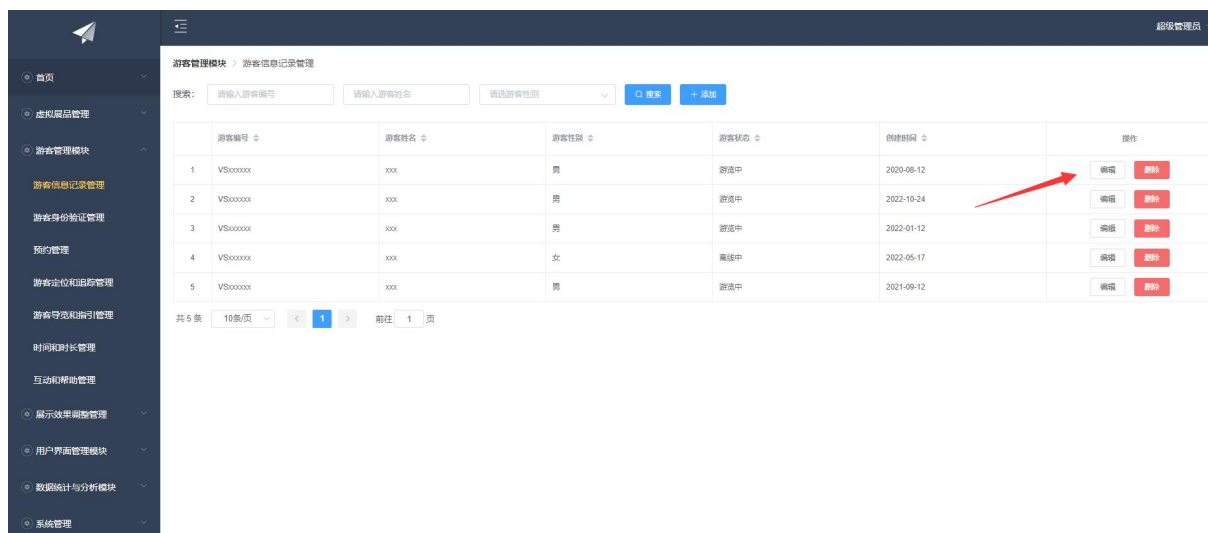
点击添加按钮，弹出游客信息添加弹窗，填写信息添加即可。





2.2.8 游客信息编辑

点击右侧编辑按钮，弹出游客信息编辑弹窗，填写信息编辑即可。



第三章 接口设计

3.1 虚拟展品信息和游客信息管理

3.1.1 虚拟展品信息信息管理接口

请求路径

/virtually/virtuallyProductList

请求方式 POST

json 参数

```
{
  "pageNo": 1,
  "pageSize": 10,
  "virtuallyProductNo": "xxxx",
  "virtuallyProductName": "xxxx",
  "virtuallyProductType": "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 |
| virtuallyProductNo | String | 是 | 虚拟展品信息编号 |
| virtuallyProductName | String | 是 | 虚拟展品信息名称 |
| virtuallyProductStatus | String | 是 | 虚拟展品信息状态 |
| 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 虚拟展品信息创建接口

请求路径

/virtually/addVirtuallyProduct

请求方式 POST

json 参数

```
{
  "virtuallyProductNo": "***",
  "virtuallyProductName": "***",
  "virtuallyProductType": "***",
  "virtuallyProductStatus": "***",
}
```

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

json 参数字段说明

| 参数名 | 类型 | 必选 | 说明 |
|------------------------|--------|----|----------|
| virtuallyProductNo | String | 是 | 虚拟展品信息编号 |
| virtuallyProductName | String | 是 | 虚拟展品信息名称 |
| virtuallyProductType | String | 是 | 虚拟展品信息类型 |
| virtuallyProductStatus | String | 是 | 虚拟展品信息状态 |
| 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 游客信息管理接口

请求路径

/virtually/sightseerList

请求方式 POST

json 参数

```
{
  "pageNo": 1,
  "pageSize": 10,
  "sightseerNo": xxx,
  "sightseerName": "xxx",
  "sightseerGender": "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 |
| sightseerNo | String | 是 | 游客信息编号 |
| sightseerName | String | 是 | 游客信息名称 |
| sightseerGender | String | 是 | 游客信息性别 |
| 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 游客信息创建接口

请求路径

/virtually/addSightseer

请求方式 POST

json 参数

```
{
  "sightseerNo": "***",
  "sightseerName": "***",
  "sightseerGender": "***",
  "sightseerStatus": "***",
  "createTime": "***",
}
```

json 参数字段说明

| 参数名 | 类型 | 必选 | 说明 |
|-----|----|----|----|
|-----|----|----|----|

| | | | |
|-----------------|--------|---|--------|
| sightseerNo | String | 是 | 游客信息号 |
| sightseerName | String | 是 | 游客信息名称 |
| sightseerGender | String | 是 | 游客信息性别 |
| sightseerStatus | String | 是 | 游客信息状态 |
| 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

/virtually/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

/virtually/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

/virtually/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

/virtually/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

/virtually/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

/virtually/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

/virtually/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

/virtually/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 虚拟展品信息信息表

简要描述:

- 虚拟展品信息信息

| 字段 | 类型 | 空 | 默认 | 注释 |
|------------------------|--------|------|------|----------|
| Id | String | true | Null | 主键 id |
| virtuallyProductNo | String | true | Null | 虚拟展品信息号 |
| virtuallyProductName | String | true | Null | 虚拟展品信息名称 |
| virtuallyProductType | String | true | Null | 虚拟展品信息类型 |
| virtuallyProductStatus | String | true | Null | 虚拟展品信息状态 |
| createTime | String | true | Null | 创建时间 |
| creator | String | true | Null | 创建人 |

- 备注: 无

4.2 游客信息表

简要描述:

- 游客信息

| 字段 | 类型 | 空 | 默认 | 注释 |
|----|--------|-------|------|-------|
| Id | String | flase | Null | 主键 id |

| | | | | |
|-----------------|--------|-------|------|--------|
| sightseerNo | String | true | Null | 游客信息编号 |
| sightseerName | String | true | Null | 游客信息姓名 |
| sightseerGender | String | flase | Null | 游客信息性别 |
| sightseerStatus | String | true | Null | 游客信息状态 |
| createTime | String | flase | 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 | flase | Null | 密码 |
| username | String | flase | Null | 用户名 |
| mobile | String | flase | 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 | flase | 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，不同角色游客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。