游戏管理系统使用和开发手册

深圳九五互娱有限公司

2019-1-8

目录

[一、 项目说明 3](#_Toc535079975)

[二、 功能介绍 3](#_Toc535079976)

[1. 用户管理 3](#_Toc535079977)

[添加用户 3](#_Toc535079978)

[编辑用户信息 4](#_Toc535079979)

[修改密码 5](#_Toc535079980)

[管理用户角色 6](#_Toc535079981)

[编辑角色 7](#_Toc535079982)

[删除用户 7](#_Toc535079983)

[2. 角色管理 8](#_Toc535079984)

[添加角色 8](#_Toc535079985)

[角色信息编辑 9](#_Toc535079986)

[管理权限 10](#_Toc535079987)

[编辑权限 11](#_Toc535079988)

[删除角色 12](#_Toc535079989)

[3. 权限管理 13](#_Toc535079990)

[添加权限 13](#_Toc535079991)

[编辑权限 14](#_Toc535079992)

[删除权限 15](#_Toc535079993)

[4. 游戏管理 15](#_Toc535079994)

[添加游戏 15](#_Toc535079995)

[编辑游戏信息 17](#_Toc535079996)

[删除游戏 18](#_Toc535079997)

[5. 平台管理 18](#_Toc535079998)

[添加平台 18](#_Toc535079999)

[编辑平台信息 20](#_Toc535080000)

[删除平台 20](#_Toc535080001)

[6. 数据源管理 21](#_Toc535080002)

[添加数据源 21](#_Toc535080003)

[编辑数据源 23](#_Toc535080004)

[测试数据源 23](#_Toc535080005)

[删除数据源 24](#_Toc535080006)

[7. 渠道管理 25](#_Toc535080007)

[添加渠道 25](#_Toc535080008)

[编辑渠道 26](#_Toc535080009)

[删除渠道 26](#_Toc535080010)

[8. 服务器管理 27](#_Toc535080011)

[服务器同步 27](#_Toc535080012)

[编辑服务器信息 28](#_Toc535080013)

[设置服务器状态 29](#_Toc535080014)

[设置默认服务器 30](#_Toc535080015)

[添加平台渠道 31](#_Toc535080016)

[删除服务器 32](#_Toc535080017)

[9. 服务器树形管理 32](#_Toc535080018)

[查看服务器信息 32](#_Toc535080019)

[编辑服务器信息 34](#_Toc535080020)

[删除服务器 34](#_Toc535080021)

[10. 玩家基本信息 35](#_Toc535080022)

[查询玩家详细信息 35](#_Toc535080023)

[玩家禁言 36](#_Toc535080024)

[玩家禁封 38](#_Toc535080025)

[11. 玩家禁言记录 39](#_Toc535080026)

[查看禁言记录 39](#_Toc535080027)

[12. 玩家禁封记录 39](#_Toc535080028)

[查看玩家禁封记录 39](#_Toc535080029)

[13. IP禁封 40](#_Toc535080030)

[添加IP禁封申请 40](#_Toc535080031)

[IP禁封 41](#_Toc535080032)

[删除禁封申请 41](#_Toc535080033)

[14. 物品流通日志 42](#_Toc535080034)

[查看物品流通日志 42](#_Toc535080035)

[15. 现金流日志 43](#_Toc535080036)

[查看现金流日志 43](#_Toc535080037)

[16. 角色创建日志 44](#_Toc535080038)

[查看角色创建日志 44](#_Toc535080039)

[17. 角色登录日志 45](#_Toc535080040)

[查看角色登录日志 45](#_Toc535080041)

[18. 充值流水日志 46](#_Toc535080042)

[查看充值流水日志 46](#_Toc535080043)

[19. 充值消费日志 47](#_Toc535080044)

[查看充值消费日志 47](#_Toc535080045)

[20. 发送邮件 48](#_Toc535080046)

[进行提交邮件申请 48](#_Toc535080047)

[编辑邮件信息 49](#_Toc535080048)

[发送邮件 50](#_Toc535080049)

[删除邮件申请 50](#_Toc535080050)

[21. 发送广播 50](#_Toc535080051)

[提交广播申请 50](#_Toc535080052)

[发送广播 52](#_Toc535080053)

[删除广播申请 52](#_Toc535080054)

[22. 全服邮件 52](#_Toc535080055)

[添加全服邮件申请 52](#_Toc535080056)

[编辑全服邮件申请 54](#_Toc535080057)

[发送全服邮件 54](#_Toc535080058)

[删除全服邮件申请 55](#_Toc535080059)

[23. 全服公告 55](#_Toc535080060)

[添加全服更新公告申请 55](#_Toc535080061)

[编辑全服公告信息 56](#_Toc535080062)

[发送全服公告 57](#_Toc535080063)

[删除全服公告申请 57](#_Toc535080064)

[24. 道具导入信息 58](#_Toc535080065)

[查看道具导入信息 58](#_Toc535080066)

[25. 道具导入 58](#_Toc535080067)

[导入道具信息 58](#_Toc535080068)

[26. 发送道具邮件 60](#_Toc535080069)

[添加道具邮件申请 60](#_Toc535080070)

[编辑道具邮件 61](#_Toc535080071)

[发送道具邮件 63](#_Toc535080072)

[删除道具邮件申请 63](#_Toc535080073)

[27. 礼包导入信息 63](#_Toc535080074)

[查看礼包导入信息 63](#_Toc535080075)

[28. 礼包导入 64](#_Toc535080076)

[礼包信息导入 64](#_Toc535080077)

[29. 申请礼包激活码 65](#_Toc535080078)

[生成礼包激活码 66](#_Toc535080079)

[下载激活码文件 67](#_Toc535080080)

[删除礼包激活码文件 67](#_Toc535080081)

[30. 激活码使用情况 67](#_Toc535080082)

[查询激活码是否有效 67](#_Toc535080083)

[31. 游客账号设置 68](#_Toc535080084)

[设置游客账号 68](#_Toc535080085)

[32. 文件中转 69](#_Toc535080086)

[文件上传 69](#_Toc535080087)

[文件下载 69](#_Toc535080088)

[文件删除 70](#_Toc535080089)

[33. 用户中心 70](#_Toc535080090)

[编辑个人信息 70](#_Toc535080091)

[密码修改 71](#_Toc535080092)

[34. 用户登录 72](#_Toc535080093)

[用户登录 72](#_Toc535080094)

[35. 退出登录 72](#_Toc535080095)

[退出登录 72](#_Toc535080096)

[36. 游客登录 73](#_Toc535080097)

[游客登录 73](#_Toc535080098)

[37. 系统后门 73](#_Toc535080099)

[三、 相关技术 73](#_Toc535080100)

[1.前端技术栈： 73](#_Toc535080101)

[2.后端技术栈： 74](#_Toc535080102)

[四、 目录结构 74](#_Toc535080103)

[前端目录结构 74](#_Toc535080104)

[后端目录结构 76](#_Toc535080105)

[五、 搭建开发环境 77](#_Toc535080106)

[1.前端开发流程 77](#_Toc535080107)

[2.后端开发流程 78](#_Toc535080108)

[六、 功能开发 79](#_Toc535080109)

[权限系统 79](#_Toc535080110)

[区服管理系统 80](#_Toc535080111)

[接口开发 81](#_Toc535080112)

[1) 定义控制器 81](#_Toc535080113)

[2) 设置接口url 81](#_Toc535080114)

[3) 获取数据 81](#_Toc535080115)

[前端组件开发 82](#_Toc535080116)

[前端组件通讯 82](#_Toc535080117)

[七、 构筑生产 82](#_Toc535080118)

# 项目说明

该项目是基于Spring Boot + Vue.js 2.x系列 + Element UI 实现的后台游戏管理系统。

# 功能介绍

## 用户管理

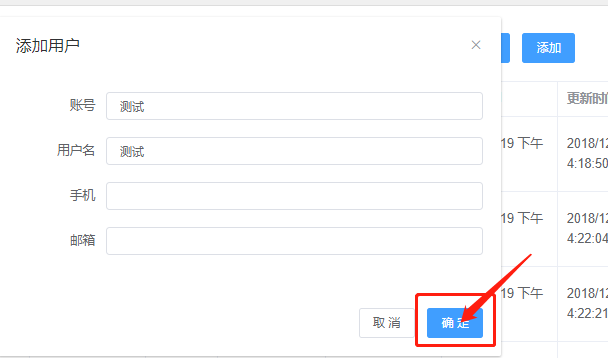
管理和显示系统所有的用户，可以进行用户添加，用户信息编辑，修改密码，编辑用户角色，删除用户等操作，每个用户可已拥有多个角色。

### 添加用户

点击用户管理进入用户管理页面，点击添加按钮，弹出添加用户对话框。



按要求输入用户信息，点击确认即可。



### 编辑用户信息

点击编辑按钮，弹出编辑信息对话框。



按要求输入用户信息，点击确认即可。

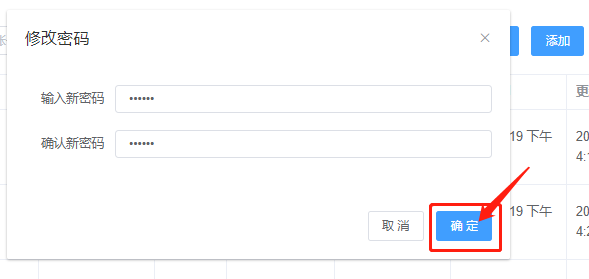


### 修改密码

点击修改密码按钮，弹出密码修改对话框



输入密码后，点击确认按钮即可



### 管理用户角色

点击管理角色按钮，弹出管理角色对话框



勾选需要变动的角色，点击添加或移除，点击确认即可



### 编辑角色

点击编辑角色按钮，弹出编辑角色对话框



按格式编辑角色，点击确认即可

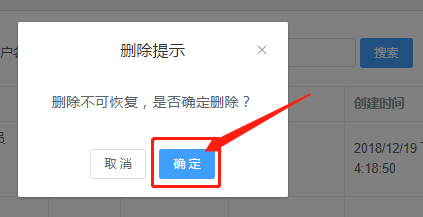


### 删除用户

点击删除按钮，弹出删除对话框



点击确认即可

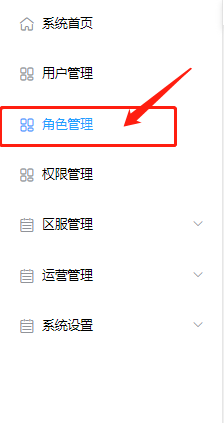


## 角色管理

管理和显示所有系统的角色，可以进行角色添加，角色信息编辑，修改角色权限，删除角色等操作，每个角色可以拥有多个权限。

### 添加角色

点击菜单栏的角色管理进入角色管理页面，点击添加按钮弹出添加角色对话框



输入角色信息，点击确认按钮即可



### 角色信息编辑

点击编辑按钮，弹出编辑角色信息对话框



修改角色信息后，点击确认按钮即可  


### 管理权限

点击管理权限按钮，弹出权限管理对话框



勾选需要修改的权限，点击添加或者移除，点击确认按钮即可

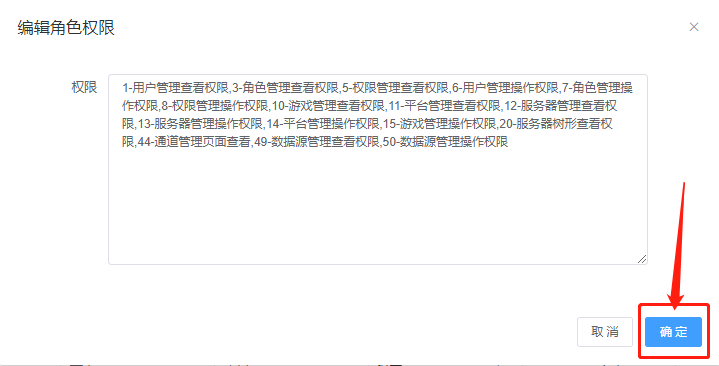


### 编辑权限

点击编辑权限按钮，弹出编辑权限对话框



按格式输入权限信息，点击确认即可



### 删除角色

点击删除按钮，弹出删除对话框



点击确认即可



## 权限管理

管理和显示所有系统的权限，可以进行添加权限，编辑权限，删除权限等操作，权限可以对功能和路由进行限制。

### 添加权限

点击菜单栏的权限管理按钮，进出权限管理页面，点击添加按钮，弹出添加权限对话框

输入权限信息后，点击确认按钮即可



### 编辑权限

点击编辑按钮，弹出编辑权限对话框



输入权限信息，点击确认按钮即可



### 删除权限

点击删除权限按钮，弹出删除权限对话框



点击确认按钮即可



## 游戏管理

管理和显示所有的游戏，可以进行游戏添加，游戏信息编辑，游戏服务器获取接口编辑，游戏删除等操作，一个游戏对应一个游戏服务器获取接口，接口可以获取该游戏下所有的服务器列表。

### 添加游戏

点击菜单栏的游戏管理按钮进入游戏管理页面，点击添加按钮打开添加游戏对话框



输入游戏信息和服务器接口地址，点击确认按钮即可



### 编辑游戏信息

点击编辑按钮，弹出编辑游戏信息对话框



输入游戏信息和服务器接口地址，点击确认即可



### 删除游戏

点击删除按钮，弹出删除游戏对话框



点击确认即可

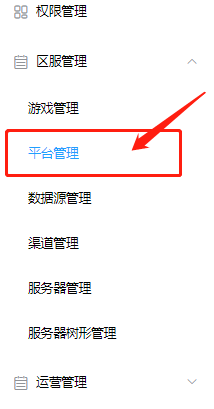


## 平台管理

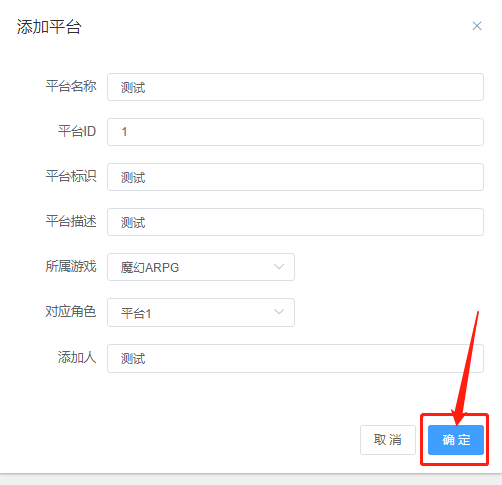
管理和显示所有的平台，可以进行平台添加，平台信息编辑，平台删除等操作，一个平台对应一个游戏。

### 添加平台

点击菜单栏的平台管理按钮，进入平台管理页面，点击添加按钮，弹出添加平台对话框



输入平台基本信息，选择所属游戏和对应的角色，点击确认即可

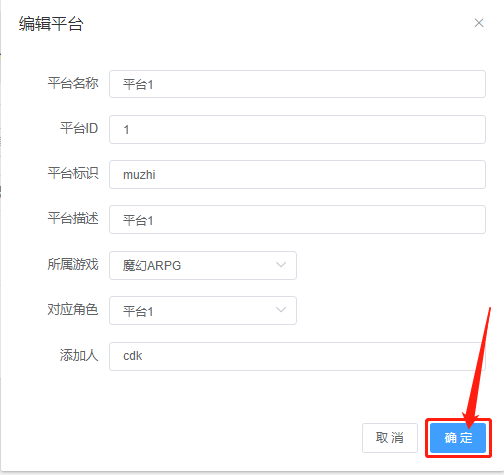


### 编辑平台信息

点击编辑按钮，打开编辑平台信息对话框



输入平台信息，点击确认按钮即可



### 删除平台

点击删除按钮，打开删除平台对话框



点击确认即可



## 数据源管理

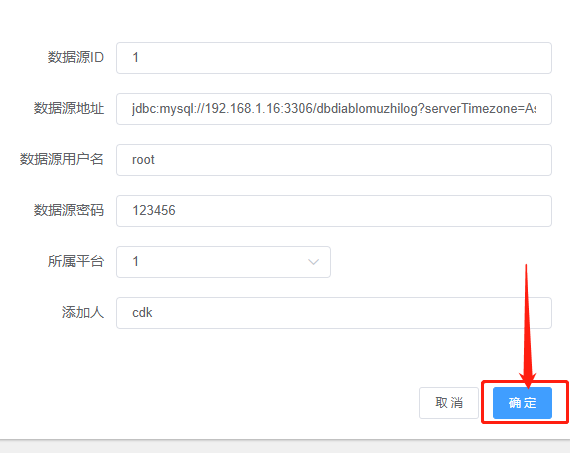
管理和显示所有的数据源，可以进行数据源的添加，数据源编辑，数据源测试，数据源删除等操作，一个平台对应一个且唯一的数据源，数据源主要用来获取该平台的所有日志数据。

### 添加数据源

点击菜单栏的数据源管理按钮打开数据源管理页面，点击添加按钮，打开添加数据源对话框



输入数据源信息，点击确认按钮即可

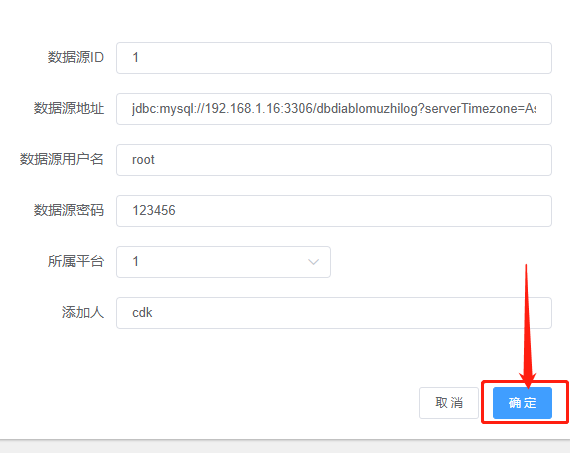


### 编辑数据源

点击编辑按钮，打开数据源信息编辑对话框



输入数据源信息，点击确认按钮即可



### 测试数据源

点击测试数据源按钮



弹出测试成功则表示数据源测试成功

### 删除数据源

点击删除按钮，弹出删除数据源对话框



点击确认按钮即可

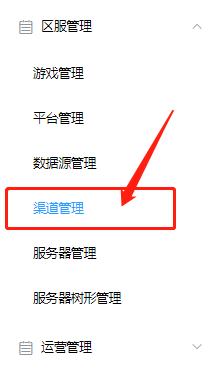


## 渠道管理

管理和显示所有的渠道，可以进行渠道添加，渠道编辑，渠道删除等操作，一个渠道对应一个且唯一的平台。

### 添加渠道

点击菜单栏的渠道管理按钮，打开渠道管理页面，点击添加按钮，打开添加渠道对话框

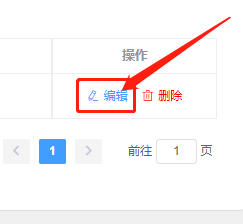


输入渠道信息，点击确认按钮即可



### 编辑渠道

点击编辑按钮，弹出编辑渠道对话框

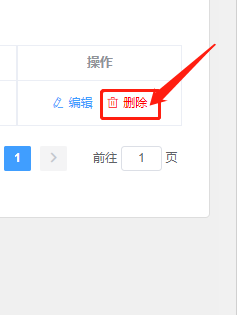


输入渠道信息，点击确认即可



### 删除渠道

点击删除渠道按钮，弹出删除渠道对话框



点击确认即可

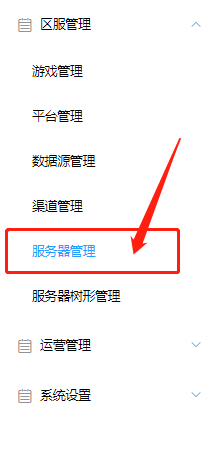


## 服务器管理

管理和显示所有的服务器，可以进行服务器同步，服务器编辑，设置服务器状态，设置默认服务器，添加平台渠道，删除服务器操作，一个服务器对应一个且唯一的平台，一个且唯一的游戏，可对应多个平台渠道。

### 服务器同步

点击菜单栏里的服务器管理按钮，打开服务器管理页面，点击同步按钮即可

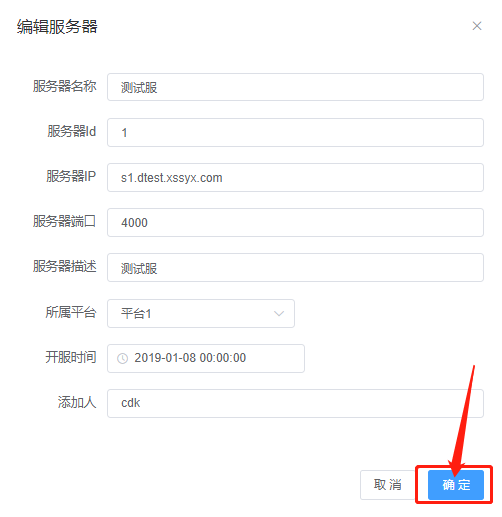


### 编辑服务器信息

点击编辑按钮，弹出编辑服务器对话框



输入服务器信息，点击确认按钮即可

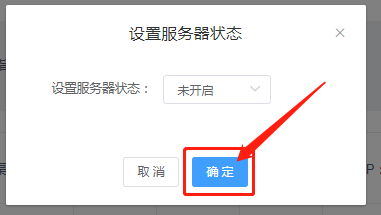


### 设置服务器状态

点击设置状态按钮，弹出设置服务器状态对话框



选择服务器状态，点击确认按钮即可



### 设置默认服务器

点击设置默认按钮，打开设置默认服务器对话框



点击确认即可



### 添加平台渠道

点击添加平台渠道按钮，打开添加平台渠道对话框



勾选需要修改的平台渠道，点击确认或者移除按钮，点击确认按钮即可



### 删除服务器

点击删除按钮，弹出删除服务器对话框



点击确认即可

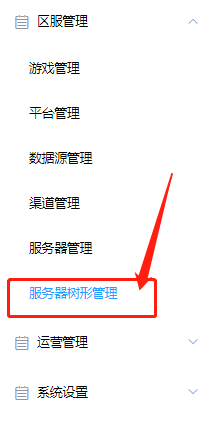
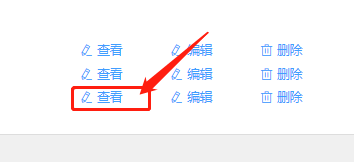


## 服务器树形管理

管理和显示所有服务器的树形结构图，可以进行服务器信息查看，服务器信息编辑，服务器删除操作。

### 查看服务器信息

点击菜单栏的服务器树形管理按钮，打开服务器树形管理页面，点击服务器右侧的查看按钮

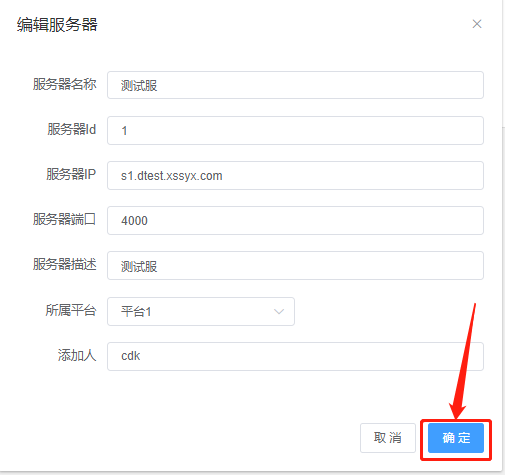
  

### 编辑服务器信息

点击服务器右侧的编辑按钮，打开编辑服务器信息对话框



输入服务器信息，点击确认即可



### 删除服务器

点击服务器右侧的删除按钮，打开删除服务器对话框



点击确认按钮即可

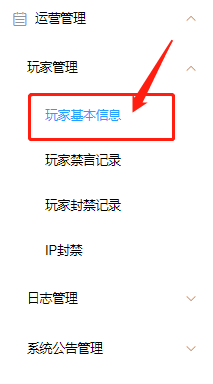
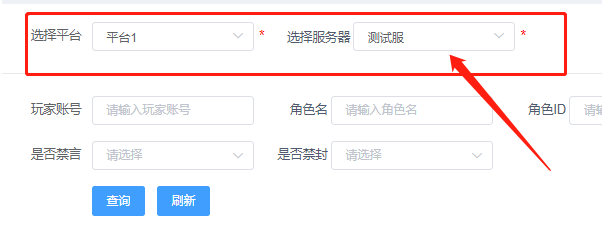


## 玩家基本信息

管理和显示不同平台服务器下的玩家信息，可以进行玩家详细信息查询，玩家禁言，玩家禁封，玩家解除禁言，玩家解除禁封等操作。

### 查询玩家详细信息

点击菜单栏的玩家基本信息按钮，打开玩家基本信息页面，选择平台和服务器

点击详细信息按钮，即可显示玩家详细信息





### 玩家禁言

点击禁言按钮，弹出禁言对话框



选择永久禁言或者显示禁言，点击确认即可



————————



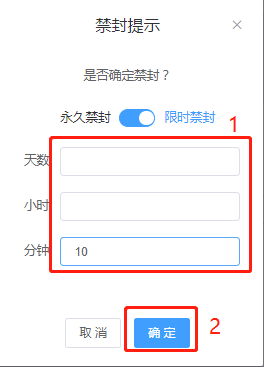
### 玩家禁封

点击禁封按钮，弹出禁封对话框



选择永久禁封或者限时禁封，点击确认即可



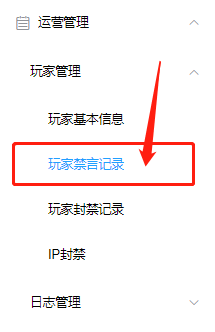


## 玩家禁言记录

显示所有玩家禁言操作的记录。

### 查看禁言记录

点击菜单栏的玩家禁言记录按钮，打开玩家禁言记录页面

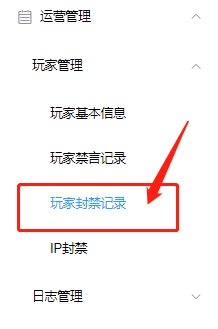


## 玩家禁封记录

显示所有玩家禁封操作的记录。

### 查看玩家禁封记录

点击菜单栏的玩家禁封记录按钮，打开玩家禁封记录页面

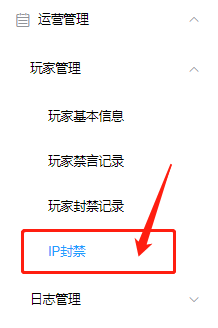


## IP禁封

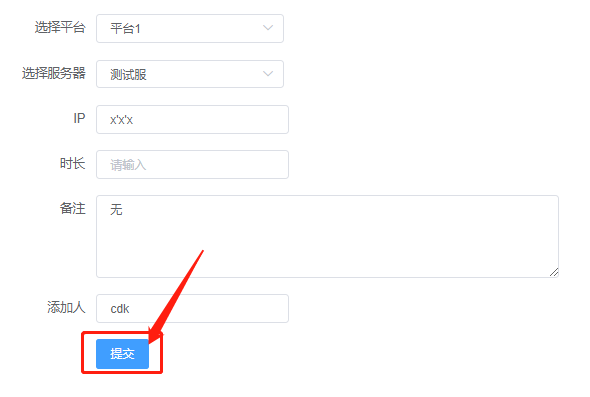
禁封物理IP地址，并显示所有禁封记录，可以进行添加IP禁封申请和IP禁封，删除禁封申请等操作。

### 添加IP禁封申请

点击菜单栏的IP禁封按钮，打开IP禁封页面



输入IP禁封申请，点击提交按钮即可



### IP禁封

点击禁封按钮即可



### 删除禁封申请

点击删除按钮即可

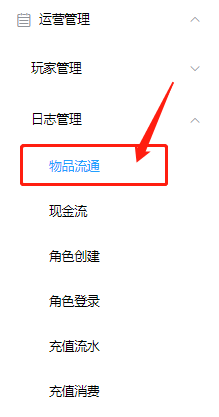


## 物品流通日志

显示指定平台服务器下20天以内的所有物品流通日志记录

### 查看物品流通日志

点击菜单栏的物品流通日志按钮，打开物品流通日志页面



选择平台和服务器，点击查询即可

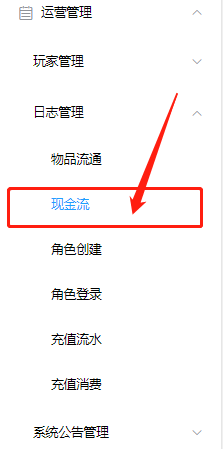


## 现金流日志

显示指定平台服务器下20天以内的所有现金流日志记录

### 查看现金流日志

点击菜单栏的现金流日志按钮，打开现金流日志页面



选择平台和服务器，点击查询即可



## 角色创建日志

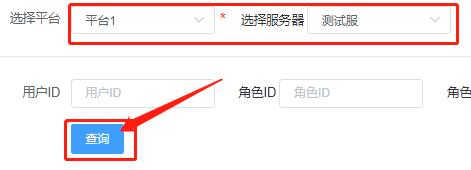
显示指定平台服务器下20天以内的所有角色创建日志记录

### 查看角色创建日志

点击菜单栏的角色创建按钮，打开角色创建日志页面



选择平台和服务器，点击查询按钮即可



## 角色登录日志

显示指定平台服务器下20天以内的所有角色登录日志记录

### 查看角色登录日志

点击菜单栏的角色登录日志按钮，打开角色登录日志页面



选择平台和服务器，点击查询按钮即可

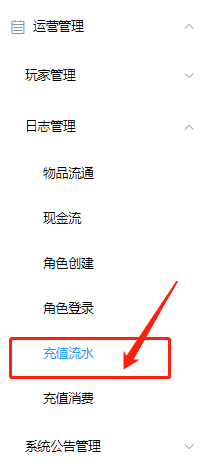


## 充值流水日志

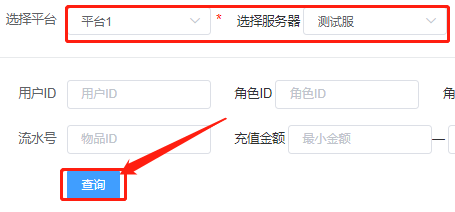
显示指定平台服务器下20天以内的所有充值流水日志记录

### 查看充值流水日志

点击菜单栏的充值流水按钮，打开充值流水日志页面



选择平台和服务器，点击查询按钮即可

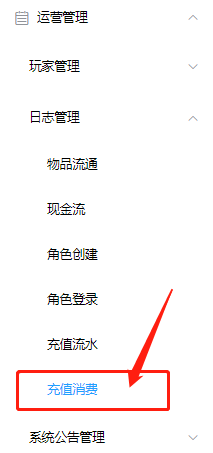


## 充值消费日志

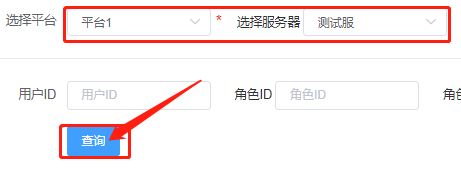
显示指定平台服务器下20天以内的所有充值消费日志记录

### 查看充值消费日志

点击菜单栏的充值消费按钮，打开充值消费日志页面



选择平台和服务器，点击查询按钮即可

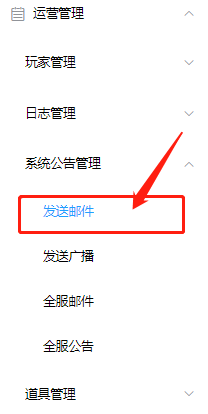


## 发送邮件

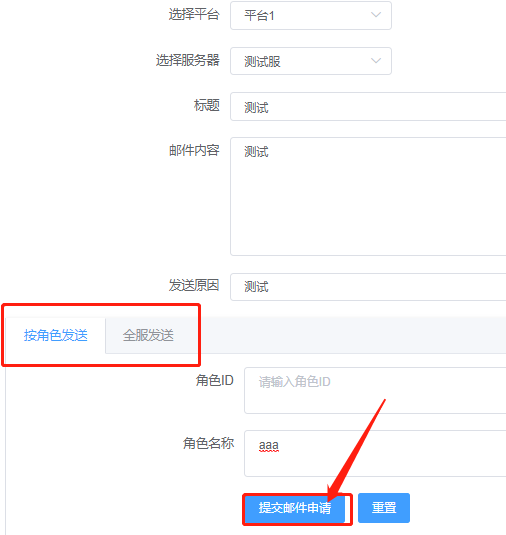
管理和显示所有发送邮件申请，可以进行提交邮件申请，发送邮件，编辑邮件申请，删除邮件申请等操作，邮件申请有两种发送方式，指定角色发送和全服玩家发送，均为单个服务器。

### 进行提交邮件申请

点击菜单栏的发送邮件按钮，打开发送邮件页面



输入发送邮件申请基本信息，选择发送类型，点击提交邮件申请按钮即可



### 编辑邮件信息

点击编辑按钮，打开编辑邮件信息对话框



输入邮件信息，点击编辑按钮即可



### 发送邮件

点击发送按钮即可



### 删除邮件申请

点击删除按钮即可



## 发送广播

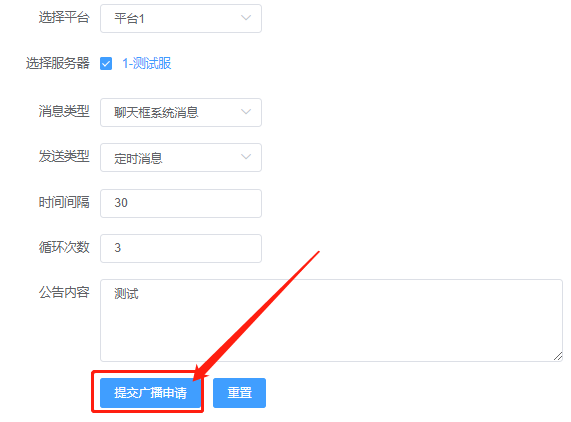
管理和显示所有发送广播申请，可以进行提交广播申请，发送广播，删除广播申请等操作，广播的消息类型分为聊天框系统消息和滚屏消息，发送类型有即时发送和定时发送。

### 提交广播申请

点击菜单栏的发送广播按钮，打开发送广播页面



输入广播申请信息，选择消息类型和发送类型，点击提交广播申请即可



### 发送广播

点击发送按钮即可



### 删除广播申请

点击删除按钮即可

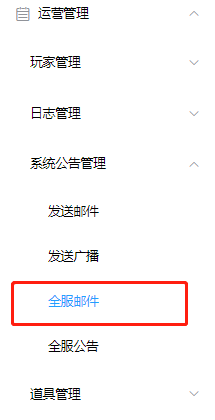


## 全服邮件

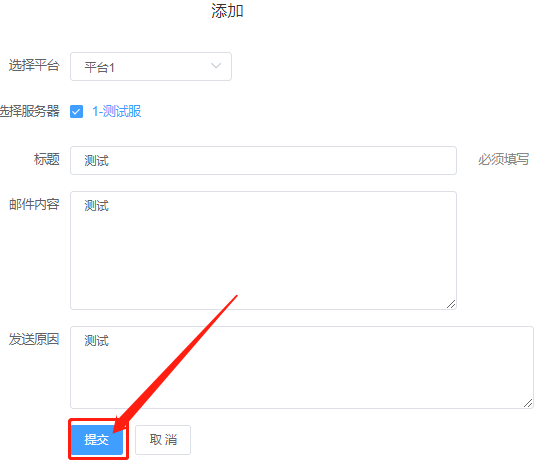
管理和显示所有全服邮件申请，可以进行添加全服邮件申请，编辑全服邮件申请，发送全服邮件，删除全服邮件申请，重新发送全服邮件等操作，可以进行多个服务器操作

### 添加全服邮件申请

点击菜单栏的全服邮件按钮，打开全服邮件页面，点击添加全服邮件申请按钮，打开添加全服邮件申请对话框



输入全服邮件信息，选择平台和服务器，点击提交按钮即可

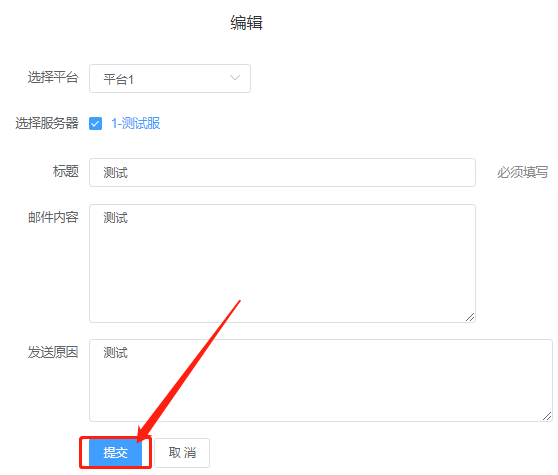


### 编辑全服邮件申请

点击编辑按钮，弹出编辑全服邮件对话框



输入全服邮件信息，点击提交按钮即可



### 发送全服邮件

点击发送按钮即可



### 删除全服邮件申请

点击删除按钮即可



## 全服公告

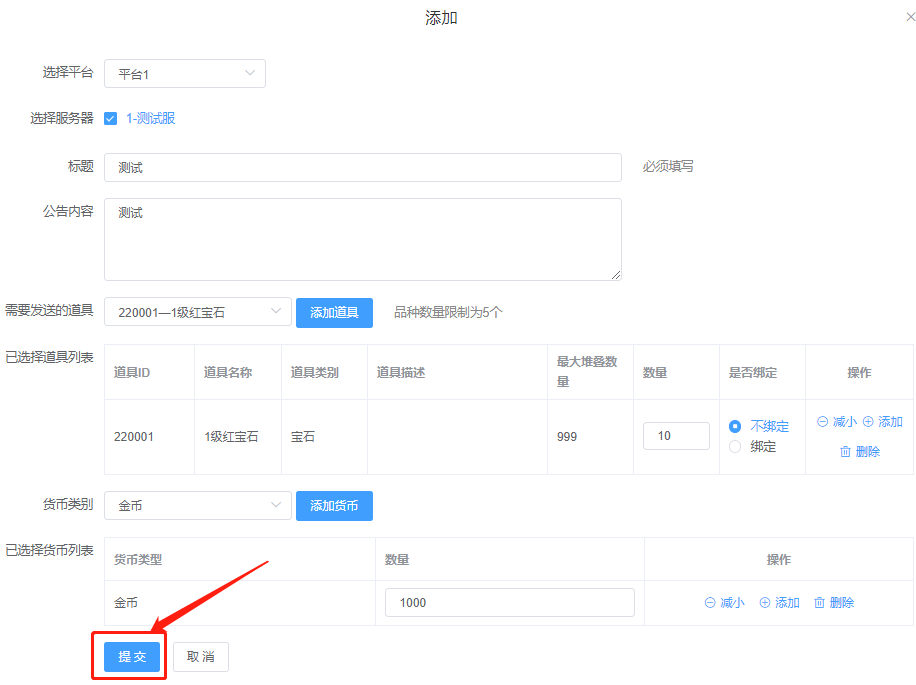
管理和显示所有的全服更新公告申请，可以进行添加全服更新公告申请，发送全服邮件申请，编辑全服邮件申请，删除全服邮件申请，重新发送全服邮件申请等操作，可以进行多个服务器操作。

### 添加全服更新公告申请

点击菜单栏的全服公告按钮，打开全服公告页面，点击添加全服公告申请按钮，打开添加全服公告申请对话框



输入全服公告信息，点击提交按钮即可



### 编辑全服公告信息

点击编辑按钮，弹出编辑全服公告弹出框



输入全服公告信息，点击提交即可



### 发送全服公告

点击发送按钮即可



### 删除全服公告申请

点击删除按钮即可



## 道具导入信息

显示指定游戏平台下所有的道具信息

### 查看道具导入信息

点击菜单栏的道具导入信息按钮，打开道具导入信息页面即可



## 道具导入

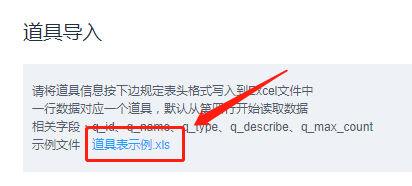
导入道具信息到指定的游戏平台

### 导入道具信息

点击菜单栏的道具导入按钮，打开道具导入页面



下载示例文件，按格式编辑道具信息



上传文件，选择平台，点击导入数据库按钮即可

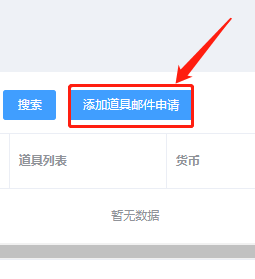
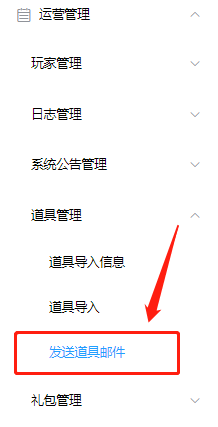


## 发送道具邮件

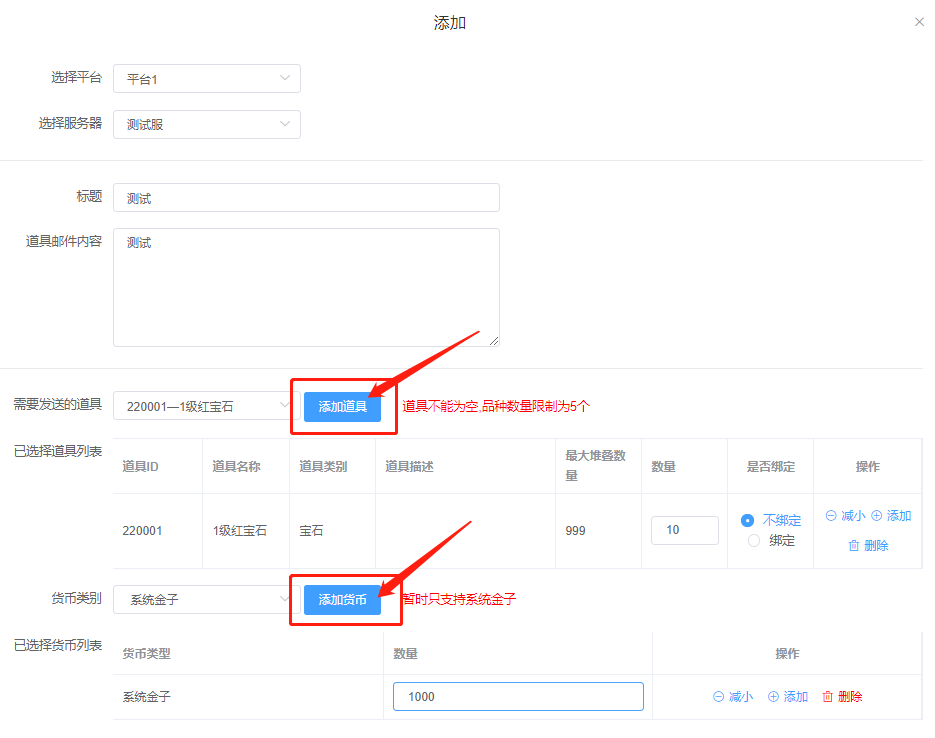
管理和显示所有的道具邮件申请，可以进行添加道具邮件申请，编辑道具邮件申请，发送道具邮件申请，删除道具邮件申请等操作，发送道具邮件方式“为指定角色发送”、“为全服玩家发送”， 均为单个服务器。

### 添加道具邮件申请

点击菜单栏的发送道具邮件按钮，打开发送道具邮件申请页面，点击添加道具邮件申请按钮，打开添加道具邮件申请对话框



输入道具邮件信息，添加需要发送的道具或者货币，选择发送类型，点击申请按钮即可





### 编辑道具邮件

点击编辑按钮，弹出编辑道具邮件对话框



输入道具邮件信息，点击编辑按钮即可





### 发送道具邮件

点击发送邮件按钮即可



### 删除道具邮件申请

点击删除按钮即可



## 礼包导入信息

显示指定游戏平台下所有的礼包信息

### 查看礼包导入信息

点击菜单栏的礼包导入信息按钮，打开礼包导入信息页面即可



## 礼包导入

导入礼包信息到指定的游戏平台

### 礼包信息导入

点击菜单栏的礼包导入按钮，打开礼包导入页面



下载礼包示例文件，按格式编辑礼包信息



上传文件，选择平台，点击导入数据库按钮即可



## 申请礼包激活码

管理和显示所有的礼包激活码数据，可以进行生成礼包激活码，下载礼包激活码，删除礼包激活码等操作。

### 生成礼包激活码

点击菜单栏的申请礼包激活码按钮，打开申请礼包激活码页面



输入申请礼包激活码信息，点击生成激活码按钮即可



### 下载激活码文件

点击下载按钮即可



### 删除礼包激活码文件

点击删除按钮即可

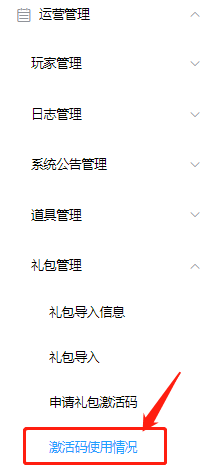


## 激活码使用情况

管理和显示所有激活码兑换记录，可以进行查询激活码是否有效等操作。

### 查询激活码是否有效

点击菜单栏的激活码使用情况按钮，打开激活码使用情况页面



选择平台，输入cdk，点击查询是否有效按钮即可

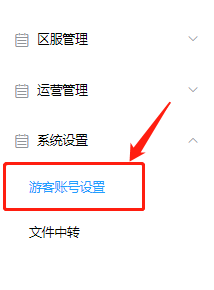


## 游客账号设置

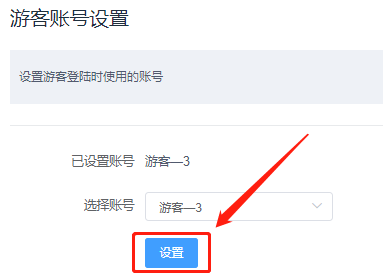
设置游客登陆时所使用的账号

### 设置游客账号

点击菜单栏的游客账号设置按钮，打开游客账号设置页面



选择账号，点击按钮即可

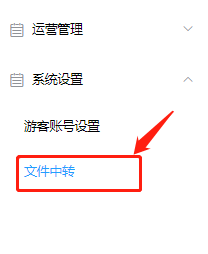


## 文件中转

管理和显示所有的中转文件，可以进行添加文件，文件下载，文件删除等操作。

### 文件上传

点击菜单栏的文件中转按钮，打开文件中转页面，点击添加文件按钮，打开文件上传对话框



选择平台，点击选取文件按钮选择文件，点击文件上传按钮即可



### 文件下载

点击下载按钮即可



### 文件删除

点击文件删除按钮即可



## 用户中心

可以显示和修改个人信息，可以修改用户登录密码

### 编辑个人信息

点击系统右上角用户名或用户头像，进入用户中心



点击信息修改页面，点击编辑个人信息按钮，弹出编辑信息对话框





输入用户信息，点击修改按钮即可



### 密码修改

点击密码修改页面，输入密码，点击修改密码按钮即可



## 用户登录

用户登录

### 用户登录

登录页面输入用户名和密码，点击登录按钮即可



## 退出登录

退出登录

### 退出登录

点击系统右上角的用户名，点击退出登录即可

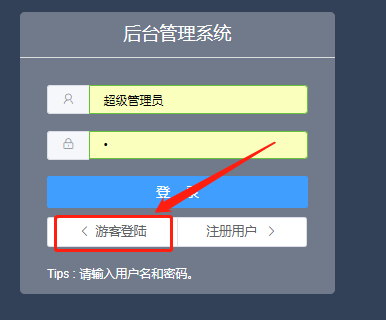


## 游客登录

以游客用户身份登陆系统

### 游客登录

登录页面点击游客登陆按钮即可



## 系统后门

# 相关技术

## 1.前端技术栈：

编程语言：**html5**、**js**、**css**

开发工具：**Visual** **Studio** **Code**

开发框架：**vue** + **axios**

包管理工具:**npm**

打包工具：**webpack**

## 2.后端技术栈：

编程语言：**java**

开发工具：**IntelliJ IDEA**

开发框架：**spring** **boot**

包管理工具: **maven资源库**

打包工具：**spring-boot-maven-plugin**

# 目录结构

## 前端目录结构

1. |-- build // webpack配置文件
2. |-- config // 项目打包路径
3. |-- src // 源码目录
4. | |-- code // 代码块
5. | |-- formatDatetime.js // 时间格式转换
6. | |-- setLocalThisUrl.js // 设置接口地址
7. | |-- components // 组件
8. | |-- common // 公共组件
9. | |-- bus.js // Event Bus
10. | |-- Header.vue // 公共头部
11. | |-- Home.vue // 公共路由入口
12. | |-- Sidebar.vue // 公共左边栏
13. | |-- Tags.vue // 页面切换标签组件
14. | |-- example // 示例组件
15. | |-- BaseCharts.vue // 基础图表
16. | |-- BaseForm.vue // 基础表单
17. | |-- BaseTable.vue // 基础表格
18. | |-- DragList.vue // 拖拽列表组件
19. | |-- Icon.vue // 自定义图标组件
20. | |-- Login.vue // 登录
21. | |-- Markdown.vue // markdown组件
22. | |-- Premission.vue // 权限测试组件
23. | |-- Upload.vue // 图片上传
24. | |-- VueEditor.vue // 富文本编辑器
25. | |-- page // 主要路由页面
26. | |-- 403.vue //403
27. | |-- 404.vue //404
28. | |-- AllEmail.vue // 全服邮件
29. | |-- AllNotice.vue // 全服更新公告
30. | |-- AppleProp.vue // 申请道具邮件
31. | |-- BackDoor.vue // 系统后门
32. | |-- BanIP.vue // IP禁封
33. | |-- CDK\_Use.vue // 激活码使用情况
34. | |-- Center.vue // 用户中心
35. | |-- Channel.vue // 通道管理
36. | |-- CreateRoleLog.vue // 角色创建日志记录
37. | |-- Dashboard.vue // 系统首页
38. | |-- DataSource.vue // 数据源管理
39. | |-- GameTable.vue // 游戏管理
40. | |-- GiftTable.vue // 礼包管理(未)
41. | |-- GiftUpload.vue // 礼包导入
42. | |-- GoodFlowLog.vue // 物品流通日志
43. | |-- Login.vue // 用户登录
44. | |-- MoneyFlowLog.vue // 现金流日志管理
45. | |-- NewGiftTable.vue // 礼包导入信息管理
46. | |-- NewPropTable.vue // 道具导入信息管理
47. | |-- PlatformTable.vue // 平台管理
48. | |-- PlayerBanLog.vue // 玩家禁封日志
49. | |-- PlayerInfo.vue // 玩家信息查询
50. | |-- PlayerProhibitSpeakLog.vue // 玩家禁言日志
51. | |-- PlayerUpload.vue // 玩家信息导入
52. | |-- PropTable.vue // 道具管理(未)
53. | |-- PropUpload.vue // 道具信息导入
54. | |-- ReCharge.vue // 充值流水日志
55. | |-- Register.vue // 用户注册(未)
56. | |-- RightTable.vue // 权限管理
57. | |-- RoleLoginLog.vue // 角色登录日志
58. | |-- RoleTable.vue // 角色管理
59. | |-- SendEmail.vue // 发送邮件
60. | |-- SendNotice.vue // 发送广播
61. | |-- ServerTable.vue // 服务器树形管理
62. | |-- Shop.vue // 充值消费日志
63. | |-- SystemConfig.vue // 系统配置(未)
64. | |-- TouristIdSet.vue // 游客账号设置
65. | |-- UploadFile.vue // 文件中转
66. | |-- UserTable.vue // 用户管理
67. | |-- Xlsx.vue // Xlsx文件导入(未)
68. | |-- router // 路由
69. | |-- defaultRouter.js // 默认路由
70. | |-- dynamicRouter.js // 动态路由
71. | |-- index.js // 设置路由
72. | |-- store // vuex
73. | |-- actions.js //
74. | |-- getters.js //
75. | |-- index.js //
76. | |-- mutations.js //
77. | |-- UtilsJs // 工具
78. | |-- Utils.js // 工具
79. | |-- App.vue // 页面入口文件
80. | |-- main.js // 程序入口文件，加载各种公共组件
81. |-- .babelrc // ES6语法编译配置
82. |-- .editorconfig // 代码编写规格
83. |-- .gitignore // 忽略的文件
84. |-- index.html // 入口html文件
85. |-- package.json // 项目及工具的依赖配置文件
86. |-- README.md // 说明

## 后端目录结构

1. |-- log // 日志
2. |-- META-INF //
3. | |-- MANIFEST.MF //
4. |-- src // src
5. | |-- main // main
6. | |-- java // java
7. | |--com // Event Bus
8. | |--cdk // Event Bus
9. | |--app // Event Bus
10. | |--configLoader // 配置加载器
11. | |--controller // controller层
12. | |--dao // dao层
13. | |--entity // 实体类
14. | |--result // 返回消息
15. | |--service // server类
16. | |--timer // 定时器
17. | |--util // 工具类
18. | |--XssFilter // 跨站脚本攻击过滤器
19. | |--Main.java // 项目主入口
20. | |-- resources // resources
21. | |-- META-INF // META-INF
22. | |--MANIFEST.MF // MANIFEST.MF
23. | |-- static // static
24. | |--dist // 前端打包文件
25. | |--favicon.ico // 项目图标(暂)
26. | |-- templates // templates
27. | |-- banner.txt // banner.txt
28. | |-- logback.xml // logback.xml
29. | |-- webapp // webapp
30. | |-- WEB-INF // WEB-INF
31. | |--jsp // jsp
32. | |--index.jsp // index.jsp
33. |-- target // 打包文件夹
34. | |-- classes // classes
35. | |-- classes // classes
36. | |-- generated-sources // generated-sources
37. | |-- maven-archiver // maven-archiver
38. | |-- maven-status // maven-status
39. | |-- game-web-server.jar // 打包文件
40. |-- .gitignore //
41. |-- application.properties //外部配置文件
42. |-- gm.sql // 数据库生成文件
43. |-- info.txt // 信息文件
44. |-- pom.xml // maven配置文件
45. |-- README.md // 说明
46. |-- tlog.xml // 日志表字段说明
47. |-- 道具表示例.xls //道具表示例.xls
48. |-- 礼包表示例.xls // 礼包表示例.xls
49. |-- 执行当前目录game-web-server程序.bat // 打包文件执行命令

# 搭建开发环境

## 1.前端开发流程

1、安装node.js：

下载相应版本的node.js,下载地址：https://nodejs.org/en/download/，下载完双击安装，点击下一步直到安装完成

2、安装npm的国内镜像：

安装完成后，附件里选择命令提示符（或者在开始的搜索框里输入cmd回车调出命令面板）输入：node -v回车，出现相应版本证明安装成功， node环境已经安装完成，由于有些npm有些资源被屏蔽或者是国外资源的原因，经常会导致用npm安装依赖包的时候失败，所有我还需要npm的国内镜像---cnpm。在命令行中输入：npm install -g cnpm –registry=https://registry.npm.taobao.org回车，大约需要3分钟，如果报错或没反应，则卸掉node.js重新安装

3、安装全局vue-cli脚手架：

vue-cli脚手架用于帮助搭建所需的模板框架。输入命令：cnpm install -g vue-cli回车等待完成。

4、安装webpack：

//全局安装

cnpm install -g webpack

//安装到你的项目目录

cnpm install --save-dev webpack

5、安装项目所需的依赖包：

首先输入：cd 项目名回车，然后输入：cnpm install回车等待安装，安装完成之后，会在我们的项目目录文件夹中多出一个node\_modules文件夹，这里边就是我们项目需要的依赖包资源。安装完依赖包之后，就可以运行整个项目了。

6、测试环境是否搭建成功：

在cmd里输入：cnpm run dev回车，项目运行成功后，浏览器打开localhost:8080（如果浏览器没有自动打开，可以手动输入）查看项目效果。

7、打包：

在cmd里输入：cnpm run build回车，打包完成后会在项目根目录生成dist文件夹

## 2.后端开发流程

1、安装JDK1.8并配置好JAVA\_HOME环境变量.

2、安装maven依赖（jar包管理）

3、将打包好的前端文件dist放在\src\main\resources\static\目录下，

4、配置数据库源

在根目录的application.properties文件下添加

spring.datasource.url=jdbc:mysql://127.0.0.1:3306/gm?serverTimezone=Asia/Shanghai

spring.datasource.username=root

spring.datasource.password=123456

5、初始化springboot项目

运行后端springboot项目，浏览器打开localhost:8011/dist/index.html即可看到项目效果

6、打包:

在项目目录下用cmd命令行输入mvn package spring-boot:repackage

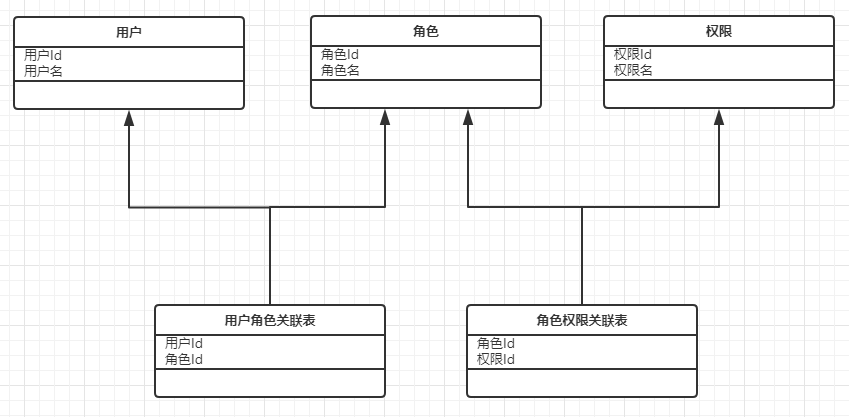
打包完成后会在项目根目录下的target文件夹下生成game-web-server.jar文件

# 功能开发

## 权限系统

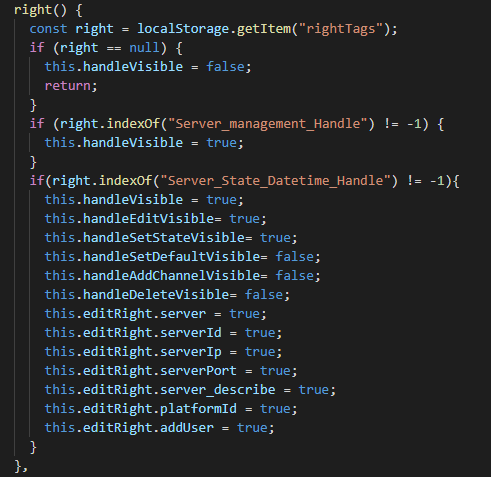
权限控制系统采用了类似RBAC模型

系统中包括用户、角色、权限模块，一个用户拥有若干角色，每一个角色拥有若干权限。一般是多对多的关系



权限绑定前端的路由系统，前端的路由采用了动态路由，每个路由都有自己的权限标识，只有当用户拥有该权限时才会添加该路由，同时主界面的菜单栏也根据路由动态的显示和隐藏。

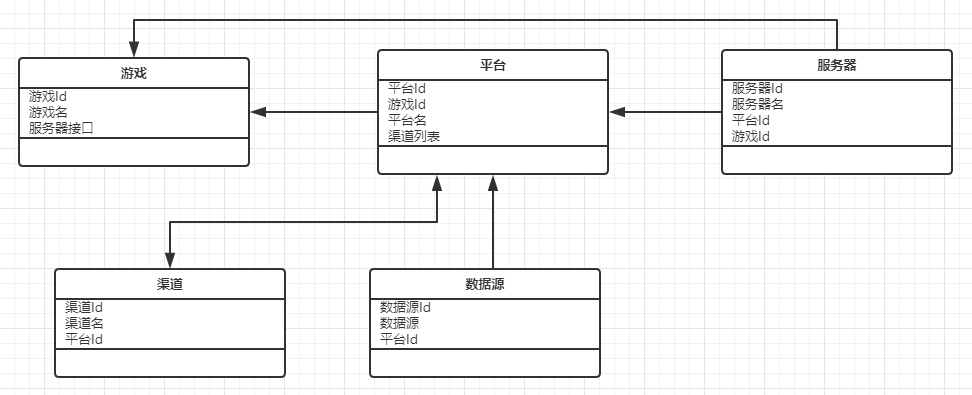
操作权限的判定需要在组件页面里动态实现显示或者隐藏，下边是操作权限实现的一个示例：



## 区服管理系统

区服管理主要包括游戏管理，平台管理，服务器管理，渠道管理，数据源管理

一个游戏有若干个平台，一个平台有若干个服务器，一个平台有若干个渠道，都是一对多的关系，一个平台有且只有唯一的一个数据源，一个游戏有且只有一个服务器列表接口，关系图如下：



其中一个平台绑定一个角色Id，系统通过判断用户所拥有的角色Id来判断用户对应那个平台，再通过平台来判断对应哪个游戏，哪些服务器。

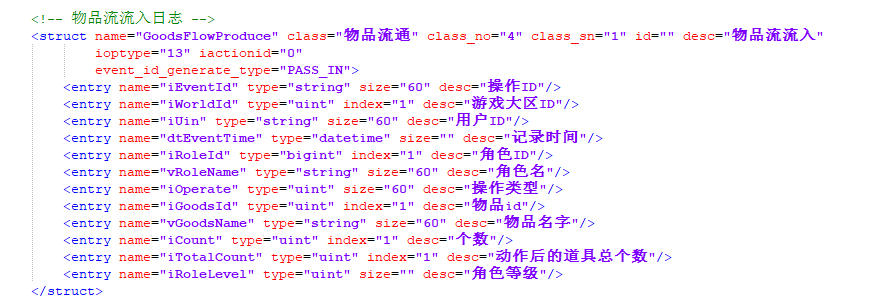
用户进入系统后可以自行切换已关联的游戏，切换游戏后，用户将只能看到当前游戏下，自己拥有的平台的数据。

## 日志系统

日志系统主要包括物品流通日志、现金流日志、角色创建日志、角色登录日志、充值流水日志、充值消费日志。

日志系统的数据全部来自平台数据源接口的数据，不同的平台有不同的数据源，具体数据源需要在区服管理里的数据源管理页面中去配置。

不同的日志数据对应着数据源数据库里的一张数据表，除了平台和服务器字段，其他的字段名均储存在一个xml文件中，前端通过接口获取到xml文件的json数据，再把不同日志的字段名映射到日志页面中。



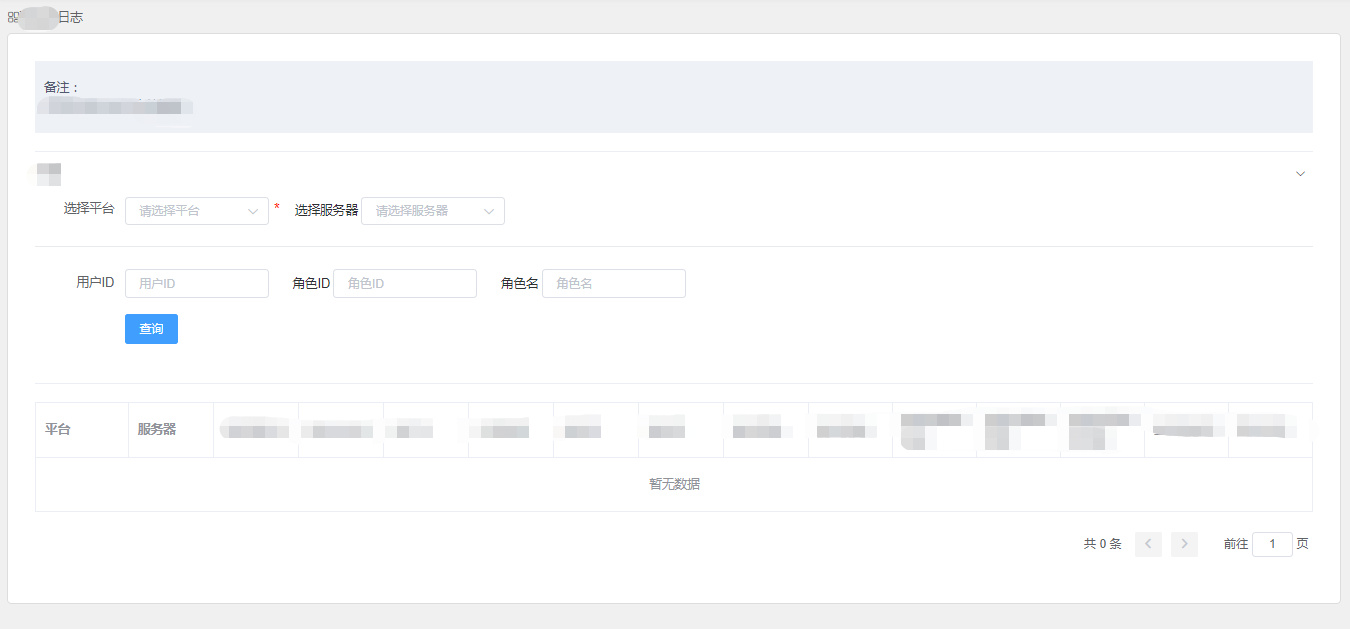
日志系统的所有接口都写在后端控制层的PlayerLogTableController.class文件中，其中

getLogXml方法是获取日志字段的接口，下边将展示添加新日志的大致步骤：

### 添加前端日志组件

需要有平台筛选框、服务器筛选框、角色信息筛选框、查询按钮、数据表格、分页器

效果图如下：



### 获取字段信息

通过接口/api/log/getLogXml获取数据源数据库表字段，获取对应的数据表的字段，将字段逐一映射到表格中。

### 添加获取表格数据接口

在后端控制层的PlayerLogTableController.class文件中添加新的获取数据接口，首先通过游戏Id和平台Id获取相应的数据源地址、数据源用户名和数据源密码，再通过数据源信息创建新的数据源连接，通过相应的sql语句查询对应数据库表的数据，最后添加数据筛选和数据分页。

### 前端组件通过接口显示数据

前端组件通过 axios 的post请求从接口获取数据并将数据显示在表格中。

### 添加路由和权限

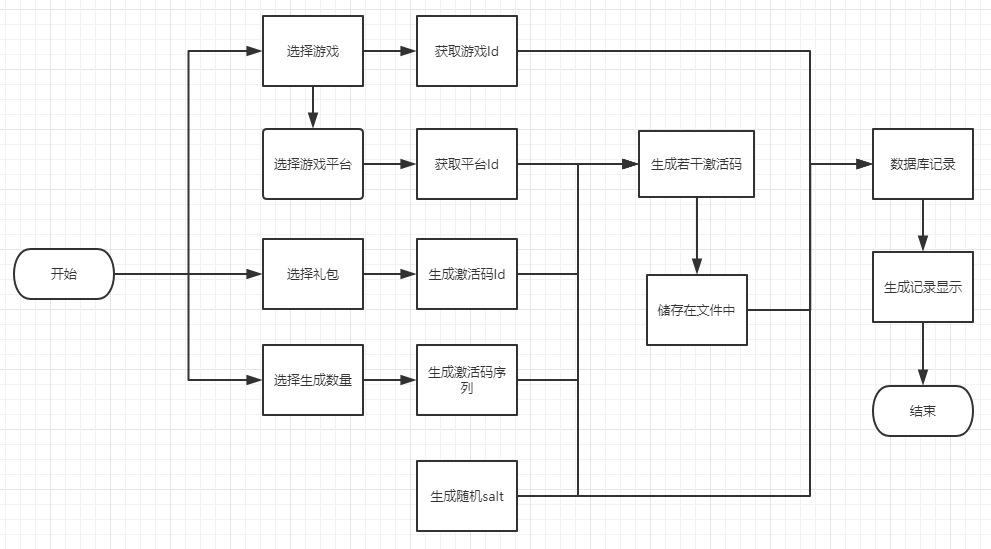
在前端的vue Router中的dynamicRouter.js中添加权限路由，示例：

setRouterItem("/Shop","充值消费","Shop\_View","Shop.vue",children,right);

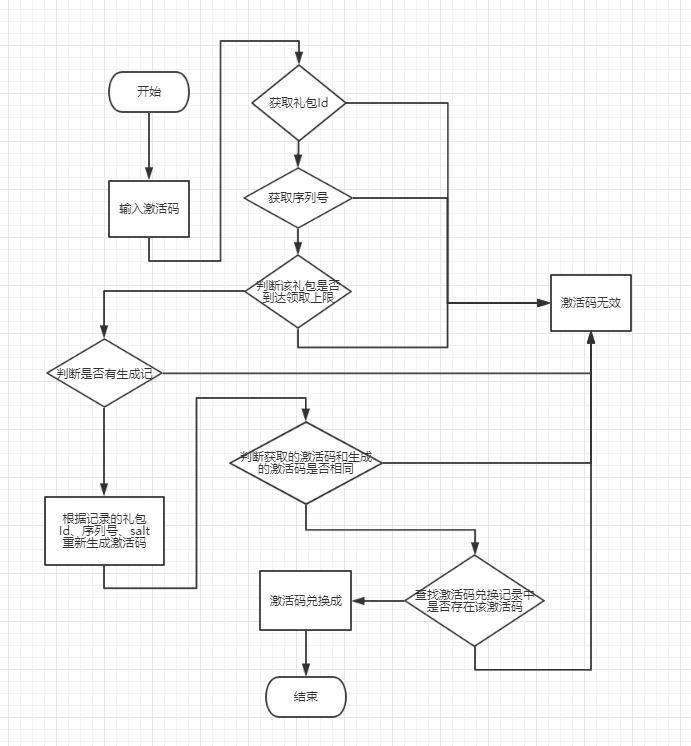
最后在权限管理中添加相应的权限即可。

## 礼包激活码

激活码生成流程图：



激活码兑换流程图：



### 激活码生成

激活码的生成需要获取游戏id、平台Id、礼包Id以及生成激活码的数量，在前端向后端传输数据时需要进行MD5加密，通过生成数量确认每个激活码对应且唯一的序列，再结合礼包Id、平台Id和随机生成的salt通过位算法生成激活码，将激活码逐一储存在文件中，最后生成一条保存礼包Id、游戏Id、平台Id、salt、开始序列、结束序列、文件地址的数据库记录。

### 激活码兑换

兑换激活码时，首先判读激活码的格式和能否从激活码中读取礼包Id和序列号，再通过礼包Id查询当前玩家角色是否超过该礼包的领取限制、若没有超过上限，继续判断是否有相同游戏平台相同礼包Id和相同序列范围的生成记录，如果有则继续判断使用该序列号、礼包Id、平台Id、salt和相同算法生成的激活码和要兑换的激活码是否相同，如果相同则判断再激活码兑换记录中是否有该激活码的兑换记录，如果没有则说明该激活码有效，兑换成功。

## 接口开发

后端接口的实现主要在Controller层

### 定义控制器

使用 @RestController注解 定义一个 Controller 控制器

@RestController注解相当于@ResponseBody ＋ @Controller

大多数情况我们只需要返回json格式的字符串而不需要返回json对象，所以加上@Responsebody就行了

### 设置接口url

使用 @RequestMapping注解 来映射 Request 请求与处理器

### 获取数据

使用@requestBody注解常用来处理application/json类型。

当使用表单上传数据时，使用@RequestParam注解获取表单参数

当使用get请求时，使用（HttpServletRequest request）获取接收到的参数

## 前端组件开发

系统里的各个页面都在单页面里边，都统称为组件(Component)，所有的 Vue 组件同时也都是 Vue 的实例，所以可接受相同的选项对象 (除了一些根级特有的选项) 并提供相同的生命周期钩子。

前端所有的组件都放在src/component/目录下，其中common文件夹下是一些公用组件，包括系统的父组件，头部组件，菜单栏组件和标签页组件，和一个bus.js（vue的组件通讯Event Bus）。Example文件夹下是一些[vue-manage-system](https://github.com/cdk-king/vue-manage-system-master)模板下的一些示例组件。Page文件夹下是系统各个功能的组件，新开发的组件统一写在page文件夹下。

## 前端组件通讯

前端组件间的通讯主要用了官方推荐的Event Bus，核心文件是common文件夹下的bus.js。

实现通讯的过程是注册Event Bus事件和广播消息

### 导入Event bus

import bus from "../common/bus";

### 注册事件

created() {

bus.$on("changeGameId",function(obj){

this.form.platformId = "";

this.getData();

}.bind(this))

},

### 触发事件

bus.$emit('changeGameId', {

gameId:command.id,

message:"已选择游戏："+command.gameName

});

### 销毁事件

beforeDestroy () {

bus.$off('changeGame');

},

**这个$on事件是不会自动清楚销毁的，需要我们手动来销毁**

# 构筑生产

## 前端

// 执行构建命令，生成的dist文件夹放在服务器下即可访问

npm run build

## 后端

在项目目录下用cmd命令行输入mvn package spring-boot:repackage

打包完成后会在项目根目录下的target文件夹下生成game-web-server.jar文件