**https://gitee.com/liuge1988/kitty-ui**

**简介**

Kitty 前端，基于 Vue + Element 实现的权限管理系统。

[**kitty**](https://gitee.com/liuge1988/kitty)

基于Spring Boot、Spring Cloud、Vue.js 、Element UI实现，采用前后端分离架构的权限管理系统，JAVA快速开发平台。

[**kitty-ui**](https://gitee.com/liuge1988/kitty-ui)

Kitty 前端，基于 Vue + Element 实现的权限管理系统。

#### 前端安装

1. 下载源码

git clone <https://gitee.com/liuge1988/kitty-ui.git>

1. 编译代码

进入项目根目录，执行 npm install, 下载和安装项目相关依赖包。

1. 启动系统

执行 npm run dev 命令，启动项目，通过 [http://localhost:8090](https://gitee.com/link?target=http%3A%2F%2Flocalhost%3A8090) 访问。

1. 项目打包

执行 npm run build 命令，启动打包，完成之后会生成 dist 目录。

1. Mock 开关

通过修改src/mock/index.js中的openMock变量，可以一键开启或关闭Mock功能。

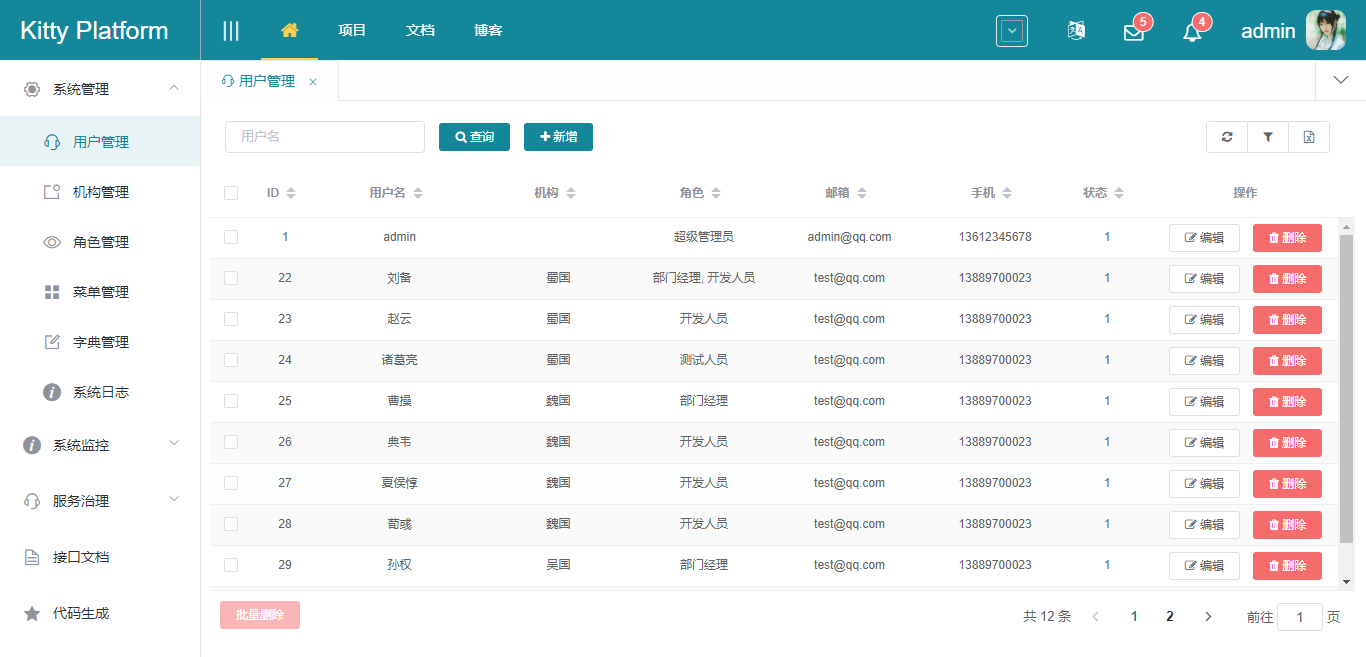
1. 修改配置

如果想自定义端口（默认是8090），可以修改 config/index.js 下的 port 属性。

后台接口和备份服务器地址配置在 src/utils/global.js，如有修改请做相应变更。







# mango

### 项目介绍

Mango后台权限管理系统是基于Spring Boot、Spring Cloud、Vue.js 、Element UI等主流前后端技术，采用前后端分离架构实现的权限管理系统。也是一款采用当前主流技术实现的界面优雅、架构优良、代码简洁、注释完善、基础功能相对完整的JAVA EE快速开发平台，前后端开发人员都可以以此为范例从中学习和汲取技术知识，也可以基于此系统开发和实现具体生产项目。

此项目为笔者和清华大学出版社合作出版的专业书籍《Spring Boot+Spring Cloud+Vue+Element项目实战：手把手教你开发权限管理系统》的配套附件，附件内容包含书籍项目相关源码、数据库脚本、项目安装和运行教程等。



### 功能列表

✔ 系统登录：系统用户登录，系统登录认证（token方式）

✔ 用户管理：新建用户，修改用户，删除用户，查询用户

✔ 机构管理：新建机构，修改机构，删除机构，查询机构

✔ 角色管理：新建角色，修改角色，删除角色，查询角色

✔ 菜单管理：新建菜单，修改菜单，删除菜单，查询菜单

✔ 字典管理：新建字典，修改字典，删除字典，查询字典

✔ 配置管理：新建配置，修改配置，删除配置，查询配置

✔ 登录日志：记录用户的登录日志，查看系统登录日志记录

✔ 操作日志：记录用户的操作日志，查看系统操作日志记录

✔ 在线用户：根据用户的登录状态，查看统计当前在线用户

✔ 数据监控：定制Druid信息，提供简洁有效的SQL数据监控

✔ 聚合文档：定制Swagger文档，提供简洁美观的API文档

✔ 备份还原：系统数据备份还原，一键恢复系统初始化数据

✔ 主题切换：支持主题切换，自定主题颜色，实现一键换肤

✔ 服务治理：集成Consul注册中心，实现服务的注册和发现

✔ 服务监控：集成Spring Boot Admin，实现全方位的服务监控

✔ 服务消费：集成Ribbon、Feign，实现服务调用和负载均衡

✔ 服务熔断：集成Hystrix、Turbine，实现服务的熔断和监控

✔ 服务网关：集成Spring Cloud Zuul，实现统一API服务网关

✔ 链路追踪：集成Sleuth、Zipkin，实现服务分布式链路追踪

✔ 配置中心：集成Cloud Config和Bus，实现分布式配置中心

##### 编译源码

在mango-ui目录下打开CMD终端，执行 npm install, 下载和安装项目依赖包。

##### 启动系统

执行 npm run dev 命令，启动项目，启动之后通过 [http://localhost:8080](https://gitee.com/link?target=http%3A%2F%2Flocalhost%3A8080) 访问。

##### 项目打包

执行 npm run build 命令，进行前端项目打包，打包完成之后会生成 dist 目录。 将生成的目录直接放置到如Tomcat之类的WEB服务器，启动服务即可访问。

##### Mock 开关

本系统采用前后端分离架构，前端若开启Mock模块，可以模拟大部分接口数据。 通过修改src/mock/index.js中的openMock变量，可以一键开启或关闭Mock功能。