# 系统介绍

## 系统简介

Kitty是基于Spring Boot、Spring Cloud、Vue.js、Element实现的权限管理系统。

Kitty的目标是搭建出一套简洁易用的快速解决方案，给用户学习和开发提供帮助。

个人博客提供本项目开发过程同步系列教程文章，手把手的教你如何开发同类系统。

## 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
* ✔ 系统日志：记录用户操作日志，查看系统执行日志记录
* ✔ 数据监控：定制Druid信息，提供简洁有效的SQL监控
* ✔ 聚合文档：定制在线文档，提供简洁美观的API文档
* ✔ 备份还原：系统备份还原，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，一键换肤
* ✔ 服务治理：集成Consul注册中心，实现服务注册和发现
* ✔ 服务监控：集成Spring Boot Admin，实现服务监控
* ✔ 服务消费：集成Ribbon、Feign，服务调用和负载均衡
* ✔ 服务熔断：集成Hystrix、Turbine，实现熔断和监控
* ✔ 服务网关：集成Spring Cloud Zuul，实现API网关
* ✔ 链路追踪：集成Sleuth、Zipkin，实现分布式链路追踪
* ✔ 配置中心：集成Config、Bus，实现分布式配置中心
* ✔ 代码生成：提供代码生成器，最大化的降低代码开发量
* ...

## 在线演示

演示地址：<http://139.196.87.48:9002/kitty>

用户名：admin 密码：admin

温馨提示： 有在演示环境删除数据的童鞋们，如果可以的话，麻烦动动小指，点击头像面板， 选择 -> 备份还原 帮忙恢复到默认备份数据，方便后来的童鞋查看，谢谢啦。

## 技术交流

为了方便大家提问和技术交流，整了个QQ群，欢迎童鞋们加入。

QQ技术交流群： 429854222

## 博客教程

本人技术博客提供同步系列文章教程, 讲解了Kitty从零开始搭建的全过程：

[Vue.js + Element UI 实现权限管理系统 前端篇](https://www.cnblogs.com/xifengxiaoma/p/9533018.html)

[Spring Boot + Spring Cloud 实现权限管理系统 后端篇](https://www.cnblogs.com/xifengxiaoma/p/9474953.html)

# 软件架构

## 后端架构

### 开发环境

* IDE : eclipse 4.x
* JDK : JDK1.8.x
* Maven : Maven 3.5.x
* MySQL: MySQL 5.7.x

### 技术选型

* 核心框架：Spring Boot 2.x
* 服务治理：Spring Cloud Finchley
* 安全框架：Spring Security 5.x
* 视图框架：Spring MVC 5.x
* 持久层框架：MyBatis 3.x
* 数据库连接池：Druid 1.x
* 日志管理：SLF4J、Log4j

### 项目结构

* kitty-common： 公共代码模块，主要放置一些工具类
* kitty-core： 核心代码模块，主要封装公共业务模块
* kitty-admin： 后台管理模块，包含用户、角色、菜单管理等
* kitty-backup： 系统数据备份备份模块，可选择独立部署
* kitty-monitor： 系统监控服务端，监控Spring Boot服务模块
* kitty-producer： 服务提供者示例，方便在此基础上搭建模块
* kitty-consumer： 服务消费者示例，方便在此基础上搭建模块
* kitty-hystrix： 服务熔断监控模块，收集汇总熔断统计信息
* kitty-zuul： API服务网关模块，统一管理和转发外部调用请求
* kitty-config： 配置中心服务端，生成GIT配置文件的访问接口
* kitty-consul： 注册中心，安装说明目录，内附安装引导说明
* kitty-zipkin： 链路追踪，安装说明目录，内附安装引导说明
* config-repo： 配置中心仓库，在GIT上统一存储系统配置文件
* kitty-pom： 聚合模块，仅为简化打包，一键执行打包所有模块

## 前端架构

### 开发环境

* IDE : VS Code 1.27
* NODE: Node 8.9.x
* NPM : NPM 6.4.x

### 技术选型

* 前端框架：Vue 2.x
* 页面组件：Element 2.x
* 状态管理：Vuex 2.x
* 后台交互：axios 0.18.x
* 图标使用：Font Awesome 4.x

### 项目结构

kitty-ui

* assets： 图标、字体、国际化信息等静态信息
* components： 组件库，对常用组件进行封装
* http： 后台交互模块，统一后台接口请求API
* i18n： 国际化模块，使用Vue i18n进行国际化
* mock： Mock模块，模拟接口调用并返回定制数据
* permission： 权限控制模块，处理权限认证逻辑
* router： 路由管理模块，负责页面各种路由配置
* store： 状态管理模块，提供组件间状态共享
* utils： 工具模块，提供一些通用的工具方法
* views： 页面模块，主要放置各种页面视图组件

# 安装教程

## 后端安装

1. 下载源码

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

1. 导入工程

使用 Eclipse导入 Maven 项目，在此之前请确认已安装 JDK 和 Maven 工具。

1. 编译代码

找到 kitty-pom 工程的 pom.xml，执行 maven clean install 命令编译一键打包。

一般来说不会有什么问题，如果还是编译不成功，可以按照优先级逐个编译试一试。

1. 导入数据库

新建 kitty 数据库，导入 kitty-admin 工程 doc/kitty.sql 脚本，导入初始化数据库。

修改 kitty-admin 下 application.yml 中的数据库连接和账号密码为自己的数据库配置。

修改 kitty-backup 下 application.yml 中的数据库连接和账号密码为自己的数据库配置。

1. 启动系统

基础必需模块（注册中心：kitty-consul，服务监控：kitty-monitor）

找到 kitty-consul 工程，根据安装说明安装注册中心，完成后执行 consul agent -dev 启动。

找到 kitty-monitor 工程下的 KittyMonitorApplication， 启动项目，开启服务监控。

权限管理模块（权限管理：kitty-admin，备份还原：kitty-backup）

找到 kitty-admin 工程下的 KittyAdminApplication， 启动项目，开启权限管理系统服务。

找到 kitty-backup 工程下的 KittyBackupApplication.java，启动项目，开启备份还原服务。

其他模块根据各自需要选择性启动，模块依赖可以参见我的博客同步教程...

注意事项：

如果需要链路追踪服务，需要安装zipkin，找到 kitty-zipkin 工程，根据安装说明安装zipkin。

如果需要配置中心服务，需要安装rabbitMQ，找到 kitty-config 工程，根据安装说明安装rabbitMQ。

## 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](http://localhost:8090/) 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。

## 代码生成

代码生成器放在独立项目，详情请见具体项目主页说明。

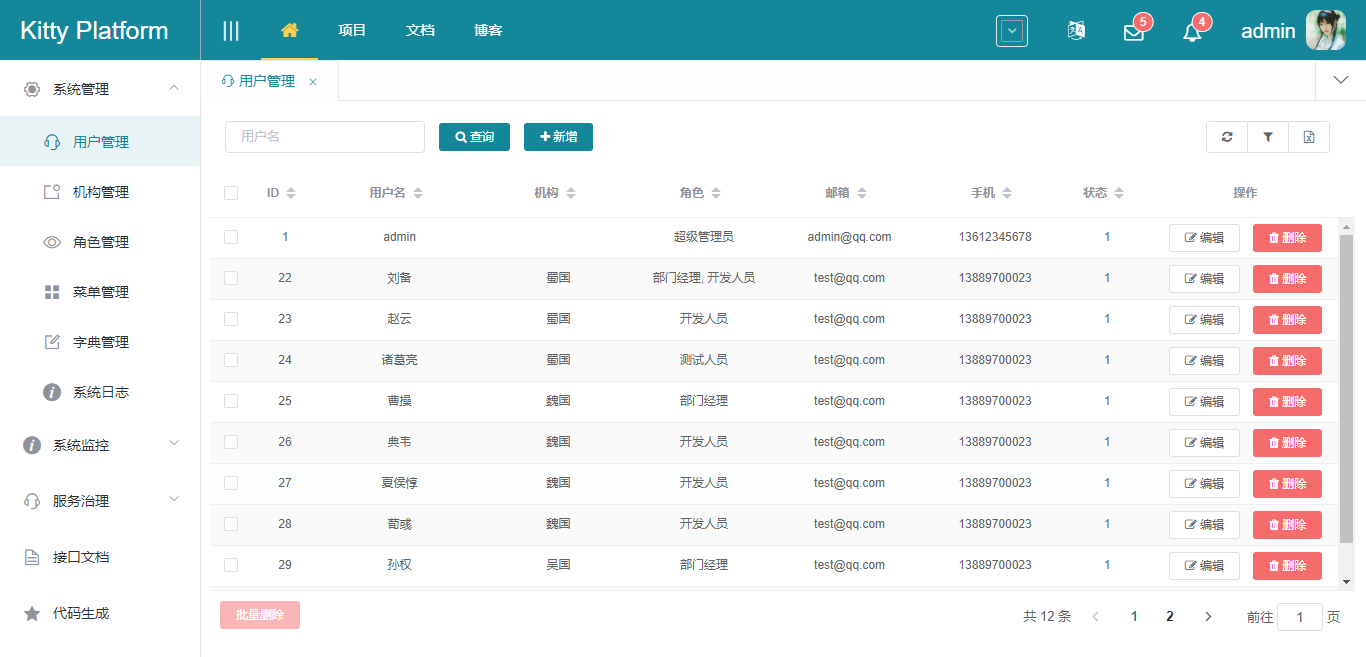
项目地址：<https://gitee.com/liuge1988/kitty-generator>

# 系统展示

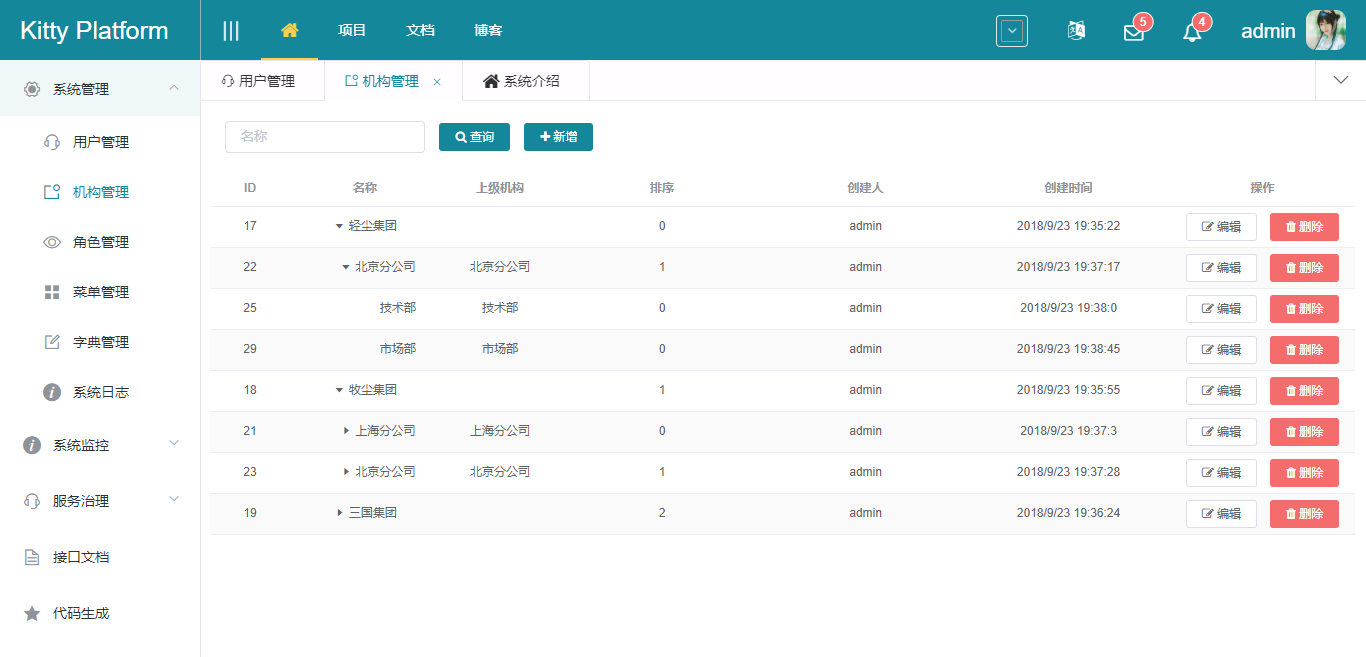
## 登录界面



## 用户管理



## 机构管理



## 角色管理



## 菜单管理



## 字典管理



## 系统日志

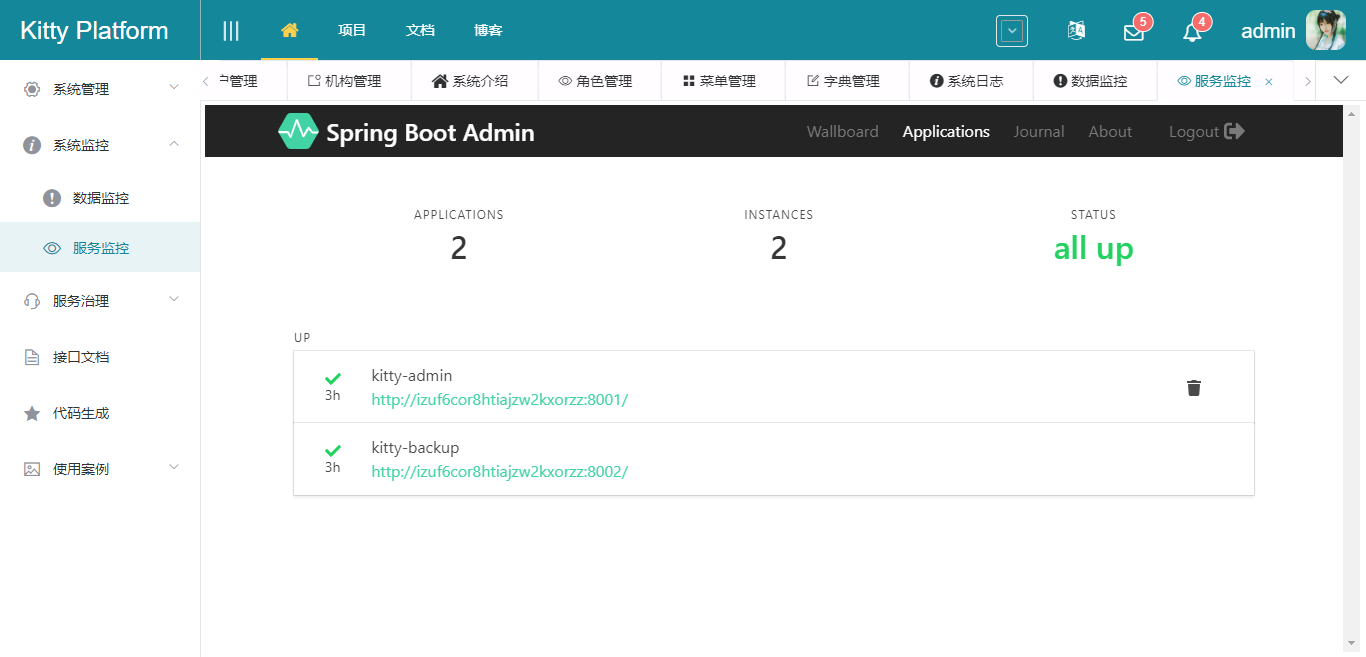


## 数据监控

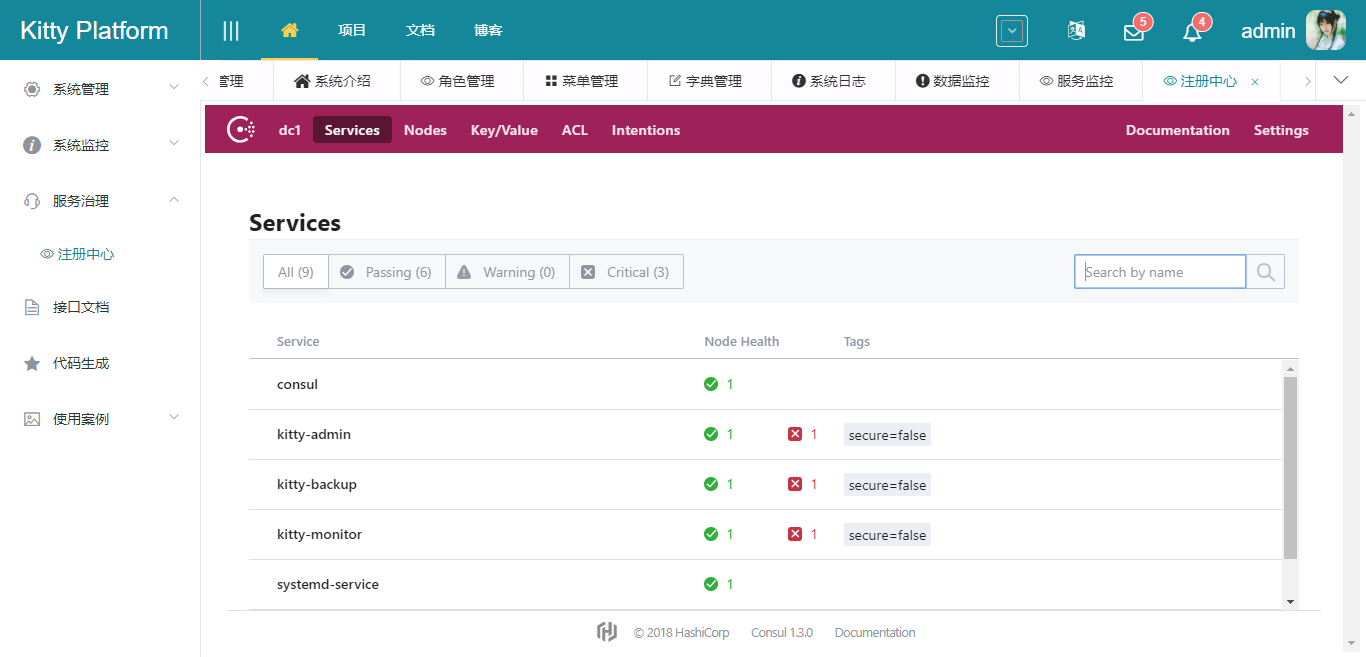
用户名：admin, 密码:admin，即服务端配置的密码



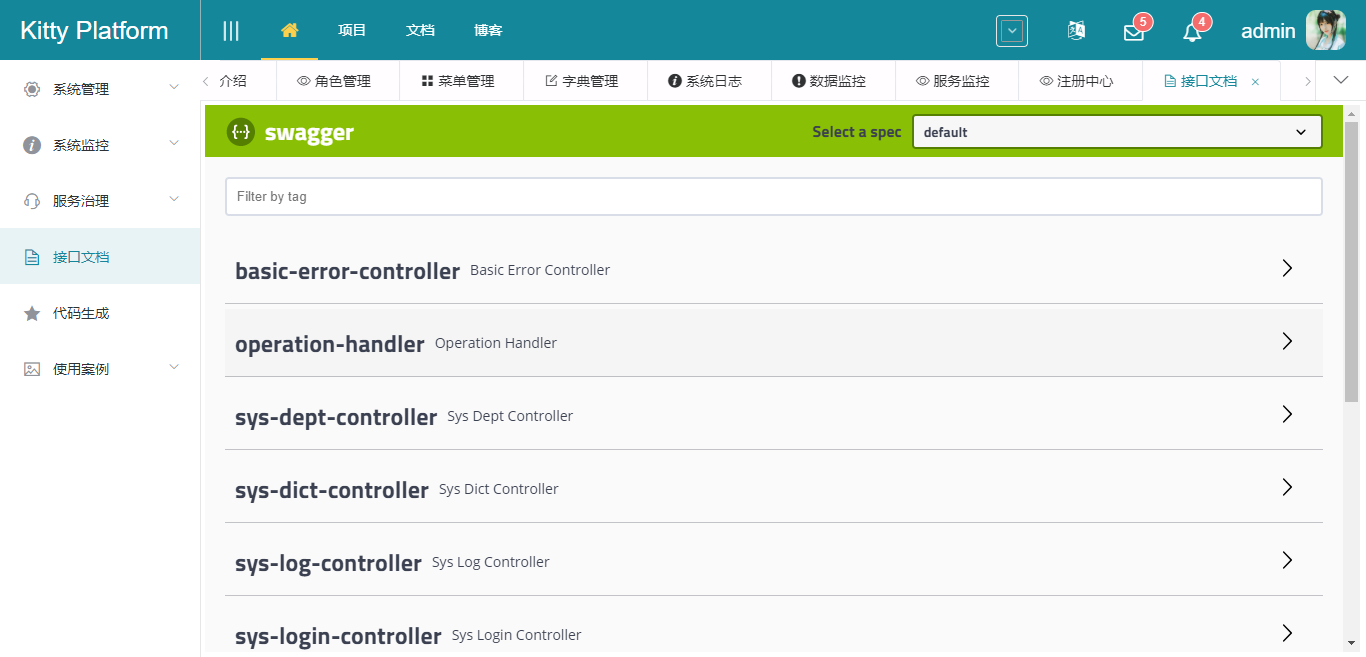
## 服务监控



## 注册中心



## 接口文档



## 代码生成



## 主题切换

