****

**学 位 论 文 诚 信 声 明 书**

本人郑重声明：所呈交的学位论文（设计）是我个人在导师指导下进行的研究（设计）工作及取得的研究（设计）成果。除了文中加以标注和致谢的地方外，论文（设计）中不包含其他人或集体已经公开发表或撰写过的研究（设计）成果，也不包含本人或其他人在其它单位已申请学位或为其他用途使用过的成果。与我一同工作的同志对本研究（设计）所做的任何贡献均已在论文中做了明确的说明并表示了致谢。

申请学位论文（设计）与资料若有不实之处，本人愿承担一切相关责任。

学位论文（设计）作者签名： 日期：

**学 位 论 文 知 识 产 权 声 明 书**

本人完全了解学校有关保护知识产权的规定，即：在校期间所做论文（设计）工作的知识产权属西安科技大学所有。学校有权保留并向国家有关部门或机构送交论文的复印件和电子版。本人允许论文（设计）被查阅和借阅；学校可以公布本学位论文（设计）的全部或部分内容并将有关内容编入有关数据库进行检索，可以采用影印、缩印或其它复制手段保存和汇编本学位论文。

保密论文待解密后适用本声明。

学位论文（设计）作者签名： 指导教师签名：

年 月 日

论文题目：银行维修系统后台子系统

专业：电子信息工程

学生：康旺 签名：\_\_\_\_\_\_\_\_\_

指导教师：黄健 签名：\_\_\_\_\_\_\_\_\_

摘 要

本系统使用的是B/S架构，使用Layui作为前端渲染框架，使用jboot框架作为后端处理框架，使用mysql作为数据的存储，通过Redis作为系统的缓存，缓存验证码和权限信息，使用jdk1.8版本，在windows环境下开发，开发出一套B/S的银行维修子系统。

本次课题设计主要有以下六个部分：维保机构信息管理、维修工信息管理、供应商信息管理、仓库进出库管理、维修订单管理、权限管理等管理模块组成，通过各个模块的协调工作，实现整体的信息化，完成银行维修子系统项目。

关键字：系统设计，信息化,信息管理，维保机构

Title:Background subsystem of bank maintenance system

Major: electronic information engineering

Student: Kang Wang signature：\_\_\_\_\_\_\_\_\_

Instructor: Huang Jian signature:\_\_\_\_\_\_\_\_\_\_

ABSTRACT

The system uses B / S architecture, Layui as front-end rendering framework, jboot framework as back-end processing framework, mysql as data storage, Redis as system cache, cache verification code and permission information. Using jdk1.8 version, a set of bank maintenance subsystem is developed in windows environment.

The design of this project is mainly composed of the following six parts: maintenance organization information management, maintenance worker information management, supplier information management, warehouse entry and exit management, maintenance order management, authority management and other management modules, such as maintenance organization information management, maintenance worker information management, supplier information management, warehouse entry and exit management, maintenance order management, authority management and so on. Through the coordination of each module, the overall informatization is realized, and the bank maintenance subsystem project is completed.

KEY WORDS: information, maintenance organization, information management, system design

目录

[摘 要 2](#_Toc24338)

[1前 言 1](#_Toc15382)

[1.1 课题研究的背景 1](#_Toc11059)

[1.2开发的目标及意义 1](#_Toc17487)

[1.2.1系统开发的目标 1](#_Toc6864)

[1.2.1系统开发的意义 2](#_Toc29247)

[1.3研究思路和研究方法 2](#_Toc22994)

[1.3.1维修工设计 2](#_Toc7528)

[1.3.2维保机构设计 2](#_Toc22502)

[1.3.3订单管理 3](#_Toc31767)

[1.3.4系统开发及应用模式 3](#_Toc10808)

[1.3.5开发工具选择 3](#_Toc20140)

[1.3.6论文整体框架 3](#_Toc6137)

[2开发环境以及技术 4](#_Toc3713)

[2.1开发环境配置 4](#_Toc4171)

[2.2技术概述 5](#_Toc1209)

[2.2.1 web概述 5](#_Toc31806)

[2.2.2 Java 5](#_Toc4273)

[2.2.3 Layui 6](#_Toc13827)

[2.2.4 Mysql 6](#_Toc25335)

[2.2.5 Jboot 7](#_Toc30348)

[2.2.6 Ajax介绍 7](#_Toc22764)

[2.2.7 shiro介绍 8](#_Toc21792)

[2.2.8 CS模式 8](#_Toc18043)

[2.3小结 8](#_Toc4278)

[3系统设计 9](#_Toc9871)

[3.1 系统可行性分析 9](#_Toc22808)

[3.1.1技术可行性 9](#_Toc12332)

[1.1硬件配置可行性 9](#_Toc15895)

[1.2操作可行性 9](#_Toc32083)

[1.3系统需求分析 9](#_Toc23892)

[3.2系统架构分析 9](#_Toc6692)

[3.2.1它的优势： 10](#_Toc27502)

[3.2.2系统整体功能结构 12](#_Toc17871)

[3.3系统逻辑功能分析 12](#_Toc23451)

[3.3.1系统逻辑结构 13](#_Toc15449)

[3.3.1系统的功能要求 18](#_Toc16621)

[3.4系统性能分析 18](#_Toc22335)

[3.5服务器安全 19](#_Toc7887)

[3.5.1安全设计方案 19](#_Toc1148)

[4主要程序设计说明 21](#_Toc19084)

[4.1登录页面 21](#_Toc20099)

[4.2主页 21](#_Toc19927)

[4.3维修工管理 22](#_Toc17697)

[4.4维保机构 24](#_Toc31395)

[4.5出入库模块 26](#_Toc22123)

[4.5.1供应商模块 26](#_Toc8924)

[4.5.2商品管理模块 28](#_Toc24609)

[4.5.3仓库模块 30](#_Toc15032)

[4.5.4出入库 31](#_Toc17713)

[4.6订单管理 31](#_Toc8888)

[4.7权限管理模块 32](#_Toc28732)

[4.8角色管理 32](#_Toc32405)

[4.9日志 33](#_Toc13127)

[4.10退出系统 34](#_Toc26795)

[5系统测试 35](#_Toc3486)

[5.1功能测试 35](#_Toc28499)

[5.2可用性测试 36](#_Toc23120)

[5.3安全性测试 36](#_Toc19446)

[结论 37](#_Toc30306)

[致 谢 38](#_Toc29561)

[参考文献 39](#_Toc22480)

# 1前 言

本次课题设计以银行监控维修为背景，设计一个维修订单处理的后台子系统。全文分为4章，主要包括：绪论、开发环境以及技术、系统设计、主要程序设计说明，系统测试、结论、致谢、参考文献。

## 1.1 课题研究的背景

网络已经步入人们的生活，并产生了深远化的影响，它成为人们生活的重要部分，比如支付、购物等方便，通过网络生活更加方便和快捷。并且形成了信息共享，资源共享的大环境，信息对于每个人来说也非常重要，也在企业发展中发挥着它的优势和作用。它可以保证系统可以完整的工作，效率更加高效。对于企业来说，系统独特的优势。如今，已经是非常重要的工具了。近年来，各类竞争越来越激烈，一个优质的服务，快捷的效率这些对于维修和被维修来说，都显得更加的重要。回望过去，人手工的方式，对事情的处理，无论是效率，资源分配,出错概率都突显出它自身的缺点，实时性无法满足如今大量数据的数据和服务。为了改掉传统，与时俱进，使得烦恼的工作将人解放出来，同时对于企业的服务的时效性和提供企业核心竞争力有强烈的作用。随着当今社会移动互联网的飞速发展，维修的自动化，方便化，管理性等发生改进已刻不容缓。也就是说，它和过去的方式已经发生了巨大变化和改进，化繁为简。

在我国，改革开发40周年的大环境下，人们生活水平不断提高，全民也在享受改革开发40周年的果实。口袋也有钱了，银行成立人民稳定投资的一种方式，安全在银行中也占有很大的地位，所以银行监控的及时维修也变的更加重要了。并且为了方便人们可以方便的存取钱，ATM机的数量也在逐渐增加，并且管理和维护存在很大的问题，并且极不方便。在网络快速共享的特性，可以成为及时获知维修信息的重要手段。2015年7月以来，国务院国家倡导互联网+，在全社会发起了互联网热,人们通过互联网的思维共享实践成果，增加收入，推动经济形态的不断演变，带动人民生活水平的改善，使用互联网通过利用现代的信息技术以及互联网通过与各个行业进行结合。

## 1.2开发的目标及意义

### 1.2.1系统开发的目标

本系统将传统的维修系统与互联网相结合，方便银行对监控的维护和监控，也方便了维修工对订单的及时获知和处理，主要是以节省维修工和维保机构的工作效率以及加快维保的速度。通过有效的信息化管理，改变服务方式以及工作质量，满足各种环境下的需求，提升维修服务水平。平台具有先进性、安全性、易维护、可扩展满足不同的需求。通过快速的方式进行信息管理和订单管理，解放劳动力。并且解决人来处理，易出错的弊病。实现网上注册，网上下单，网上节点，网上支付，网络管理。

### 1.2.1系统开发的意义

人们生活水平的提供，节奏加快，特别是今年来移动互联网技术的快速发展，人们越来对效率有更高的要求，慢在如今更加显得格格不入，当银行发现自己的监控发生故障，可以最快的方式将信息传递给维修工，并且可以快速的找到空闲的维修工并且可以及时的进行维修，当维修工闲的时候，如何更加快速的得到满足需要维修订单的需求，这个时候维保系统就就是一个不错的选择，它做到了及时发布订单和处理订单，节约了时间，不同于传统的方式，有订单，找谁修，谁来修的问题，也使得资源更加的合理分配，人员的合理分配。

并且计算机可以方便的进行统计和使用，只需要或者不需要培训便可以进行使用，节约了人力和物力，并且如今计算机支出的形式也越来越多样，从文字到图片，在到视频。并且也支持各种统计方式，通过扇形图、条形图、折线图等方式进行统计和展现。以满足不同的需求，节约了统计人力、物力以及财力。

## 1.3研究思路和研究方法

论文采用了总体分析与规划、分布实施、web技术与网络技术相结合，通过互联网技术，实现信息共享。实践与学习相结合的方式，首先查看已有后台管理系统的需求，以及功能模块的实现步骤，功能需求，然后根据自己的设计需求分析自己的系统实现，并借鉴别人系统优秀的部分，并完成自己的原型设计，分析各个部分的技术以及展现方式，布局设计，数据的展示位置等综合考虑。

### 1.3.1维修工设计

在进行基于web的银行后台管理系统中，维修工作为订单的执行者，对订单的处理者，他是比较重要的一部分，他必须符合一般要求，并且需要符合系统的要求。考虑的因素有：管理员可以进行对其进行各种操作，首先就是增删改查，在用户进行接单之后，需要将其状态进行管理，在已经接单的情况下不可以重复接单，管理员可以指定员工进行订单处理。

### 1.3.2维保机构设计

维保机构是被维修的一方，它需要加入到系统中，并且可以对其信息进行修改设置，管理员可以对其进行增删改查操作，也可以对网点进行操作，可以根据机构对网点进行设置。维保机构也可以对自己的网点进行操作，那么就需要系统具有一定的易操作性，会用电脑的人，就可以具备这项技能，而不是需要别人来培训，就可以熟练的操作系统。

### 1.3.3订单管理

这个操作的大部分都是维保机构的操作，所以它最重要的是易操作，不需要特殊的培训，它也是维修工和维保机构进行维修下单的维修操作中重要的一部分，维保机构有维修需求之后，通过订单项进行下单处理，维修工可以对订单进行处理，这一部分主要是在手持设备完成的。

### 1.3.4系统开发及应用模式

系统是基于web的开发的系统，充分的利用互联网Internet技术，客户端使用简单的浏览器，对于使用者不需要特殊的客户端设置，并且对于服务端管理和维护也比较的方便，对于硬件的要求比较低。页面美观大方，使用jboot框架，并且开发速度快，部署简单，扩展性比较好。

### 1.3.5开发工具选择

使用具有强大插件支持的开源软件eclipse进行整个项目的代码开发，并且eclipse对于硬件和内存的要求也比较低，一般电脑也容易达到。并且也是相对比较流行的软件，不需要太多的时间学习软件的使用，就可以进行开发操作。

### 1.3.6论文整体框架

本课题设计使用了java ee技术框架，使用layui、jboot和数据的银行维修后台系统，实际意义在于：

(1)可以方便化、信息化的处理需求和满足需求。

(2)提高企业的服务效率

(3)降低了成本，节约了人力

(4)使得服务更加规范化，实现科学管理。

# 2 开发环境以及技术

本次课题设计通过前期选题，确定选择题目，并对其进行整体分析，以及需求分析，确定可行性，然后对各个模块进行分析，使用axure进行原型分析，并对其进行修改和优化，确保其完整性和健壮性能。其次，对整体数据库的分析，并对数据库进行整体设计。下来，对程序进行编码，完成各个模块的代码书写。最后，对其进行测试，并对bug进行修改，做最后的完善。

整体使用C/S架构，主要使用的技术有ajax、echart、html、css、js和layui书写整个前端页面，并进行数据展示，使用java、jboot框架，jboot快速上手的优势，快速搭建后台系统，并且自动生成服务层和数据库层的代码，简化书写，并且启动快，可以选择多种服务器作为web服务器。

使用shiro+jwt作为权限的处理，通过token传递认证信息，使用开源软件eclipse完成整个项目的编码，jboot作为控制层，使用jboot框架快速上手和生成服务端和数据库端的代码，通过maven构建工具进行整体项目构建，项目通过maven多模块的项目架构，使用rpc进行服务端和控制层的数据交互，它可以支持多种注册中心，比如zookeeper或者consol作为注册中心。使用强大的layui以及其对图表等优秀的支持作为前端页面数据展示。Mysql 5.7开源免费数据库作为数据存储。

## 2.1开发环境配置

本系统的开发环境如下：

(1)开发工具：eclipse、Hbuilder

（2）服务器undertow

（3）使用语言：java、js、css、html

（4）使用框架:jboot、layui

（5）数据库：mysql 5.7

（6）浏览器：Chrome

（7）开发平台：windows 7

（8）注册中心：consul

（9）缓存使用Redis.

本系统的服务器端：

（1）操作系统：Window 7

（2）Web服务器：undertow

（3）浏览器：Chrome

## 2.2技术概述

### 2.2.1 web概述

（1）web定义

Web即就是全球广域网，通常我们也称为万维网，一般地它是一个基于超文本和http的资源，通过浏览器进行访问，同时也支持安全的https，它是通过http+ssl进行数据加密，用在一些安全的操作，比如一行的业务等，如今人们安全意识的增强，http+ssl不仅限于安全比较高的场景下，无论是使用不安全的http请求还是使用相对比较安全的https请求，它们都是可以进行全球性的，并且可以动态交互的跨平台的分布式信息服务，超文本是一种用户接口方式，用于显示文本以及图片以及视频等相关的内容，只需要通过安装有浏览器的设备就可以进行资源访问和展示，超文本1.0仅仅支持文本，在http2.0之后，它已不再只是简单的文本，也包含多媒体的数据，并且也具有了动态交互的效果，用户可以进入输入数据，将数据在通过服务器交互然后动态的对页面进行渲染。

它具有图形化，他可以在一页上显示色彩丰富的图像和文本图像，并且可以显示视频、音乐、图像等。具有于平台无关性，无论在什么设备或者系统，只有有浏览器就可以通过互联网对资源进行访问，没有任何平台限制，并且操作简单。

(2)Web工作原理

一般情况下，服务分为服务端和客户端，服务端提供服务，数据的产生地，客户端是资源的获取者，它通过互联网技术将服务端的数据进行获取在浏览器上进行对超文本文件的展示。当我们在浏览器键入一个网址的是时候，首先会通过域名服务器进行域名的解析，找出服务器的IP地址，下来通过解析出的IP，对所请求资源所在的目标服务器进行发送get/post请求，在建立连接的过程中，客户端和服务端需要进行三次握手建立连接，在数据交互结束之后，又通过四次挥手客户端和服务端断开连接，为了效率，后来使用长连接，并不是结束之后，就立即断开，http请求可以请求服务器的图片、视频、文本等资源进行获取。通过浏览器进行展示。为了效率，如今不会再执行完就断开操作，它还会连接一段时间，如果再无连接的情况，执行挥手断开连接。

### 2.2.2 Java

Java一门面向对象语言，具有强大和简单易用的特性，并且易学，但是不同于c++语言，它抛弃了多继承、指针等复杂的概念，它具有一系列的优点，成为现在主流的开发语言，并在编程语言居于前列。它处理可以开发桌面应用程序、web应用程序还可以开发嵌入式，并且也在大数据方面有很大的作用。

(1)简单：

Java语言的学习和使用，相对于c和c++有很多的相似性，并且它不在使用一些难以理解的概念。Java语言使用引用代替难以学习的指针，并且也去掉了难以理解的虚继承。

(2)面向对象

Java语言提供类，接口、抽象等对象的特性，在java中没有多继承的概念，但是提高多实现的方式，并且不同于面向过程，使得问题更加简单，不在关系每一个步骤，将问题变为对象来进行处理。

(3)垃圾自动回收

Java将程序员从繁琐易出问题的垃圾回收解放出来，java后台程序会及时的释放不在被引用的资源，并将其分为多个年龄段，进行分年龄回收。、

(4)网络技能

Java可以连接网络，通过net包中的方法，用于处理http或者https，并且可以对get和post请求。通过URL就可以获取网络资源或者局域网络的资源，就像访问本地数据一样。

(5)健壮性

可以使用泛型等，将运行期错误变为编译器错误。

(6)可移植性

Java语言的数据类型大下是固定的，消除了数据存储空间大小不同的问题；在传输的过程中，也是使用固定格式传输，消除了传输过程中字节顺序的干扰。

(7)解释性

程序在编译时变为了字节码文件，它可以在任何有解释器上执行java字节码，开发过程更加的快捷。

(8)支持多线程

Java语言支持多线程，可以及时的响应代码程序，以及快速的处理请求，将数据快速的交给操作系统进行处理。

### 2.2.3 Layui

Layui是一个简单方便使用的前端UI框架，它对表格、echart、导入导出的强大支持，并且有丰富的案例，方便页面开发和使用，并且有丰富的官方文档和详细的在线案例，学习过程中也可以方便的通过社区交流和学习，并且提供了基本完整的后台摸版，并且轻量级，引入即可使用。并且开发速度快。它更多的面向后端开发，它具有体积轻盈，外在极简，适合快速入门开发，区别于vue的MVVM的UI框架，后端开发无需太多配置，适合快速开发。

### 2.2.4 Mysql

Mysql是一个分为免费的和收费的关系型数据库，是目前最流行、使用最多的的关系型数据库之一，在如今web服务器系统开发方面，占据着重要的地位，它有数据库和表的概念，方便了项目中数据使用的管理，并且具有比较高的灵活性和速度。它虽然并不是最完美的数据库，仍存在者不足之处，但是仍然是最受欢迎的，它开源免费，节约成本。并且它支持单点、辅助、集群的结构，使用于不同的环境和应用中。

优点：支持千万条数据的数据仓库；使用的平台比较广；开源免费；优越的性能

缺点：没有存储程序的功能；

具有以下特点：

简单易用:它遵循SQL规范，并且是一个高性能的数据库系统。

价格：对于大多数使用者来说是免费的。

小巧：安装是仅仅需要小的存储空间

支持查询语句：它支持SQL99语法，并且是所以数据都选用的语言

性能：可以同时处理多个连接到服务器，无用户个数限制，多个机器可以连接一个，并且可以通过客户端、程序、第三方工具等多种形式连接。

它支持多种存储引擎，并且支持插入式的存储引擎，支持的引擎有：MyISAM、InnoDB、BDB、Memory、Merge、Archive,其中MyISAM是默认的数据库引擎。

MySql它可以通过不同的语言与数据进行交互，几乎支持所有的语言类型，还有系统管理和控制工具，实现数据备份、恢复、mysql复制、集群等。还具有连接池，可以将用户的用户名、密码、效验等放入缓存中。当用户端将SQL语句发生给数据库服务器的时候，如果是查询，那么先进行查缓存，如果有数据，那么就将缓存中的数据返回，如果缓存中没有，然后通过解析器将sql语句解析生解析树，接着对其进行预处理，并对查询操作进行优化，执行相应的查询计划，最后执行速索，引擎，将数据返回，并将数据写入缓存中。

### 2.2.5 Jboot

Jboot是一个微服务框架，它和spring boot相比更加容易上手。它对服务器开发进行了强大封装，非常方便使用，同时支持DB+Record模式，灵活方便的操作数据库。还具有更加强建的AOP。支持Dubbo、Motan等方式的RPC方案。MQ的消息队列，还有强大的缓存，分布式等该性能缓存框架。

### 2.2.6 Ajax介绍

Ajax不是对整个页面进行刷洗，仅仅对用户所关心的部分，进行与服务器交互，可以节约资源，加快访问效果。使用过程中，实现代码在js片段中，可以将数据从服务器获取到，然后对服务器的数据进行使用js的循环等方式，进行处理，依次来渲染页面。Ajax它也不是新技术，仅仅是一个成熟技术的整合，是一个轻量级的解决方案，使用HTML或者XHTML，通过js脚本语言，使用xml进行与其他的应用程序进行通信，并且是居于其它成熟的技术。

### 2.2.7 shiro介绍

Shiro作为一个权限管理框架，它可以完成涉及用户参与的权限控制，权限属于系统的安全部分，对用户访问的资源进行流程控制。安装一定的安全规则和策略，一般的权限管理包括身份认证和授两个部分，对于需要访问的资源首先进行身份的认证，身份认证之后的用户才可以对资源进行访问。简单说就是对来访者的合法性进行判断。

### 2.2.8 CS模式

CS模式是由客户端和服务端组成，使用http请求进行数据访问的，作为客户端，只需要一个装有浏览器的设备据可以进行对资源的访问，不同于BS模式，需要同时开发客户端和服务器端,并且升级等操作，大多仅仅只需要对服务器进行就可以了，无需关心客户端，维护方便。所以它具有分布性强，可以充分地展现PC的处理能力响应比较快，开发简单，页面比较漂亮美观，可以使用多种方式进行，共享性强、成本低等特点，但是用户不确定造成数据安全问题、服务器要求比较高。

## 2.3小结

本章主要锁使用的语言进行了简单的介绍，本次毕业设计使用的开发语言java，它的特性以及优点进行了介绍。下来又介绍了mysql数据库的引擎，mysql包括哪些部分，以及在查询操作中，做的工作和流程。

# 3系统设计

## 3.1 系统可行性分析

第二章对所用技术进行介绍，那么在本系统中的语言选用的可行性进行验证，并且在如今计算机技术的发展中，各种系统的使用，为人们的生活带来方便和快捷。

### 3.1.1技术可行性

基于浏览器，通过http请求的维修后台系统的设计，关键的技术点在于如何的获取数据，如何将数据存入数据库，实现一个完整的流程操作，并且是准确无误的进行流程控制。采样layui作为前端页面展示，首先它美观大方，可以对数据进行展示，并且支出表格、直方图、饼状图等，后台使用jboot进行开发，可以对数据库读取的各种操作，并且效率高，数据库使用mysql数据存储，它在如今的应用场景中使用很广泛，并且也有许多的成功案例，并且免费开源，也是一个成熟的技术，所以整体的技术选型是可行的。

### 1.1硬件配置可行性

系统使用的eclipse，占用内存小，启动快，无论是在安卓还是web开发，都可以快速的编译运行，一般电脑都可以达到要求。使用Redis作为数据缓存，它有liunx版和电脑版，并且软件比较的小。使用consul作为注册中心，将服务器注册到consul，并且在 windows上，它比较小，并且启动方便。数据库也很小巧，也不会有其它问题。

### 1.2操作可行性

通过浏览器进行系统的访问，只需要局网就可以进行consul的正确启动，完成系统Redis的连接，并且可以完成基本的安装、调试、运行。并且使用和操作方便，基本不需要任何培训，因此在操作可行性上，开发方案可行。

### 1.3系统需求分析

无论做什么系统，都需要进行系统分析，在系统设计中是一个很重要的部分，通过分析将软件的总体性能和功能描述为具体的规范文档。依次来进行系统开发。

## 3.2系统架构分析

一般传统的单体系统会三层架构，通常包括了页面展示的表示层、数据业务逻辑处理的业务层以及数据库操作的数据访问层，它们之间相互协调，是软件开发中经典的开发模式之一。但是这样虽说进行了分层，通常也就是逻辑上对系统进行的分层，并没有实现在物理上进行分层，它的编译、打包、部署等操作，都是在一个进程的应用程序中的，我们也将其称为单体的架构应用。但是随着业务逻辑的加大，以及系统功能的完善，使得单体项目越发变得笨重和庞大，最明显的事一个启动都需要花费好长的时间，并且也回造成，维护性、扩展性和灵活性的降低，并且使得修改成本的加大，维护困难。项目的分层发展过程，开始的一层，比如在jsp中操作数据库。后来将数据业务层和逻辑层分开， 这个时候数据的业务层和数据库层仍在一起，只不过将逻辑层分开了，这就是所谓的二层结构。最终的三层架构是将它们三者都进行分开。那么三层的优势在哪里呢？一方面可以解决调用复杂、代码职责不明确，代码升级和替换不方便，不利于维护。有利于整个项目的应用程序的架构标准。同时也解决了开发过程中根据技术调配人员，各司其职，更可以将个人专业技能进行展现，三层架构成为了最经典的开发模式。但是它仍是一个单体。我们将逻辑、业务、数据库层分开，但是无论如何最终将器也打包到一个应用程序中去了。最终会在一个机器的一个进程中运行。都是一个包、一个部署，一个进程中运行的程序，我们称为单体架构。通常是一个jar或者是一个war。

### 3.2.1它的优势：

（1）易于开发

单体项目，开发的相对简单和方便，一般从数据库到服务层、逻辑层，最终去到页面进行页面的操作。开发易于理解。实现方便，也比较方便，可以使用Eclipse、IDEA等进行开发，并且可以有效加载整个应用程序的依赖，方便项目开发人员对系统进行运行、调试等操作。

（2）易于测试

单体容易测试，它运行在一个进程中、启动集成环境或者将其部署到某个环境就可以运行，只需要将进程启动。

（3）易于部署

一般只需要将war或者是jar加载启动就可以了。只需要正确打包，将一份放入到服务器就可以了。

（3）易于水平伸缩

因为它将一个功能打成一个jar文件，并且只在一个进程中，可以将其进行克隆，创建服务器节点，运行在一个新的运行环境中，同时也可以使用负载均衡器来采用分发策略，有效的将请求分发到不同的站点上。

（4）挑战：

单体已无法满足当前互联网的飞速发展，随着业务不断的扩张，需求不断的加多，单体如何为用户提供可靠的服务，如果处理在高峰时用户的流量大使得系统响应变慢的问题。并且维护和测试的成本加大、构建和维护的成本显著增加。

本次课题设计是开发银行监控维修的后台系统，它包含了员工管理、机构管理、订单管理、仓库管理等多个模块，它们都具有增删改查的方法，对于仓库还需要有导出的功能，并且在整个系统中还需要权限进行管理，通过权限管理，实现不同用户实现不同的操作。它们需要与数据库进行交互，将数据保存到数据库。整个系统的架构，如下图3-1，浏览器的http请求作为访问整个系统的一个入口，他通过互联网与系统所在的服务器进行通信。在项目启动的时候，首先会将服务注册到consul中，以供给下来访问使用，将服务层等操作也放入到consul中，请求和处理通过服务的方式进行交互。

RPC远程过程调用可以看做为通过客户端/服务端交互的模式，客户端发起请求，服务端进行响应。它们的交互如下图，交互流程为：

（1）客户端调用本地的一个代理存根，发送请求到达服务器端，等待应答信息。

（2）在服务器端，服务器处于睡眠状态，客户端请求将其唤醒。

（3）服务代理获得请求参数后，交由服务端的服务代码对其进行处理。

（4）应用程序处理结束之后，服务器进行应答处理，等待下一次的请求。

（5）客户端代理存根接收应答信息，交给客户端的调用代码进行处理。

对于RPC中的留存，留存是在客户端和服务端都需要进行留存的，在服务端和客户端之间提供了一种客户端请求，服务端响应的交互方式，如图3-1。

调用

调用代码

调用代码

调用

返回

返回

调用

返回

代理存根

代理存根

调用

调用

返回

返回

调用

返回

RPC库

RPC库

图 3-1 RPC调用逻辑图

远程方法调用，它是一种面向对象实现，远程调用的弊端，使用代理存根的方式，屏蔽了通信双方的底层调用细节，在客户端调用就像本地调用一样，不必强调是客户端本地，还是远程调用，使得分布式变得更加简单和方便。但是带来好处的同时，也带来了弊端和问题，首先耦合度比较高，大多数的远程实现是依赖于语言或者平台的，因此也就限制了语言的使用。第二点就是灵活性变差，依赖于编程语言，传输一般是二进制数据，所以使用的时候需要序列化和反序列化，因为客户端和服务端都有代理存根，那么当一方的数据发生改变的时候，就需要将相应端的进行修改，来保持代理存根一致。

### 3.2.2系统整体功能结构

银行维修后台子系统

认证成功之后，根据权限访问资源

权限模块

统计模块

订单模块

基础设置模块

仓库管理模块

机构管理模块

维修工管理模块

图3-2 系统整体功能结构

## 3.3系统逻辑功能分析

无论是管理员还是普通用户，当他们完成认证之后，就会根据权限去访问已经授权的资源，在这个里面，不同的用户登录，系统分配不同的角色，不同的角色在用户认证成功之后就会将权限缓存缓存在Redis中，方便之后的使用，为了加快系统的响应效率，在前端页面显示的时候，通过jboot提供的权限方案对资源进行控制，达到不同的用户展示不同的按钮，不同的用户执行不同的操作。权限数据也缓存在Redis中。

维修工管理模块:里面有维修工信息，并且可以对信息进行删改查操作，以及详情信息，可以方便对维修工信息的操作。

机构管理模块：机构信息的展示，增删改操作，并且可以对网点进行操作操作，查看详情等操作。

订单页面可以通过此部分，进行下单，以及订单的处理，以及服务的订单附件，这一部分是系统涉及核心最多的部分。

权限模块：这一部分包括资源的管理、角色的分配、用户管理，以及授权操作，这一部分更多涉及的是系统的安全，本系统使用shiro作为整个系统权限的控制部分，这个框架可以完成基本的权限要求。

### 3.3.1系统逻辑结构

控制层：管理员和用户都是通过浏览器进行登录的，然后通过互联网将请求发送到控制层，这个层中有着整合系统的控制路径根据数据库查出的结果，然后进行身份验证，确定角色，不同的角色在去加载数据库中不同角色不同的权限信息，不同的权限将会看到不同的操作和页面布局。

服务层：服务层为控制层服务，作为数据库的控制层，可以访问数据库，从数据库中获取数据，并传递给控制层，它作为一个控制层和数据 库的桥梁，服务层也需要注入到consul中，控制层可以从consul获取到服务，并进行调用，和本地程序一样，这样也容易解耦，服务层可以方便的进行控制，并且可以重复使用服务。

管理员

机构用户

浏览器（认证页面）

Consul

（注册服务）

Internet

功能层

服务层

Redis

(缓存)

数据库

图3-2系统逻辑图

数据库是整合系统数据存储和访问的地方，可以是mysql、redis等其他的数据库是整个系统中最重要的部分，它在web项目中可以位于任何地方，可以是本台机器，也可以是远程机器。它可以接受服务器的控制，执行持久化或者是查询操作。

在系统中使用shiro做为权限管理控制，授权流程如图3-2，shiro权限管理，它通过主题subject，对shiro提供的api进行操作，他可以是人，也可以是程序，这里是人输入的用户信息，通过SecurityManager进行管理，SecurityManager是验证过程中真正的验证执行者，所有的安全验证的处理都是通过subject与SecurityManager管理器进行交互的。

访问系统资源

分配权限

身份认证

认证是否通过？

NO继续认证

YES

权限控制

是否拥有权限

访问拒绝

YES

继续访问

图 3-3 shiro身份验证图

当用户需要登录的时候，用户通过系统提供的用户登录界面来输入验证信息实现系统身份认证，当用户输入用户名和密码验证用户身份认证通过后，在进行查询数据库判断登录提供的认证信息进行权限的分配，将权限放入到Redis缓存中，以便下次快速响应客户端的请求，如果没有通过认证，那么就再次的到达登录页面，重新输入认证信息。

（1）查询逻辑

整个系统中有多个查询，查询不同的内容，但是大概逻辑基本相同，通过客户端提交查询数据，在服务端通过控制层处理，通过rpc向服务层发起请求，查询数据库，将数据最终显示在页面上。查询的流程图如图：

查询开始

查询输入

控制层调用数据库

是否有结果

提示：无数据

无

渲染

退出

图 3-4 查询逻辑图

（2）修改逻辑

修改逻辑执行的时候，需要首先查询数据，是否存在这个Id所存在的信息，如果存在那么就执行修改操作，如果不存，就提示次信息不存在给提提示。然后返回到之前操作的界面。

开始修改

填写表单

返回操作前的页面

提交数据

ID是否存在

无

提示用户不存在

有

保存数据到数据库

结束

图 3-5 修改逻辑图

（3）用户登录逻辑

用户在输入用户名和密码等验证之后，会首先查询系统的数据库，在根据用户名确定是什么样的用户角色，然后在根据不用的身份加载出不同的用户角色，根据角色得到最终的权限和可以访问的资源，并将其放入到Redis中，如果是管理员登录成功，那么就进入到管理员界面，如果是管理员登入系统，登录之后系统就会自动的展示出管理员的界面。如果输入的用户名和密码不匹配，系统会出现弹框提示用户验证失败，然后让用户进行重新输入验证信息。

登录密码及用户类型验证

请求用户名及密码

用户存在且密码正确

用户不存在或密码错误

是否系统管理员?

进入后台管理模块

其他用户管理模块

操作结束?

操作结束?

退出系统

开始

用户授权

是

是

否

否

是

是

是

图3-6 登入系统程序流程图

### 3.3.1系统的功能要求

在整个系统中，不仅仅有管理员，还有其他用户，他们以不同的身份会出现不同的功能模块，比如管理员具有至高无上的权利，它可以做一些普通用户无法完成的功能，控制权限分配，这就是普通用户没有的，和无法实现的部分。只有通过合法的方式进入系统才可以在系统中查看到符合自身权限的部分，对规定的资源进行访问。

管理员进入系统，可以对机构、维修工、订单、权限等都进行操作。

维保机构人员进入系统，可以查看自己的订单和维修量，可以对自己的网点进行修改的查看。

整个系统满足以下几方面的要求

（1）满足不同用户角色的应用需求

根据需要，灵活的设置功能，从维修工到维保机构、仓库管理和订单处理等以及进行整个系统的页面统计和最终结果的展示，整个过程的参与者和执行者有管理员，维保机构人员、以及维修工，他们用户根据不同的角色进行划分，形成三种角色。它们分别是：

管理员：可以对系统的一些高级功能或者权限等资源管理或者是进行变更，以及对它们信息进行操作与维护。

机构人员：可以对网点进行操作，对自己机构进查看的管理，同时可以发布维修订单，对订单的结果进行确认，完成整个维修订单的结束。

维修工：可以查看订单，并可以对订单进行接单操作，执行维修操作。

（2）满足多种统计的需求

统计对于系统来说比较重要，系统我们需要实时的查看系统的状态，以及数据的变动状态，满足各种查询和展示需求，需要支持饼状图和条形图，使得结果易读取，易统计，易操作。

（3）满足多种功能需求

系统处理对于维修工和机构的管理，还对供货商、仓库出入库，以及导入导出进行了设计，可以查看仓库物品的数量，供货商信息，以及其他具体信息。管理员可以对维修工的状态进行查询，并且还可以指定维修工进行维修单的处理。

## 3.4系统性能分析

使用jboot开发，通过consul注册服务，客户端取服务的方式，在运行是，服务端将服务暴露出去为客户福安使用，运行是，每个微服务可以单独的进行运行，有可能是在云虚拟机上，也有可能是运行在一个机器的docker容器上，并且数据存储，每个服务可以有单独的数据库，这个数据库可以是任意的数据库mysql、MongoDB等，一般地，会将一个单进程运行的庞大的单体项目通过业务逻辑等方式分解为多个单独的微服务模块，解决了庞大单体的开发复杂性问题，并且使用RPC作为消息沟通的载体，每个单体的服务更加的易开发和方便维护。开发可以由不同的团队进行开发，可以使用多种语言，相比单体，它的灵活性更高，可以单独部署。

## 3.5服务器安全

面临各种的网络环境，网络多种多样，一般网络安全分为两个维度，一个是网络结构，比如通过防火墙、内外网分离等方式，还有一种是通过软件上实现其安全性。常见的程序级别的安全手段有信息加密、认证、授权和安全协议等范式，信息加密技术通常最常见的方式包括一下三种：对称加密、非对称加密和散列加密。单向散列加密的方式通常为使用不同长度的明文最终为长度固定的一个密文，转换过程不可以逆向进行。对称加密使用是同一个秘钥，非对称加密是的是公钥和私钥。整体的验证授权完成的就是“你是谁”，“你可以干啥的问题”，前面已经阐述了本系统使用shiro进行权限验证，但是到现在为止，我们的整个数据是通过ajax得到，并且各个接口并不是存在本地，而是通过http进行接入的。那么需要明确的两个问题是：

（1）不同服务之间如何完成验证

（2）各个服务之间如何实现授权

### 3.5.1安全设计方案

http是无状态的，那么每次访问服务器的时候都是一个新请求，对于这种情况最常见的方式就是使用session，因为它可以记住状态，第一次访问的时候就会给一个sessionID，这个办法很方便的将进行了http请求无状态的问题，但是有一个问题，就是服务器要保存这有个session的数据和认证的信息，也就是说用户状态的维护，我们将它们放在了服务端，无端的增加了服务端的压力。

为了减少服务端的压力，可以使用在客户端的方式进行处理的方式，这也就是使用客户端token的方式进行权限验证的方式，它的好处就是不需要服务端进行状态的维护，在token中就保存了相关的信息，并且token也具有可扩展性。

近年来，随着Restful API和微服务的使用和兴起，token的认证已经成为项目进行权限验证和交互的一个重要手段了。此时应该知道，浏览发送一个请求是，带到服务端的sessionId仅仅使用一个key，但是token传递的不单单是一个key，它保存的是验证通过用户的身份验证信息，通过token完成基本的工作机制。它的实现步骤如下：

用户使用包含用户名和密码的身份凭证从客户端发送请求。

在服务端第一次接收到浏览器的请求之后，服务器就会生成一个有效的token字符串，将其放入到响应头中，并返回给客户端。

再次发送请求的时候，就需要客户端通过浏览器使用带有token参数的请求，然后获取服务端的资源返回给客户端进行展示。

授权中心

身份凭证

身份凭证

客户端应用

Token

Token

身份凭证

用户

授权验证

认证过滤器

授权过滤器

授权中心

认证过滤器

认证过滤器

资源

认证过滤器

API控制器

图3-7 token授权流程图

本次课题设计使用jwt，作为身份验证，在服务端和客户端进行交互过程中，传递信息。该信息可以被验证和信任，它也支持跨域，jwt和session的不同点在于，session仅仅拿着数据的一个标识，仅仅是一个字符串，标识我是谁，jwt将用户的信息编码之后存储在请求头中，这样有个好处就是减轻了服务器的压力。

# 4 主要程序设计说明

项目代码完成之后，通过部署，可以查看到项目运行的基本效果，下面是项目中的部分运行图。

## 4.1登录页面



图 4-1 用户登录

登录页面是一个很重要的页面，它是外部人员进入进行，对系统查看和操作的一个通道，用户输入用户名密码之后，验证通过之后就可以进入到系统的内部，在书写的时候，登录页面的布局相对来说比较的简单、美观以及布局可以给人一个舒畅的感觉。无论是管理员或者是系统的其他用户，都是通过这个页面进行身份验证。登录页面的布局如上图4-1。

登录页面需要向服务器传递用户名、密码以及验证码。只有通过系统验证的用户才有权限进入到页面的主页，如果验证失败，页面会对用户进行一定的提示，验证让使用者再次的进行重新验证。

## 4.2主页

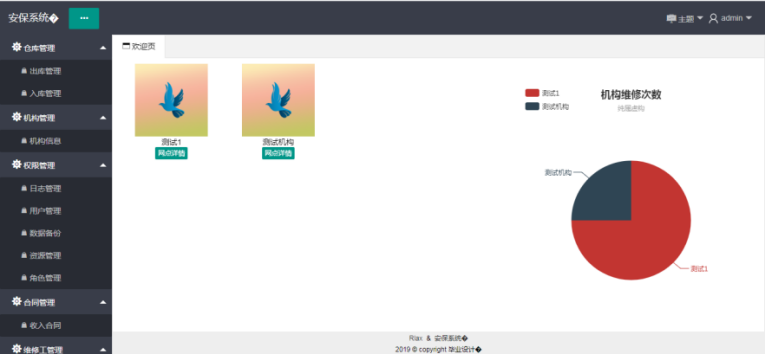


图 4-2 首页展示图

主页是用户通过登录权限进入到系统中看到的第一个系统页面，它的布局也很简洁，左边作为整个系统的导航菜单栏，右边的主页面主要分为两个部分，一部分用于显示维保机构的列表，另一个部分用于显示出每个维保机构的维修次数，使用饼状图显示，下面的部分显示的是系统信息。在每个列表的下方可以点击查看网点的详情如上图4-2。网点详情里面通过表格和柱状图展示该机构网点的维修次数，如图4-3和图4-4。

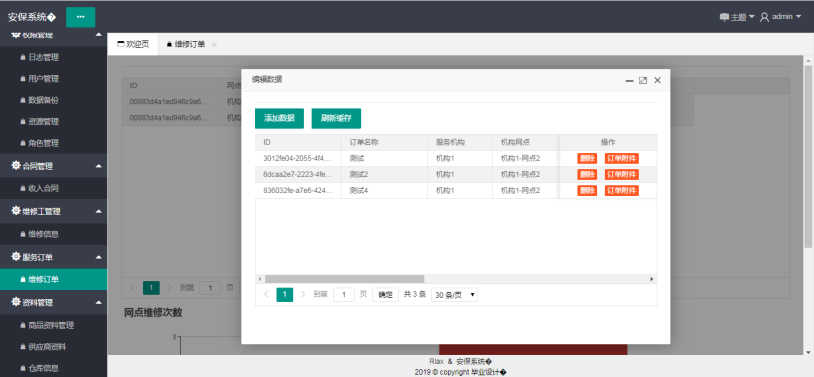


图 4-3 网点详情图



图 4-4 网点维修统计图

## 4.3维修工管理

这一部分主要是对维保机构中的维修工进行管理，其它功能模块的页面布局和这个类似，使用一个表格用来显示维修的基本信息之外，在列表的后面用来显示维修工的基本操作，通常为编辑、删除、修改操作，因为这一部分的操作是在单一的数据上操作，实现方式有一下几种，一种是单选框，一种是在数据上增加一个数据的点击事件或者直接使用多选框的方式，选中多个然后进行数据的操作。上面的部分一个是进行数据的增加操作，还有一个是刷新整个页面的数据。还有速索操作，当用户输入查询条件的时候，数据通过ajax的方式进行与数据库的通信，然后将数据得到显示在表格上，由于使用的layui显示数据，所以这一部分是非常简单的方便的，只需要将服务器数据通过异步的方式得到，就可以进行渲染，如图4-5。

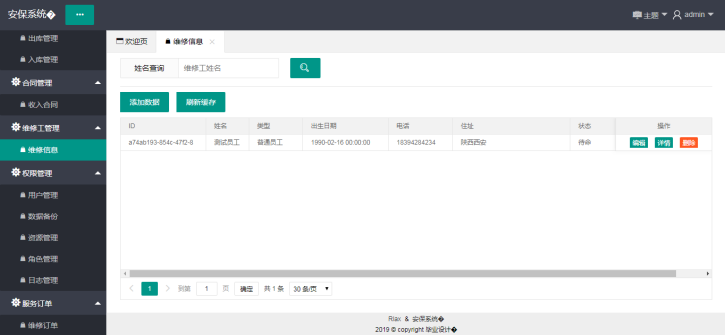


图 4-5 维修工信息列表

维修工信息进行增加时，鼠标点击上方的增加数据按钮，进入增加界面，如下图4-6。在信息输入维修工的信息结束之后，鼠标点击下面的添加按钮，然后系统会自动跳转到列表页面。

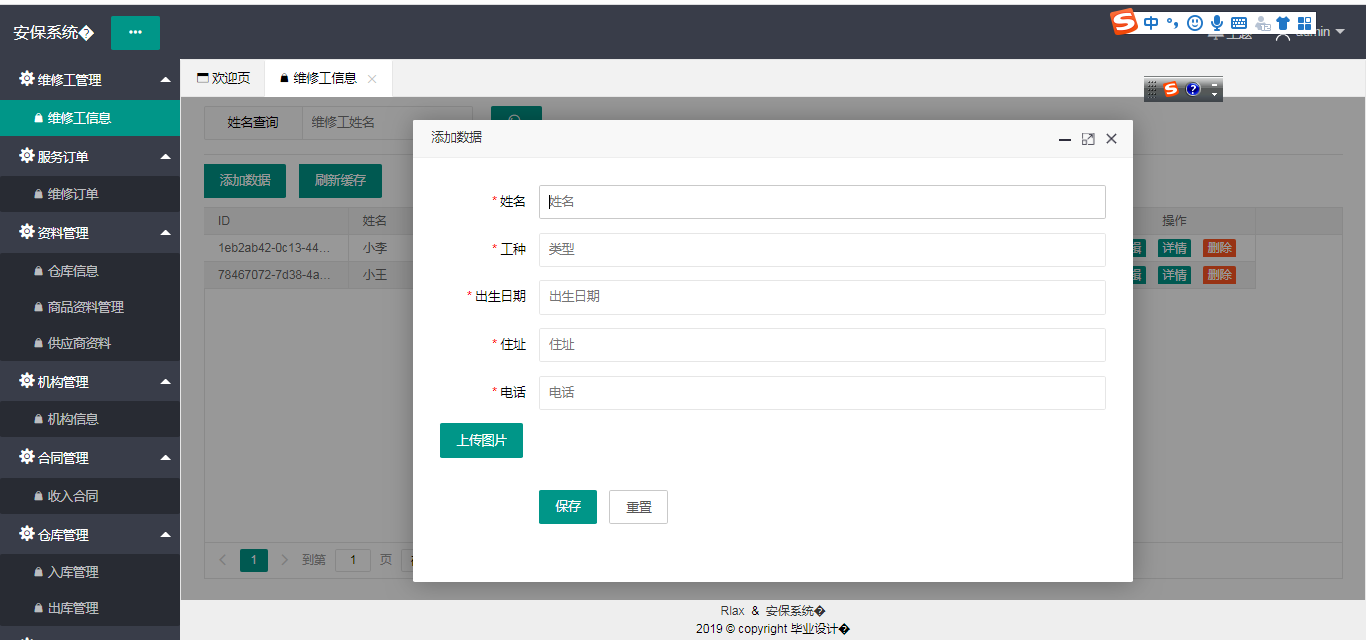


图 4-6 添加维修工

维修工信息进行修改维修工信息时，鼠标点击列表信息后面的编辑按钮，进入编辑界面，如下图4-7，编辑结束，会跳转到列表页面。

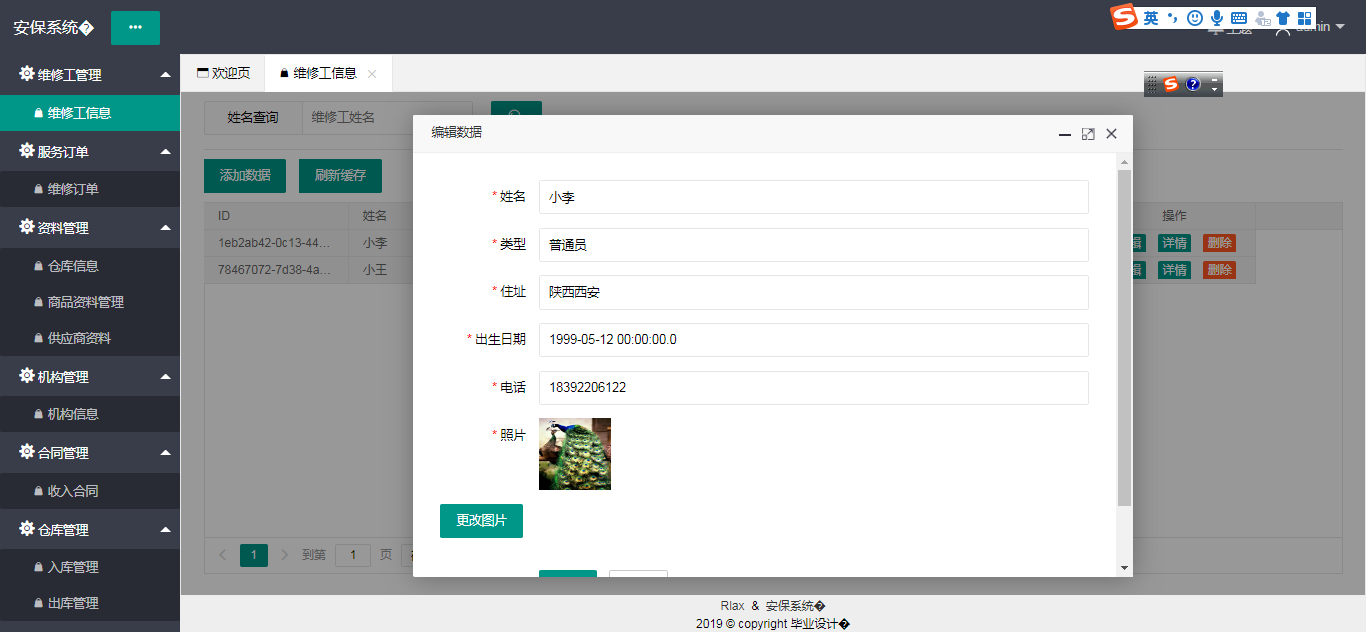


图 4-7 编辑数据

对维修工信息详情进行查看，鼠标点击列表信息后面的详情按钮，进入详情界面，如下图4-8。



图4-8 查看详情

## 4.4维保机构

维保机构列表，对于维保机构，具体来说保扩两个部分，一部分是维保机构，一部分是维保机构下的网点，这部分我的处理方式是首先在一部分的显示中，我仅仅显示维保机构和网点的个数，并不会进行里面维保网点的显示，当用户需要查看网点的时候，就需要通过点击网点详情的方式进行获取网点信息，在进行显示，机构有每个机构的logo，所以这一部分也是可以进行图片上传的，并且可以查看每个机构的信息和网点的信息，以及签约的信息。

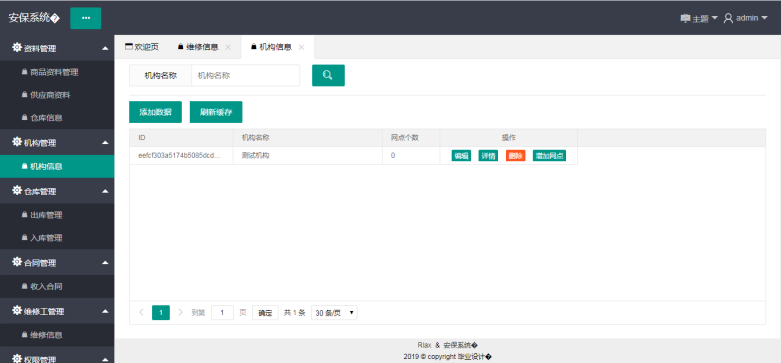


图 4-9 维保机构列表

也可以对维保机构进行网点详情进行展示，鼠标点击列表信息后面的详情按钮，如下图4-10。

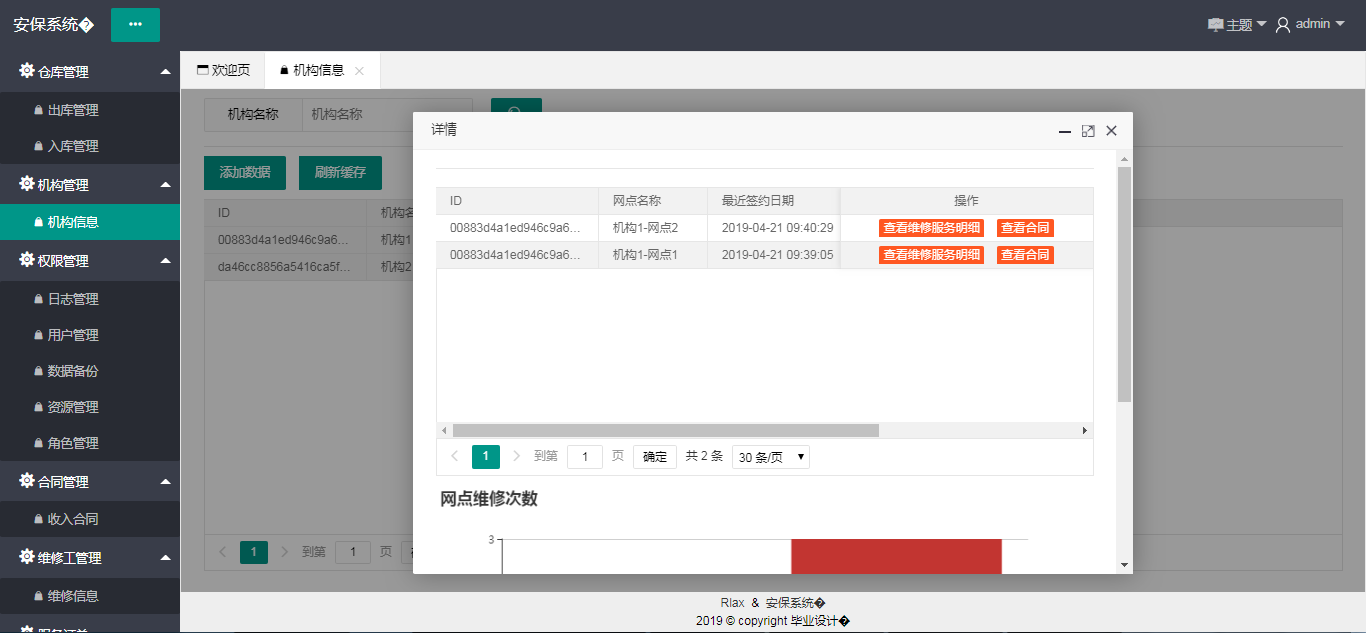


图 4-10 维保机构网点详情

对需要对维保机构进行编辑的时候，鼠标点击列表信息后面的编辑数据按钮，如下图4-11。

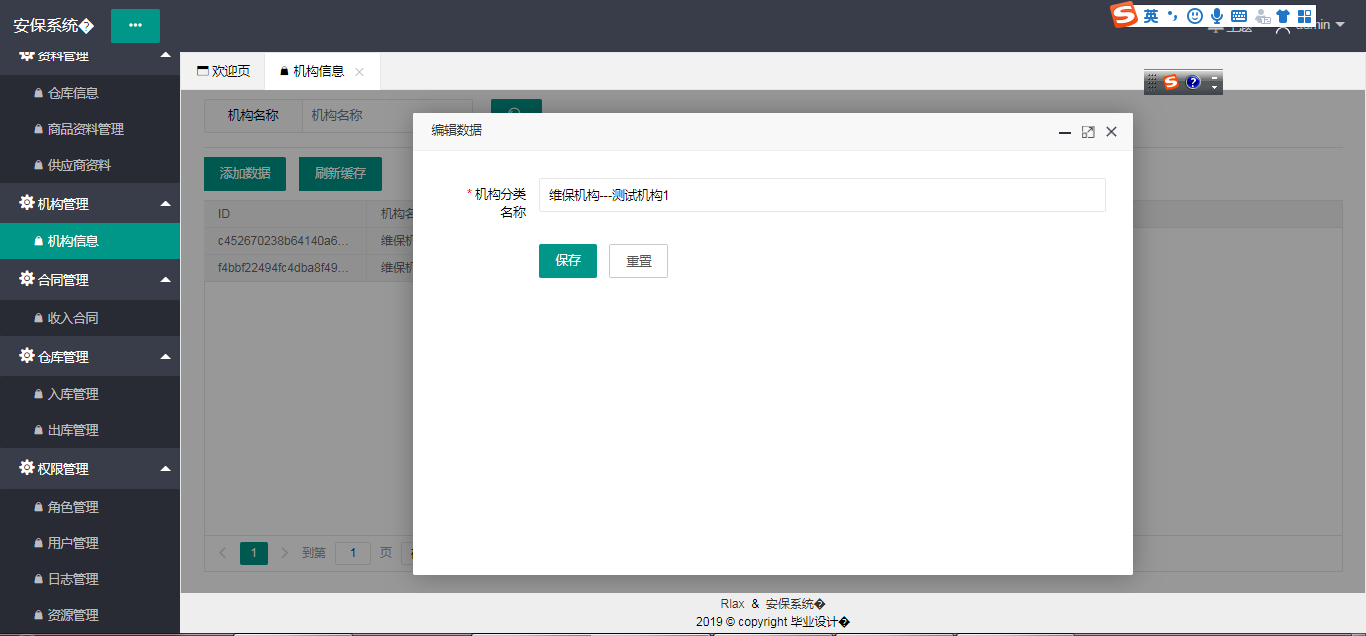


图 4-11 编辑机构名称

也可以对维保机构进行删除，鼠标点击列表信息后面的删除按钮，会弹出删除的确认按钮。用户可以根据自己的需要选择删除或者是不删除。如下图4-12。

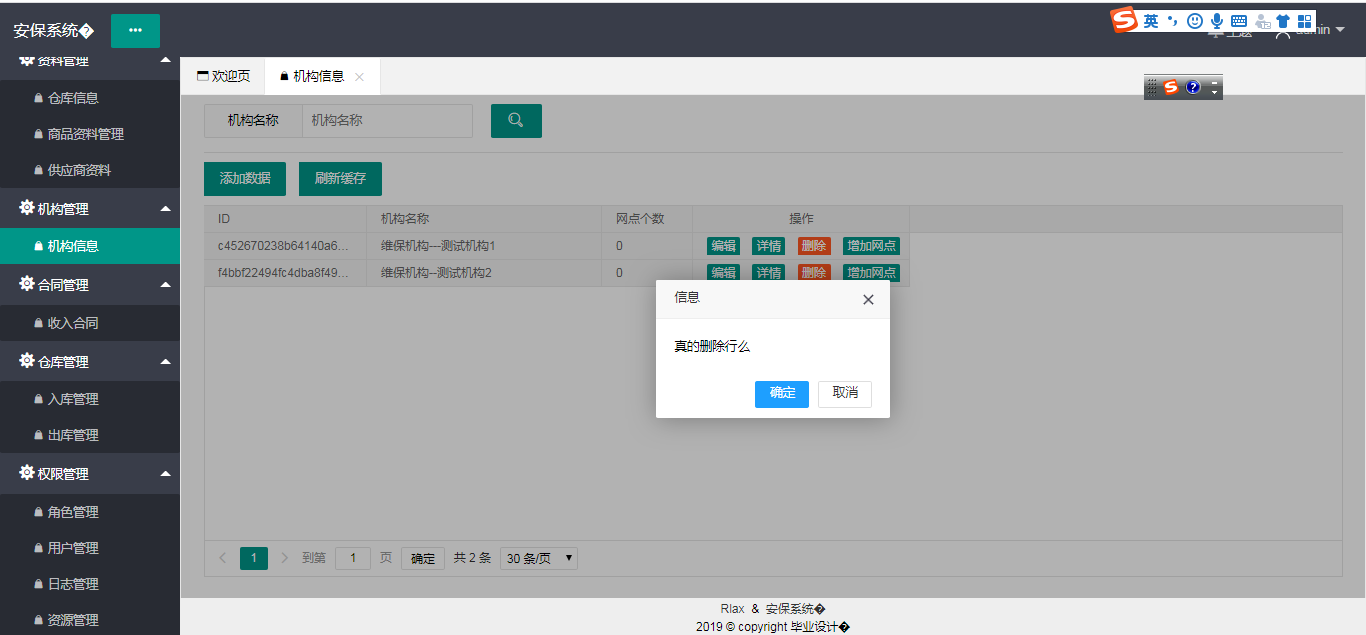


图 4-12 删除维保机构

上面的部分主要是机构的基本操作，下来介绍第二部分，也就是对维保机构的网点进行增加，鼠标点击列表信息后面的增加网点按钮，如下图4-13。

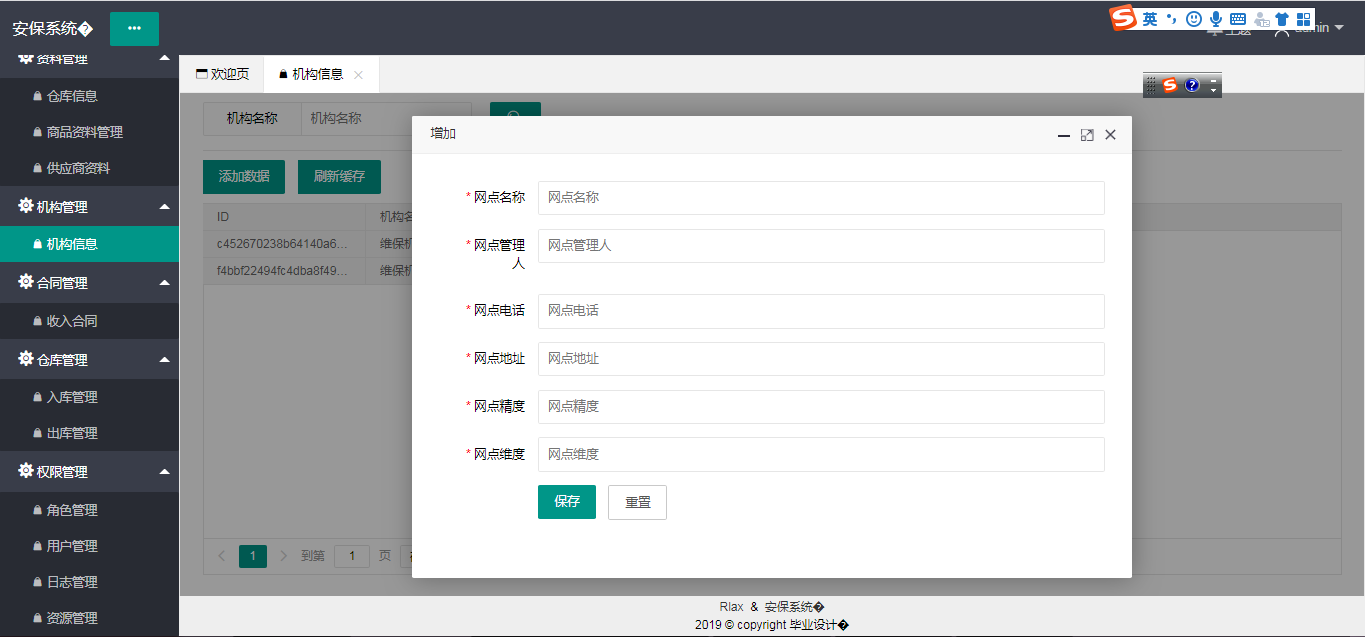


图 4-13 增加机构网点

## 4.5出入库模块

### 4.5.1供应商模块

这一部分主要是为仓库管理那一块服务的，维保机构中的物料处理是非常重要的，供应商管理在进行物料操作的时候，方便用户查看物料供应的供应商信息。用户也可以根据需要选择不同供应商提供的同种商品。这里也只要是增删改查为主。下图为供应商的资料列表，后面通过按钮进行操作，如图4-14。

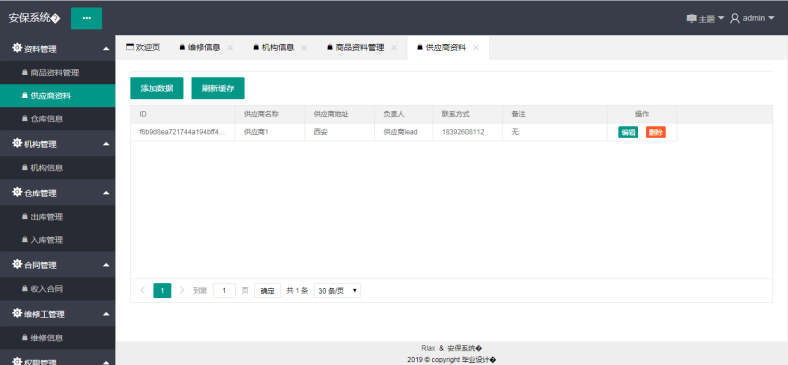


图 4-14 供应商资料列表

拥有权限的用户可以对供应商进行增加，鼠标点击列表信息后面的增加按钮，如下图4-15。



图4-15 供应商列表增加

供应商信息进行修改维修工信息时，鼠标点击列表信息后面的编辑按钮，进入编辑界面，如下图4-16。

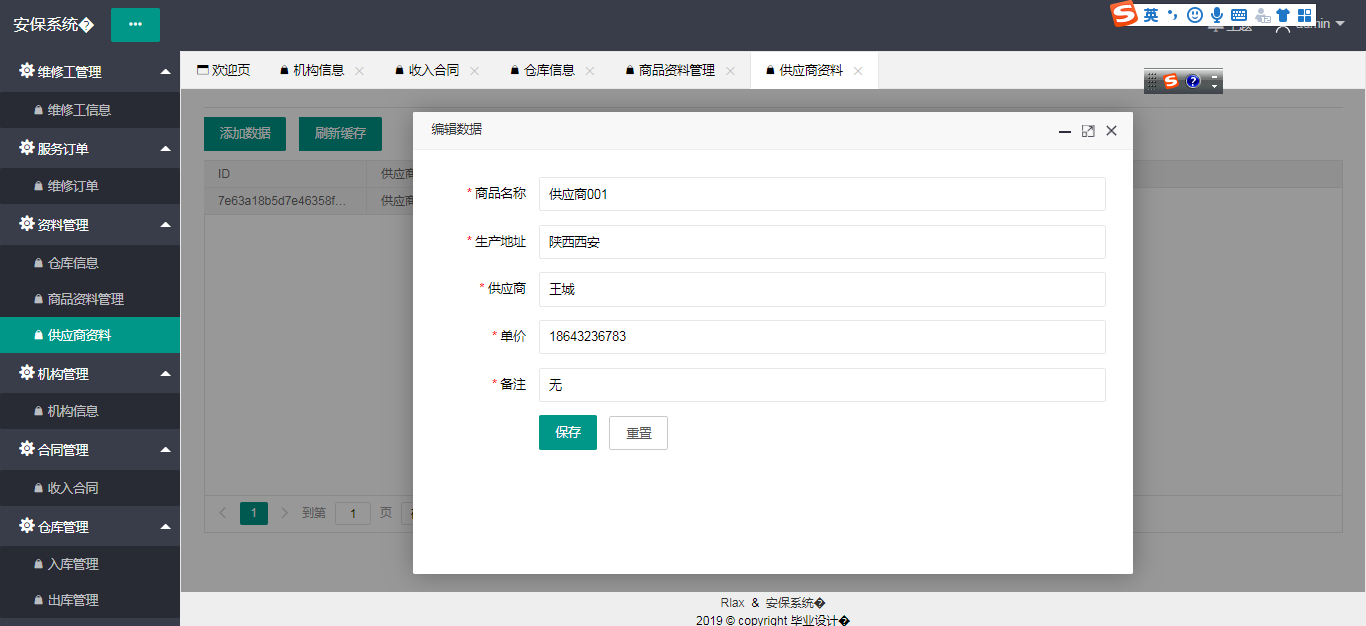


图4-16 编辑数据

也可以对供应商信息进行删除，鼠标点击列表信息后面的删除按钮，如下图4-17。

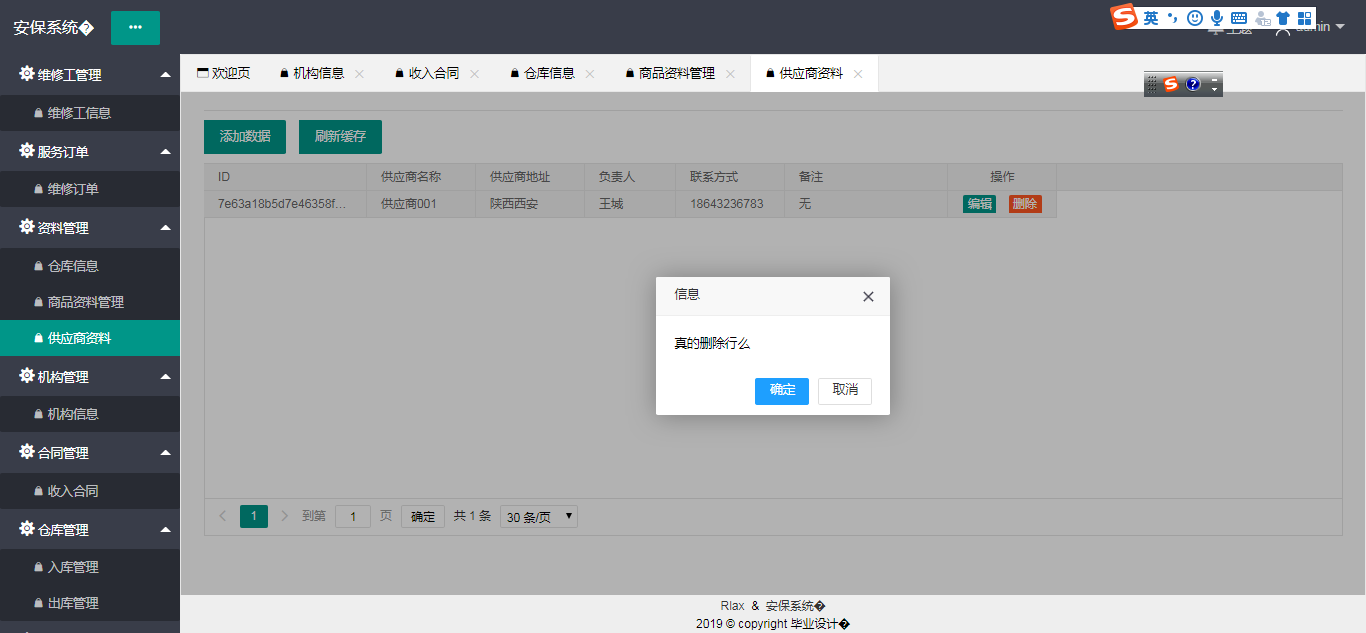


图 4-17 删除供应商信息

### 4.5.2商品管理模块

通过table表格进行商品信息的罗列，也可以通过后面的操作按钮，进行其他操作的处理，比如查看供应商信息，对于系统的管理员或者是其他用户具有操作供应商信息权限的来说，可以对于供应商信息进行供应商信息维护的操作，没有权限的普通用户只能进行查看，不可以进行其他操作，如图4-18。

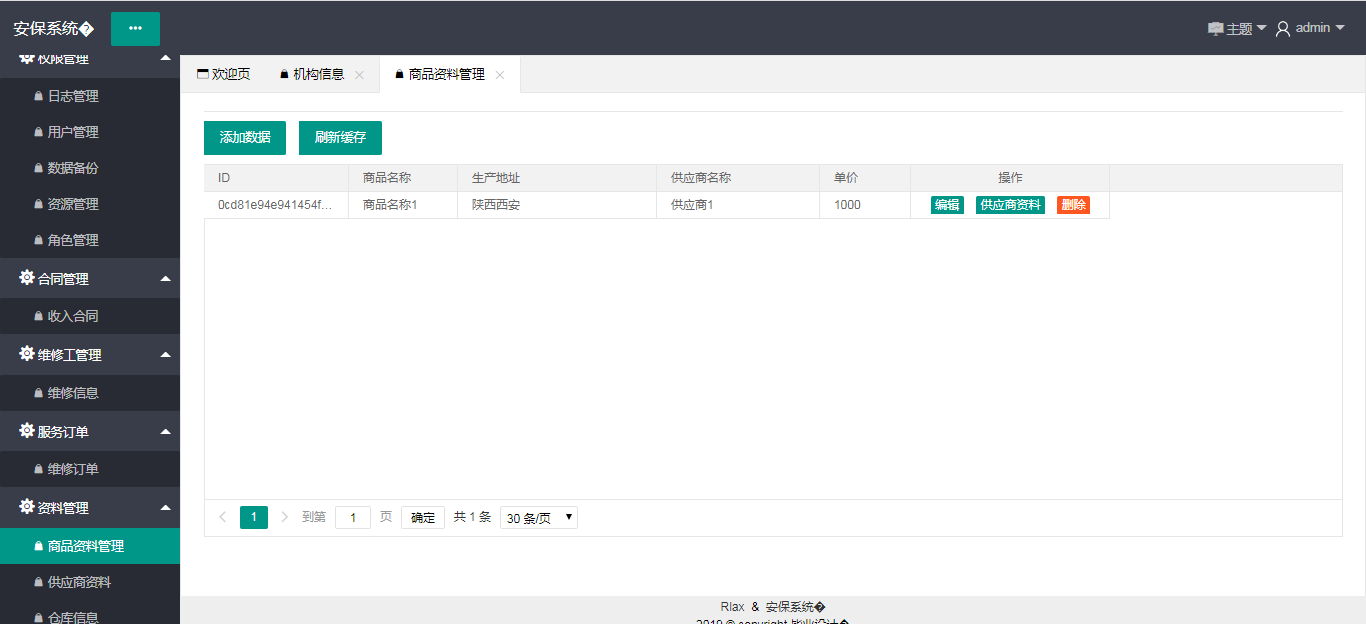


图 4-18 商品列表

拥有权限的用户可以对商品信息进行增加，鼠标点击列表信息后面的增加按钮，如下图4-19。

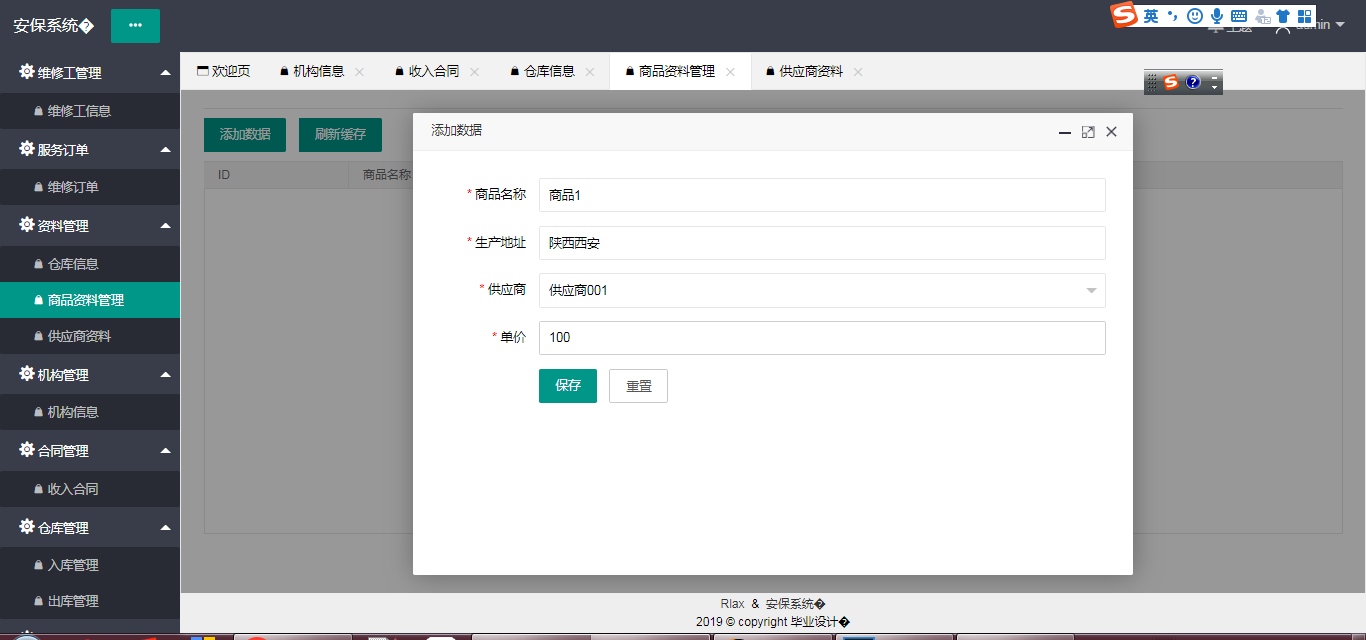


图 4-19 添加数据

用户对商品信息进行修改操作时，鼠标点击表格后面的编辑按钮，系统将会出现编辑界面，如下图4-20。

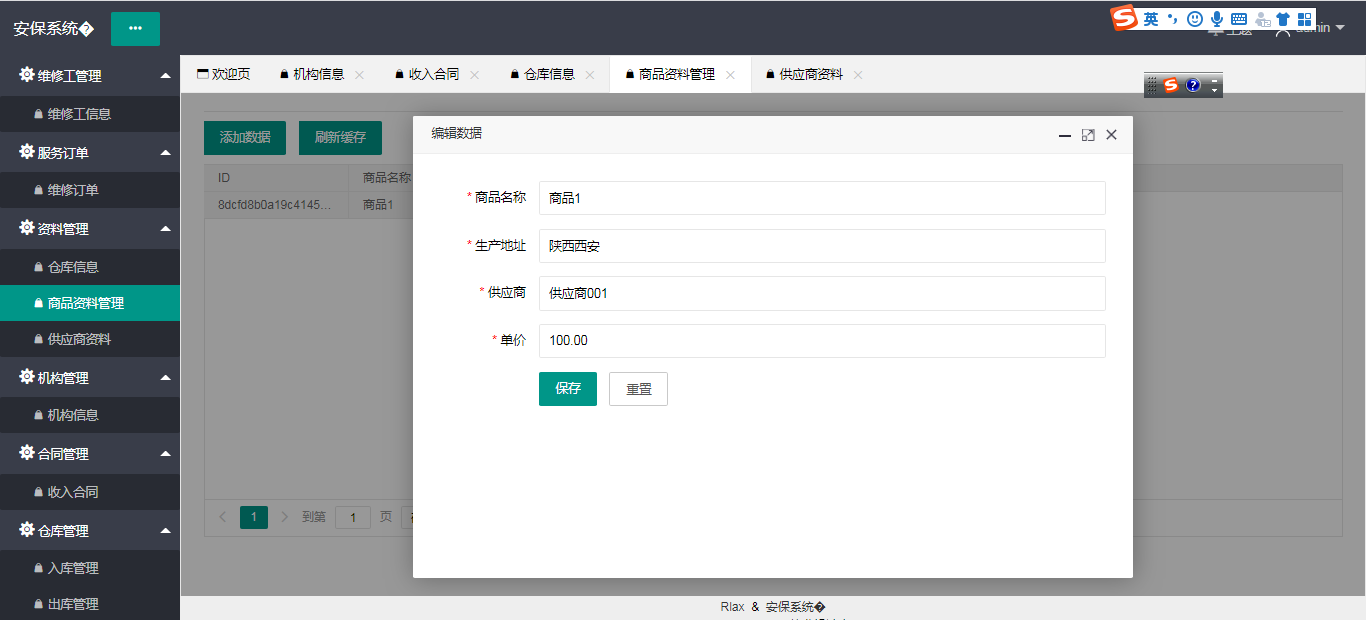


图 4-20 编辑商品数据

也可以对商品进行删除，鼠标点击列表信息后面的删除按钮，如下图4-21。

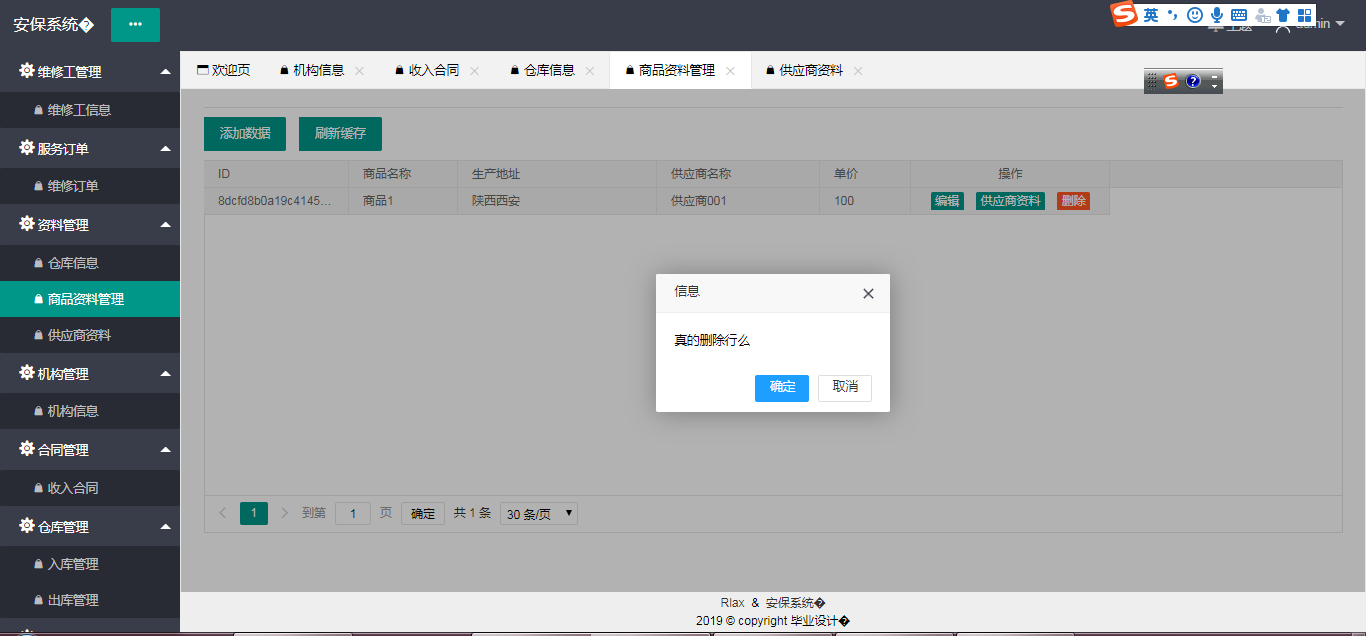


图 4-21 删除商品信息

对商品供应商详情的查看，如图4-22。



图 4-22 查看供应商信息

### 4.5.3仓库模块

仓库模块是对出入库提供服务，仓库信息，它对于具有操作权限的用户来说具有增删改查操作，仓库信息的处理，方便在之后出入库时更加方便物料所处的位置和仓库信息。如图4-23

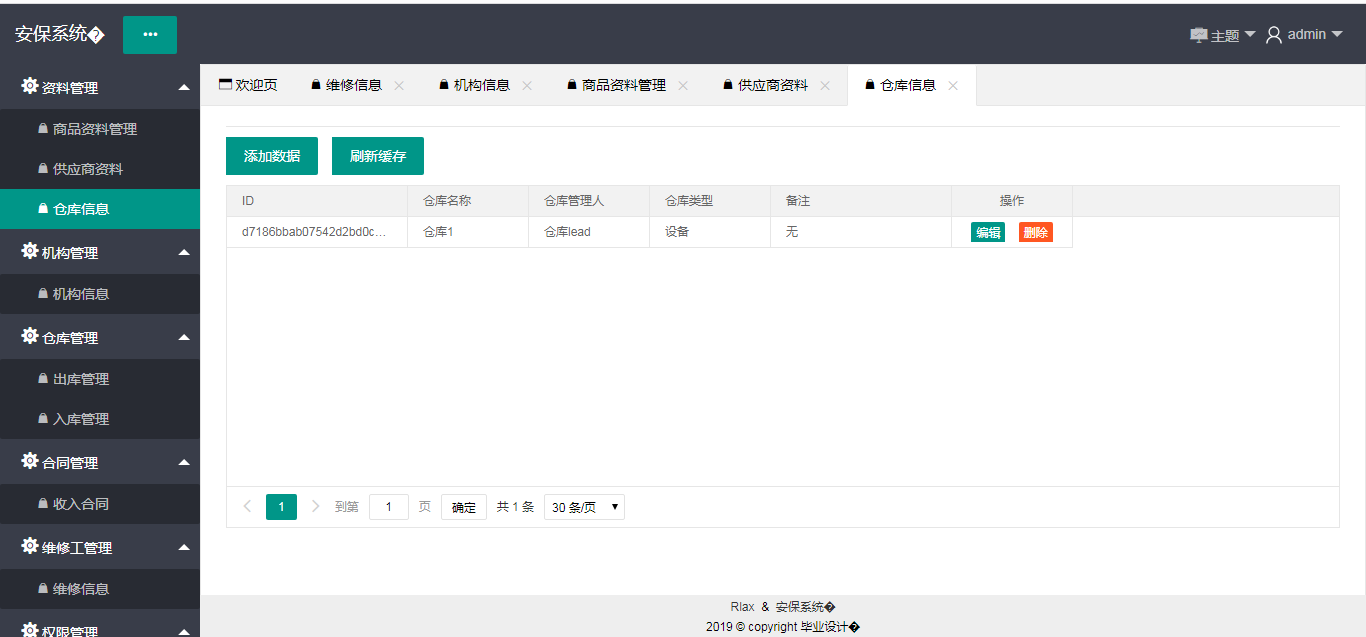


图 4-23 仓库信息列表

### 4.5.4出入库

出入库包括入库部分和出库部分。

入库部分，主要是对从供应商那里得到的物料进行录入系统，并将其放入仓库，并且可以查看进入仓库的合同，方便之后出库时操作。

出库部分：当用户下单之后，就会生成订单等信息，维修工在进行维修的时候，就会对所使用的物料进行出库操作，通过申请物料之后，系统会根据维修工的表单，生成出库信息。有权限的用户的用户可以对其进行操作，如图4-24。

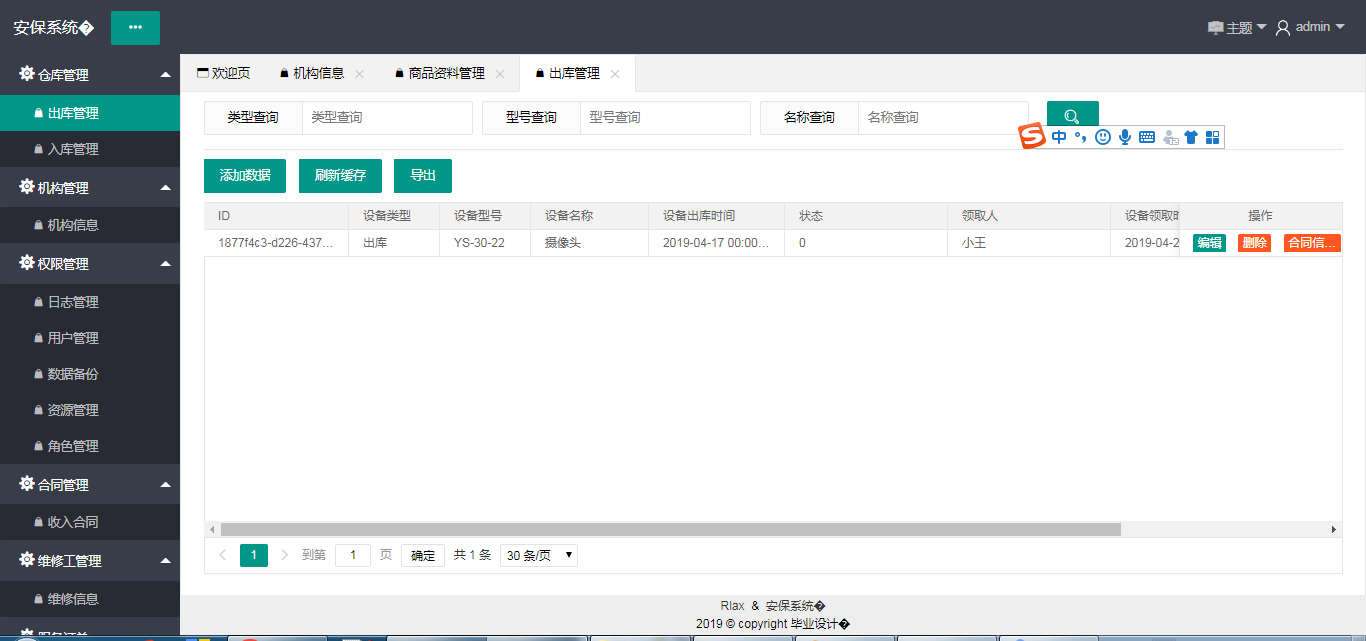


图 4-24 出库信息列表

## 4.6订单管理

用户下单之后就可以在这里看到，用户下单之后，可以在这里看到订单名称，服务机构，以及服务的网点和负责人等信息，这里还可以对维修工是否进行接单，以及接单时间、处理时间、以及处理完成的结束时间进行查看。同时也提供了附件的查看，通过图片的方式进行了展示，如图4-25。

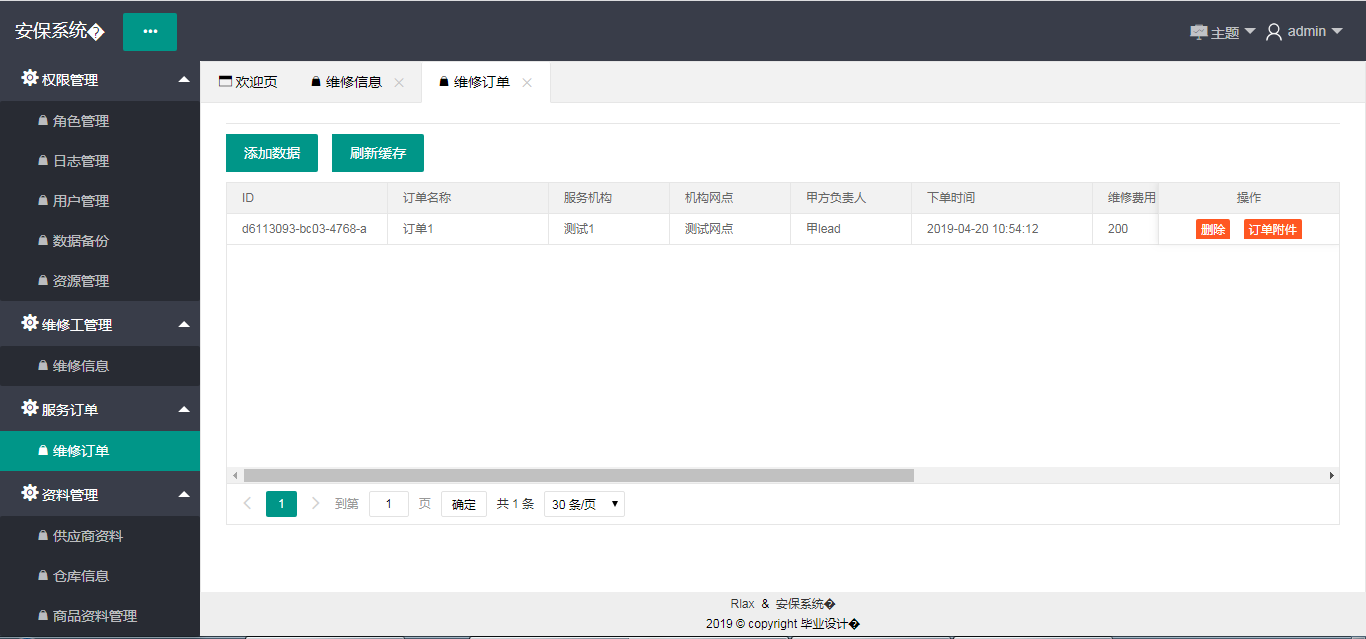


图 4-25 维修订单

## 4.7权限管理模块

权限模块包括有资源管理模块、角色管理模块、权限分配模块三个部分共同来实现的系统权限的管理。

资源管理：这里的布局不同于其他的页面，这里使用的是一个树形结构进行展示的，通过树形结构，方便查看每一个模块所包含的资源，和每一个模块所包含的操作。右边是一个增加的布局，可以对右边的列表项进行增删改查操作，对资源进行处理，如图4-26。

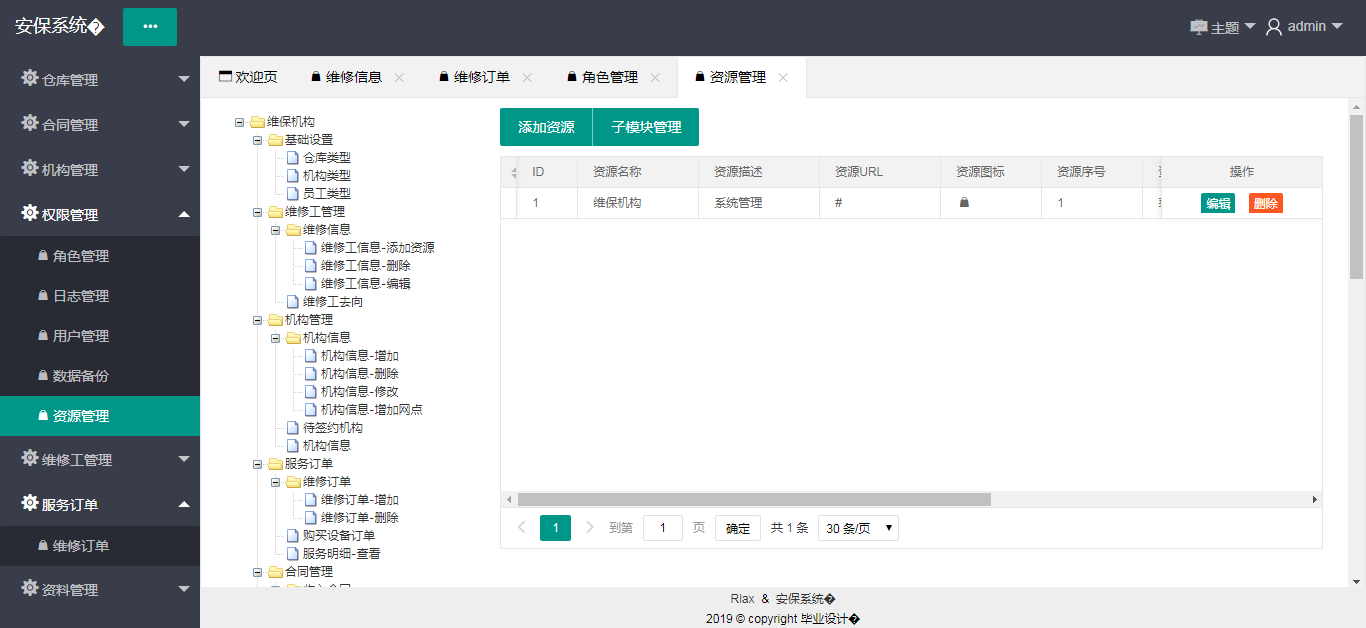


图 4-26 资源列表

## 4.8角色管理

对于权限管理是非常重要的，这种的处理方式是将上一部分的资源分配给一个角色，这个角色将会拥有给它分配的权限，在以后如果有新用户加入到系统的时候就会将角色给某个用户，当用户拥有了角色所具有的资源之后，就会获取到自身范围内的权限，对系统进行操作，如图4-27。

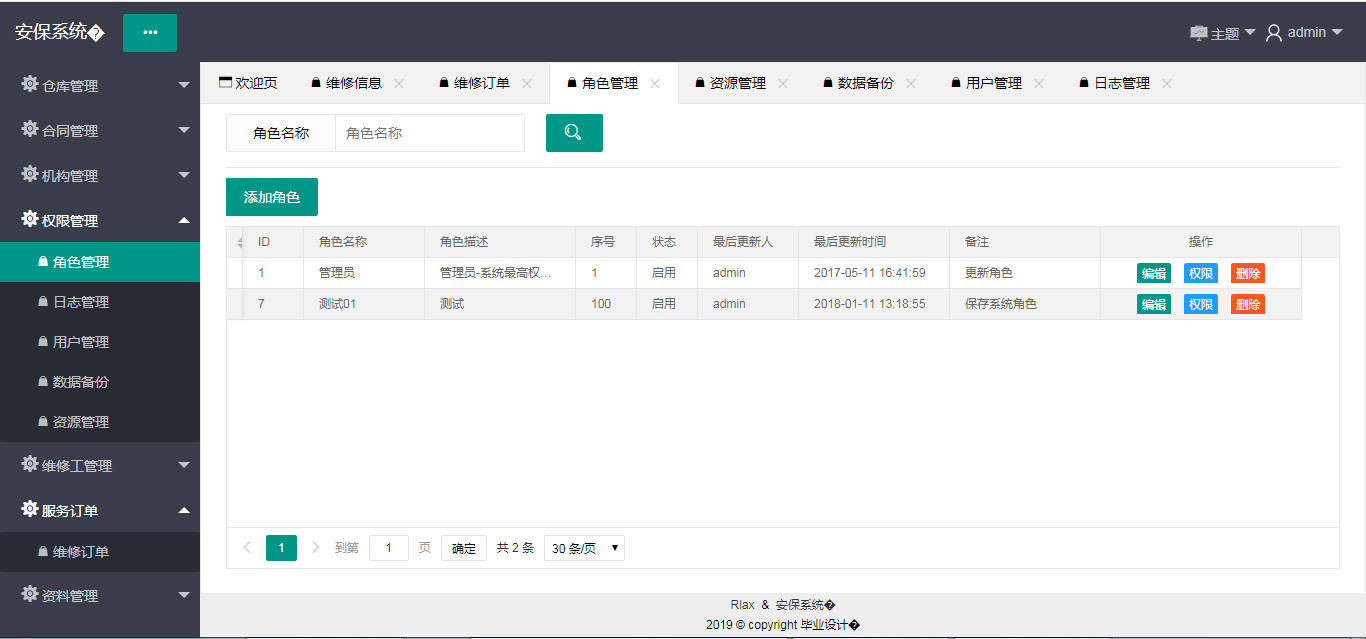


图 4-27 角色管理

用户管理：用户管理这里已经没有其它更加多的功能了，这里只可以进行用户的增加、删除和修改，如图4-28。

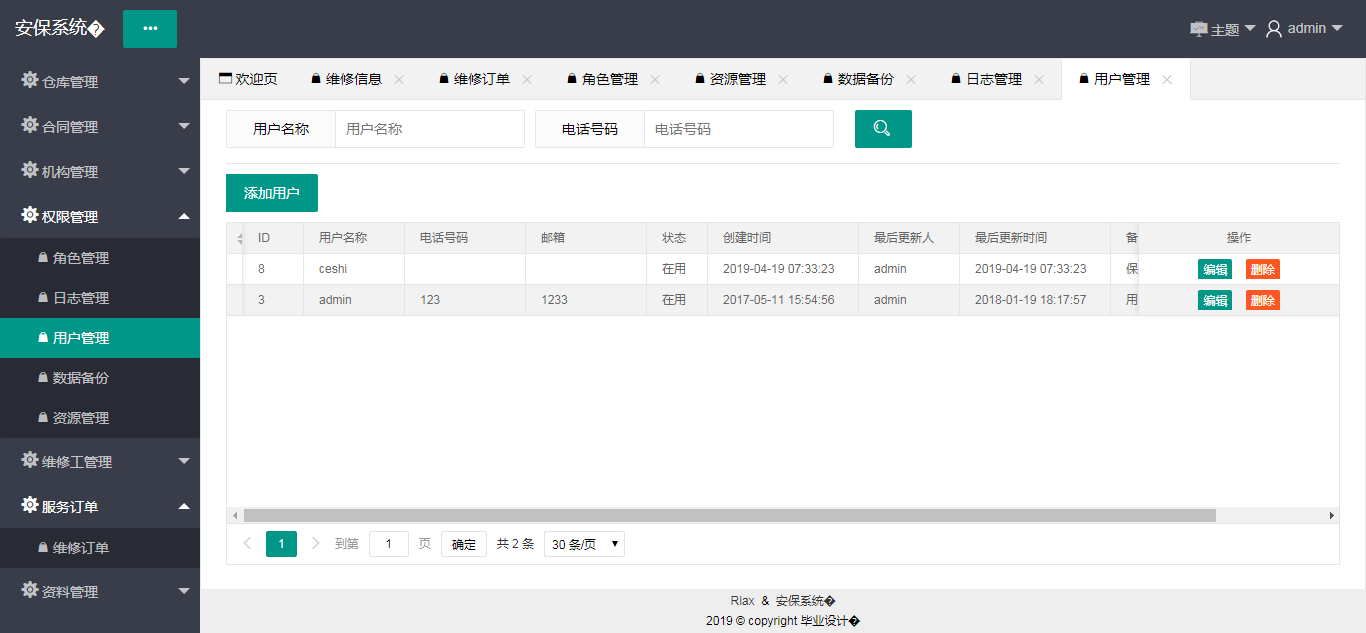


图 4-28 用户管理

## 4.9日志

日志记录了系统登录的用户、登录的时间、登录IP、浏览器、访问时间，可以通过页面进行查看，同时也为日志提供了速索操作功能，可以通过用户名，IP和URL进行速索。因为数据比较多，所以采用分页技术，使用layui框架对分页有良好的支持，可以很方便的进行分页，如图4-29。

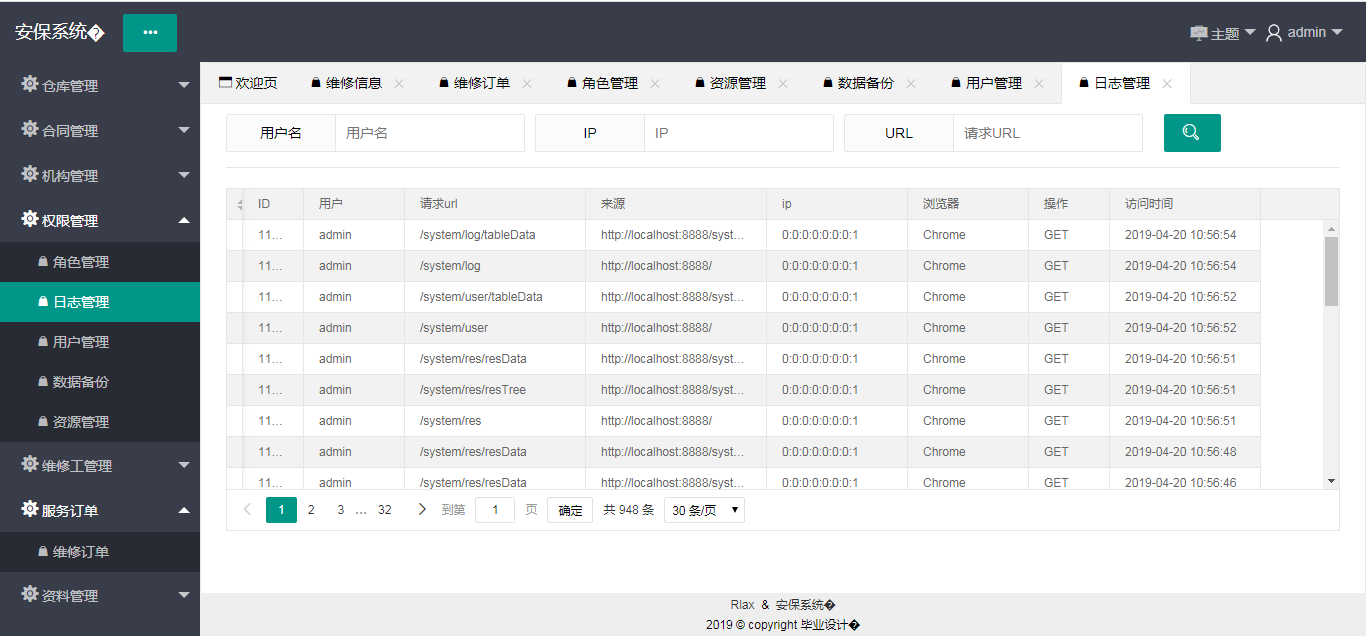


图 4-29 日志列表

## 4.10退出系统

使用结束可以点击用户图像，然后退出系统。退出系统之后会返回到登录页面。

# 5系统测试

项目测试是为了保证系统安全可靠的主要手段，测试阶段需要排除系统中潜在的错误，最终将一个可靠的、稳定的的、安全的产品给交付给客户使用。

系统测试的用例是系统测试中的关键技术，它的基本要求是，少量的测试数据，但是这些数据需要全面的的、完善的将系统进行测试，尽可能的发现系统存在的问题和缺陷，并且在测试中需要对系统的不足或者错误进行记录.本系统的部署之后，通过浏览器进行访问所有的功能和页面，并查看每个功能模块的功能时候完善，排除潜在的问题。测试分为以下几个部分：

## 5.1功能测试

每个超链接的测试，首先对所有的连接进行测试，是否正确的通过连接到达指定的页面，并且查看所连接到的页面是否正确渲染，每个页面上的提示框等是否正确弹出，并且检查是否有的页面成为独立的页面，没有连接链到它们。

表单测试：在本系统中表单比较多的，从最开始的登录一直到里面的整改操作都离不开表单，测试的时候，需要确保表单的正确性，以及效验信息的正确性，比如智能输入电话号、数字、非负数等这类验证。

数据测试：最终的一部分，数据库提供着整个系统中的数据管理工作，它的失效、宕机都会造成整个系统的瘫痪，这一部分主要验证的有用户提交的字段是否和数据的字段一致，数据库返回数据的异常。针对用户的误操作或者是恶意操作，可以通过页面的表单验证进行处理，对于返回数据的错误或者异常，这个需要注意设计的逻辑。然后对其进行测试。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试项 | 操作步骤 | 预期结果 | 实现结果 | 比较 |
| 点击每个超链接 | 使用鼠标点击 | 每个超链接都可以正确链接 | 点击正确 | 通过测试 |
| 表单效验 | 添加正确数据和非法数据 | 正确无提示非法有错误提示 | 正确无提示非法有错误提示 | 通过测试 |
| 数据库测试 | 页面表单输入数据 | 非法数据被拦截，正确数据操作数据库 | 和预期相同 | 通过测试 |

## 5.2可用性测试

系统中的按钮以及页面中可点击事件测试。当用户欲进入系统的时候，对于浏览器打开页面中可以点击或者是可以处理的事件进行操作的时候，确保每一个按钮或者是点击事件可以正确的做出响应和处理并且对于页面中的布局在测试的时候进行完善，确保每一个布局的用途和作用，最重要的是用户的使用习惯和体验，尽量使得页面美观、舒适、大方。

兼容性：使用不同的浏览器进行测试，用户使用的浏览器多种多样，需要满足各种浏览器的要求，所以测试的时候，需要满足主流浏览器展示。在不同的版本之间进行兼容，不同构建设置的适应性。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试项 | 操作步骤 | 预期结果 | 实现结果 | 比较 |
| 页面  按钮 | 点击所有  页面按钮 | 正确处理  点击事件 | 正确处理  点击事件 | 通过  测试 |
| 浏览器  兼容性 | 使用主流浏览器打开，进行操作 | 每个页面都和设计时相同 | 不同浏览器打开都和预期相同 | 通过  测试 |

## 5.3安全性测试

主要是对用户的输入进行验证，输入数字的地方不要输入字符串，输入日期的地方不要输入数字，用户名长度的，密码规格进行检查，确保系统的安全可用。不同的用户登录，出现本用户登录所拥有的权限界面。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试项 | 操作步骤 | 预期结果 | 实现结果 | 比较 |
| 页面输  入效验 | 输入框中输入非法数据 | 提示输入非法数据 | 非法数据有提示 | 通过  测试 |
| 用户  登录 | 使用不同角色的用户登录 | 不同角色显示自身权限内的布局 | 页面没有显示权限之外的按钮等 | 通过  测试 |

# 结论

本系统通过，原型设计、数据库设计、前端页面书写、后端代码书写等操作，直到论文完成，花费了几个月的时间和精力。在黄老师的指导下完成了毕业设计，大学的最后一场考试，从中发现了自己的众多不足和缺陷，发现自身的弊病，无论是知识还是自己习惯上的，在完成的过程中查阅了好多的资料和文献，查看了好多博客，也感谢同学的热心帮助。到此论文到此结束，在接下来的快要离校的日子里，将自己的不足和缺陷抓一下。本次设计实现了银行维修后台系统的一部分功能，主要的功能如下：

（1）支持平台后端对维修工信息进行操作，实现了信息管理的基本功能。

（2）支持超级管理员和普通的用户登录系统。

（3）支持权限分配、权限资源管理。

（4）支持维保机构信息管理，包括维保机构网点的管理。

（5）支持仓库的管理，商品资料的管理。

（6）支持订单的处理，下单和处理订单。

（7）支持维保机构的合同管理。

在首页上可以对页面进行简单的统计，包括维保机构信息的展示，以及网点的统计部分，通过图表的方式进行展示，从而使得维修统计一体化。在设计中也发现了很多的不合理，还需要继续的进行完善。

# 致 谢

经历这几个的学习和摸索，终于完成了毕业设计，在制作过程中也遇到了很多问题和麻烦，都在同学和老师的指导下克服了。首先，感谢我的导师黄健老师的指导!在题目设计和论文撰写工作中，黄健老师对我要求十分严格，在制作过程中并进行了认真指导，给予了很多鼓励和帮助，从中学到了许多新的知识和经验。黄健老师对他指导的学生认真负责并且它也比较的平易近人，在论文书写过程了对我们进行了指导，并且不厌其烦的帮我们解答问题，修改论文中有问题的地方。

在毕设设计过程中，在老师的带领下学习了许多的软件和平台，并且在老师的悉心指导的过程中学习到了许多平时上课无法获得的知识，老师渊博的知识、严谨的教学态度，深深的影响着我们，也会我们之后工作学习起到了模范的作用，也感谢所查询资料的作者，从中学到了很多知识和启发。

其次，感谢身边同学和各位老师们的指导和帮助!

最后，谨向所有关心和帮助过我的老师、同学和家人表示我最真诚的谢意!

# 参考文献

[1] 周亚辉. Java数据库系统项目开发实践[M]. 北京: 科学出版社, 2007

[2] 王能斌. 数据库系统教程(第2版)(上册)[M]. 北京:电子工业出版社, 2008

[3] 萨师煊, 王珊. 数据库系统概论[M]. 北京:高等教育出版社, 2001

[4] 赵增敏. JSP动态网页设计[M]. 北京: 电子工业出版社, 2003

[5] 邓文渊, 陈惠贞, 陈俊华. 挑战JSP与网页数据库设计[M]. 北京: 中国铁路出版社,2004

[6] 宣小平. JSP数据库系统开发实例导航[M]. 北京:人民邮电出版社, 2003

[7] 罗娟. Web与数据库技术[J]. 计算机工程.1998,24(8)

[8] 徐进明. JSP网站开发技术[M]. 北京: 中国工业出版社, 2003.2

[9] 汤承林, 徐春艳, 刘三满. SQL Server数据库应用基础与实现[M]. 北京: 电子工业出版社, 2008. 5

[10]樊海玮。基于WEB的网络化在线考试系统研究与实现.硕士学位论文，长安大学，2006：9-11