D{5}1A{N8WS26{M7`62$WXC copy

JIANGXI AGRICULTURAL UNIVERSITY

**本 科 毕 业 论 文 设 计**



题目**： 基于WEB的实迈义齿物料管理系统**

**学 院： 信息与技术学院**

**姓 名： 谭琪**

**学 号： 20162483**

**专 业： 计算机科学与技术**

**班 级： 计科1602**

**指导教师： 欧阳文臣老师**

**二0一 九 年 六 月 五 日**

[摘要 I](#_Toc17119)

[Abstract II](#_Toc19084)

[1 绪论 1](#_Toc16958)

[1.1 开发背景和意义 1](#_Toc26602)

[1.2 系统开发关键技术与开发环境 1](#_Toc26059)

[1.2.1 WEB语言 1](#_Toc8170)

[1.2.2 javaScript与jquery 2](#_Toc20324)

[1.2.3 Spring框架 2](#_Toc1918)

[1.2.4 SpringMVC框架 2](#_Toc1918)

[1.2.5 Mybatis框架 2](#_Toc1918)

[1.2.6 Shiro框架 2](#_Toc1918)

[1.2.7 Apache服务器 2](#_Toc1918)

[1.2.8 MySQL数据库 2](#_Toc19626)

[1.2.9 系统开发环境 2](#_Toc2665)

[2 可行性分析 3](#_Toc26708)

[2.1 技术可行性 3](#_Toc3150)

[2.2 经济可行性 3](#_Toc7584)

[2.3 社会可行性 3](#_Toc3738)

[3 系统需求分析 5](#_Toc14024)

[3.1 功能性需求分析 5](#_Toc20253)

[3.1.1 登录模块 5](#_Toc3311)

[3.1.2 首页模块 6](#_Toc21175)

[3.1.3 库存列表 6](#_Toc2200)

[3.1.4 入库记录模块 6](#_Toc630)

[3.1.5 出库记录模块 6](#_Toc28363)

[3.1.6 新增物料模块 6](#_Toc22191)

[3.1.7 新增供应商模块 6](#_Toc22191)

[3.1.8 管理员列表模块 6](#_Toc22191)

[3.2 非功能性需求分析 7](#_Toc10313)

[3.2.1 安全性 7](#_Toc10201)

[3.2.2 可靠性 7](#_Toc23427)

[3.2.3 易用性 7](#_Toc15146)

[3.2.4 维护性 7](#_Toc14129)

[3.2.5 可移植性 7](#_Toc4434)

[3.3 系统优势 8](#_Toc31422)

[4 项目数据库设计 9](#_Toc28608)

[4.1 数据库的选择与创建 9](#_Toc29404)

[4.2 系统全局E-R图 9](#_Toc8528)

[4,3 数据库E-R图 9](#_Toc5124)

[4.4 数据表设计 12](#_Toc4456)

[5. 项目系统详细数据 16](#_Toc27055)

[5.1 系统流程及开发流程 16](#_Toc15541)

[5.2 视觉设计 16](#_Toc2177)

[5.3 主要功能模块实现 17](#_Toc25291)

[5.3.1 登录板块 17](#_Toc9002)

[5.3.2 首页实现 18](#_Toc6662)

[5.3.3 库存列表板块 19](#_Toc23354)

[5.3.4 入库列表板块 17](#_Toc9002)

[5.3.5 出库列表板块 18](#_Toc6662)

[5.3.6 新增物料板块 19](#_Toc23354)

[5.3.7 新增供应商板块 17](#_Toc9002)

[5.3.8 管理员列表板块 17](#_Toc9002)

[6. 项目系统测试 21](#_Toc32053)

[6.1 测试环境 21](#_Toc8629)

[6.2 测试方法 21](#_Toc9041)

[6.2.1 用户界面测试 21](#_Toc26068)

[6.2.2 单元测试 21](#_Toc18995)

[6.2.3 性能测试 21](#_Toc21389)

[6.2.4 回归测试 21](#_Toc9614)

[6.2.5 验收测试 21](#_Toc28941)

[6.3 浏览器测试 22](#_Toc26361)

[7. 结论与展望 23](#_Toc25901)

[7.1 结论 23](#_Toc20254)

[7.2 后期展望 23](#_Toc5716)

[参考文献 24](#_Toc22916)

[致谢 25](#_Toc3495)

# 摘要

# Abstract

# 绪论

## 开发背景和意义

## 

## 1.2 系统开发关键技术与开发环境

### 1.2.1 WEB语言

WEB语言分为静态语言和WEB动态语言，WEB静态语言就是通常所见到的[超文本标记语言](https://baike.baidu.com/item/%E8%B6%85%E6%96%87%E6%9C%AC%E6%A0%87%E8%AE%B0%E8%AF%AD%E8%A8%80" \t "https://baike.baidu.com/item/WEB%E7%BC%96%E7%A8%8B%E8%AF%AD%E8%A8%80/_blank) （[标准通用标记语言](https://baike.baidu.com/item/%E6%A0%87%E5%87%86%E9%80%9A%E7%94%A8%E6%A0%87%E8%AE%B0%E8%AF%AD%E8%A8%80" \t "https://baike.baidu.com/item/WEB%E7%BC%96%E7%A8%8B%E8%AF%AD%E8%A8%80/_blank)下的一个应用 ），WEB动态语言主要是[ASP](https://baike.baidu.com/item/ASP/128906" \t "https://baike.baidu.com/item/WEB%E7%BC%96%E7%A8%8B%E8%AF%AD%E8%A8%80/_blank)，[PHP](https://baike.baidu.com/item/PHP/9337" \t "https://baike.baidu.com/item/WEB%E7%BC%96%E7%A8%8B%E8%AF%AD%E8%A8%80/_blank)，JAVASCRIPT，[JAVA](https://baike.baidu.com/item/JAVA/85979" \t "https://baike.baidu.com/item/WEB%E7%BC%96%E7%A8%8B%E8%AF%AD%E8%A8%80/_blank)，CGI等计算机[脚本语言](https://baike.baidu.com/item/%E8%84%9A%E6%9C%AC%E8%AF%AD%E8%A8%80/1379708" \t "https://baike.baidu.com/item/WEB%E7%BC%96%E7%A8%8B%E8%AF%AD%E8%A8%80/_blank)编写出来的执行灵活的互联网[网页](https://baike.baidu.com/item/%E7%BD%91%E9%A1%B5/99347" \t "https://baike.baidu.com/item/WEB%E7%BC%96%E7%A8%8B%E8%AF%AD%E8%A8%80/_blank)程序

随着软件行业的快速发展，web语言制作上已经不仅仅是简单的界面了，可以通过开发粗出来的各种前端框架进行界面的设计，实现界面功能的校验，以及传递数据到后台的功能；

### 1.2.2 javaScript与jquery

javaScript是一种非常松散的面向对象语言，也是WEB开发中极受欢迎的一门语言，轻量级，基于对象，基于浏览器内核以及事件驱动的客户端脚本语言，毫不夸张的说，它是整个WEB前端开发的核心技术。Js语言的弱类型使得js的编程更加随性不被语法束缚过多而更加注重逻辑。不论是目前最流行的主流框架vue、node还是react它们的核心技术依旧是javascript语言。Js具有相对安全、跨平台、事件驱动等众多优点

接下来介绍的jquery是目前最流行的轻量级javascript库，它的轻量级是其他同类js库不及的，服务器压缩后的jquery仅20kb上下的大小。若用一句话形容jquery，那便是“用最少的代码做最多的事”。Jquery在js的基础上对其进行了封装，使得原先的代码编程量在jquery这里只需要几句代码便可解决，并且解决了js在各主流浏览器的兼容问题。具有可扩展性强，允许在命名空间上增加新函数、ajax支持、跨浏览器等众多优点

### 1.2.3 Spring框架

### Spring框架是由于软件开发的复杂性而创建的。Spring使用的是基本的JavaBean来完成以前只可能由EJB完成的事情。然而，Spring的用途不仅仅限于服务器端的开发。从简单性、可测试性和松耦合性角度而言，绝大部分Java应用都可以从Spring中受益，在实际的开发过程中Spring框架可以很好的与程序相结合，减少代码的冗余度，管理JavaBean的生命周期，提供更为简单的操作代码的方式。

1.2.4 SpringMVC框架

很多应用程序的问题在于处理业务数据的对象和显示业务数据的视图之间存在紧密耦合，通常，更新业务对象的命令都是从视图本身发起的，使视图对任何业务对象更改都有高度敏感性。而且，当多个视图依赖于同一个业务对象时是没有灵活性的。SpringMVC是一种基于Java，实现了Web MVC设计模式，请求驱动类型的轻量级Web框架，即使用了MVC架构模式的思想，将Web层进行职责解耦。基于请求驱动指的就是使用请求-响应模型，框架的目的就是帮助我们简化开发，SpringMVC也是要简化我们日常Web开发。

### 1.2.5 MyBatis框架

MyBatis 是一款优秀的持久层框架，它支持定制化 SQL、存储过程以及高级映射。MyBatis 避免了几乎所有的 JDBC 代码和手动设置参数以及获取结果集。MyBatis 可以使用简单的 XML 或注解来配置和映射原生信息，将接口和 Java 的 POJOs(Plain Old Java Objects,普通的 Java对象)映射成数据库中的记录。

### 1.2.6 Shiro框架

Apache Shiro是一个强大且易用的Java安全框架,执行身份验证、授权、密码和会话管理。使用Shiro的易于理解的API,您可以快速、轻松地获得任何应用程序,从最小的移动应用程序到最大的网络和企业应用程序。基本上涉及到用户参与的系统都要进行权限管理，权限管理属于系统安全的范畴，权限管理实现对用户访问系统的控制，按照安全规则或者安全策略控制用户可以访问而且只能访问自己被授权的资源。权限管理包括用户身份认证和授权两部分，简称认证授权。对于需要访问控制的资源用户首先经过身份认证，认证通过后用户具有该资源的访问权限方可访问。

### 1.2.7 Apache服务器

Apache是目前最流行的WEB应用服务器之一和开放源码的HTTP服务器，据统计，它在世界上的使用率排名第一，几乎可以在所有广泛使用的计算机上运行。具有快速可靠以及通过API扩展等优点

### 1.2.8 MySQL数据库

和其他数据库系统相比，MySQL有点与众不同。MySQL并不完美，却足够灵活，能够适应高要求的环境，例如Web类应用。同时，MySQL既可以嵌入到应用程序中，也可以支持数据仓库、内容索引和部署软件、高可用的冗余系统、在线事务处理系统（OLTP）等各种应用类型。

### 1.2.6 系统开发环境

数据库：MYSQL数据库；

开发环境：SSM+Shiro集成环境；

代码编辑器：HBuilder，Eclipse编辑器；

调试工具：chrome，IE调试工具。

# 2 可行性分析

## 2.1 技术可行性

## 2.2 经济可行性

## 2.3 社会可行性

# 3 系统需求分析

从软件工程的理论基础出发理解，需求分析在整个软件生命周期中是最重要的一个阶段。软件需求分析报告的质量对软件开发的前期功能设计和后期维护以及扩展等影响是非常深远，它的存在意义十分巨大。高质量的软件需求分析对软件的整个开发过程将起到事半功倍的作用，有句话说的好，“磨刀不误砍柴工”。若前期无法准确用户需求和产品功能需求，后续的需求修改将会导致和带来一定程度甚至巨大的损失。

该系统分为多个模块，图3-1是该系统功能分析图。

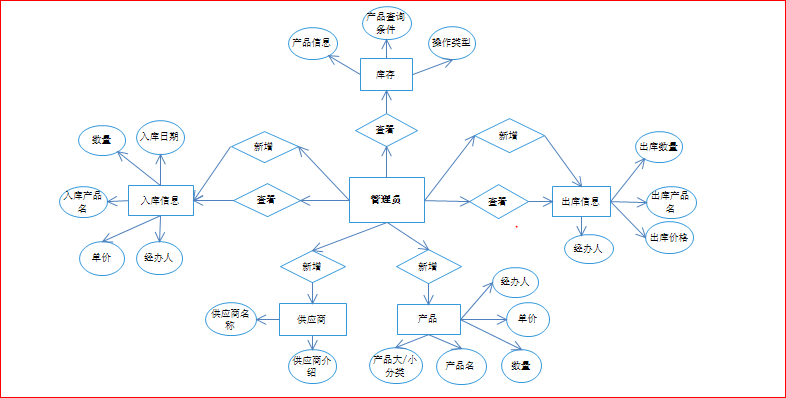


图3-1是系统功能分析图

## 3.1 功能性需求分析

### 3.1.1 登录模块

在登录模块，浏览器输入URL地址，首先进行的是拦截判断是否登录，是：直接进行系统首页执行操作；否：跳转登录界面，填写对应的登录信息，并且将登录数据进行验证，根据Shiro框架解析当前的数据进行验证：验证成功进行首页跳转的同时，判断用户权限，根据用户权限显示不同的菜单列表; 验证失败跳转登录界面，并且提示错误信息

### 3.1.2 首页模块

登录成功的用户进入首页列表，根据用户权限显示不同的菜单列表，根据点击首页左侧菜单进行对应列表界面的显示

普通管理员菜单列表为：首页，库存列表，入库记录，出库记录，新增物料，新增供应商

超级管理员菜单列表为：首页，库存列表，入库记录，出库记录，新增物料，新增供应商

### 3.1.3 库存模块

库存列表模块主要是给管理员显示当前数据库产品表中产品的库存信息，对数据进行分页显示，根据管理员点击的页数，展示不同页的库存数据; 管理员可以在此模块中根据自己选择的大类分类编号、小类分类编号、查询对应的产品库存数据信息，也可以根据输入的产品名称进行查询相应的产品信息；在此模块管理员可以选中产品并且对产品执行入库、出库操作、还可以进行批量的删除产品数据；点击“出库查询”、“入库查询”、“新产品入库”进行跳转不同的界面

### 3.1.4 入库记录模块

该模块向管理员提供当前所有的入库产品信息，可以根据管理员输入的“产品名称”模糊匹配查询、输入日期、经办人、供应商查询对应入库记录；可以根据点击“批量删除入库记录”图标进行对应入库记录的删除，当未选择产品时，点击对应图标，给出对应的错误操作提示；

### 3.1.5 出库记录模块

### 该模块向管理员提供当前所有的出库产品信息，可以根据管理员输入的“产品名称”模糊查询、出库日期、经办人查询对应出库记录；可以根据点击“批量删除出库记录”图标进行对应出库记录的删除，当未选择产品时，点击对应图标，给出对应的错误操作提示；

### 3.1.6 新增物料模块

管理员输入新增产品的信息，对输入的产品信息进行校验（产品名是否唯一校验，其他输入信息是否符合小验格式）、校验都成功的情况下，进行产品的新增操作，插入数据库表中；

### 3.1.7 新增物料模块

管理员输入新增供应商的信息，对输入的信息进行校验（供应商名称是否唯一校验）,校验成功插入数据库表；

### 3.1.8 管理员列表模块

### 该模块只对超级管理员开放操作，起始展示管理员列表；超级管理员可以执行以下操作：

### 修改管理员信息操作：

### 选择管理员信息，跳转信息修改界面，执行修改；对修改后的界面进行校验（密码格式、状态码校验），校验都成功，修改数据库信息，跳转管理员记录列表；

新增管理员操作：

点击新增按钮，跳转新增界面，输入新增数据，对新增数据进行输入校验（管理员名称唯一、密码格式、状态码格式），校验成功，将数据插入数据库

删除用户操作：

根据选择的数据，进行删除（物理删除，状态修改为0）

## 3.2 非功能性需求分析

### 3.2.1 安全性

（1）没有登录的用户不允许进入系统执行操作

（2）管理员只有通过认证并且授权成功才能进行物料管理系统

（3）普通管理员没有操作管理其他管理员的权限，只有超级管理员具有该权限

### 3.2.2 可靠性

所谓可靠性，我的理解是当网站发生某种故障，该平台具有相对的容错率，适时调整情况而不至给用户带来困扰和损失，在发生错误时可以快速恢复至原先状态。在系统正式发布之前会进行各种功能测试，

### 3.2.3 易用性

该系统必须具备易用易理解易操作等基本特点，以便用户快速了解系统功能的核心。要求系统的模块规划合理，用户学习了解该系统的时间较短，操作及运行效率高，浏览内容简单快速避免在找寻信息中浪费不必要的时间。

### 3.2.4 维护性

要求项目的开发必须遵循模块化与分层的思想，将系统中的业务逻辑有规划地分成若干个以业务逻辑为单元的相互独立并且易于理解的模块，即函数或者方法，这样在后期系统的迭代中便更加利于维护。一些特殊的模块，像全局资源的访问，必须最好相应对策，最好只保证共享可读性，不被修改或者覆盖。其次，避免代码的重复性，消除代码的冗长性是很有必要的操作，形成软件产品代码高内聚低耦合

### 3.2.5 可移植性

避免模块化过程中对代码的封装传递过多的参数，优化代码设计，设计模块化定要避免高耦合情况的出现。系统运行的环境，开发的环境甚至是前端页面的展示均要求适应各个环境或者主流浏览器。在后期对系统进行维护的过程中，要考虑前期产品已展现的功能，避免改动过大，造成系统用户使用困难；

## 3.3 系统优势

良好的页面设计：本系统项目的前端设计，结构分层清晰，简单舒适又不缺大气感是给人最直观的感受，避免图片的压缩堆积造成用户的视觉疲劳，也不会使用引起人不适感的效果与图片。

# 4 项目数据库设计

## 4.1 数据库的选择与创建

该项目系统存储数据选择的是MYSQL数据库，数据库名称为kucun

## 4.2 系统全局E-R图

E-R图就是实体联系图，程序设计初期便需要通过实体联系图来确定实体与实体之间的相互关系，实体联系图能够清晰无误地表达出程序的结构，它是描述现实世界与概念模型最有效的方法之一。其中，“矩形框”表示的是实体，“椭圆图标”表示的则是实体属性，用“实心线段”将属性与相应的实体链接起来，此模型的组成包括以上三个方面：属性、实体、联系。结合该项目路系统概念模型的特点以及设计方法，建立E-R模型图如下。

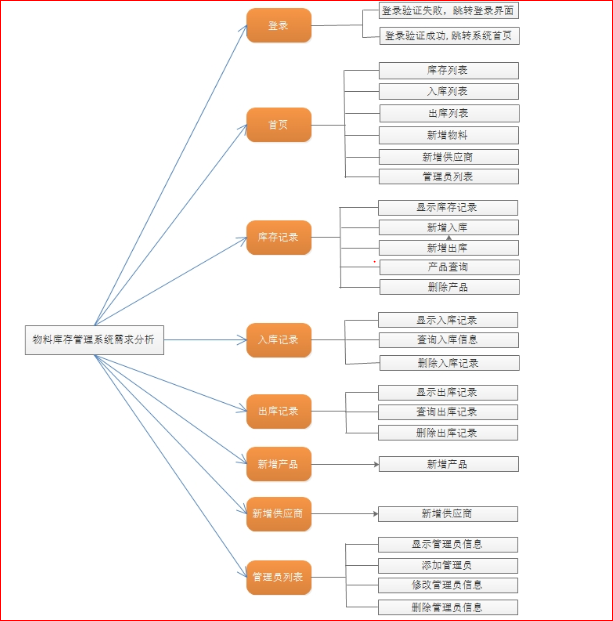


图4-1 数据库总体E-R图

## 4,3 数据库E-R图

1. 管理员实体主要用于存储管理员ID、用户名（username），用户密码(password)、用户状态(state)，加密盐值(salt)等信息。

管理员实体属性图如图4-2：

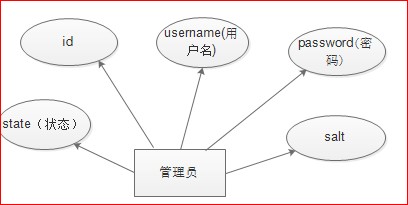


图4-2 管理员实体属性图

（2）角色实体主要用于存储角色ID、角色名（name）等信息。

角色实体属性图如图4-3：

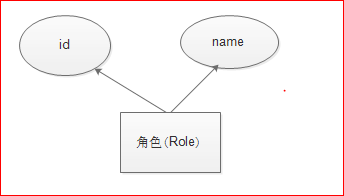


图4-3 角色实体属性图

1. 管理员-角色实体主要用于存储用户ID、角色ID等信息。

管理员-角色实体属性图如图4-4：

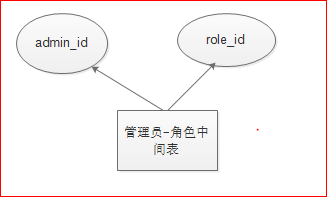


图4-4 管理员-角色实体属性图

（4）菜单实体主要用于存储菜单ID、菜单名称、菜单权限名等信息。

菜单实体属性图如图4-5：

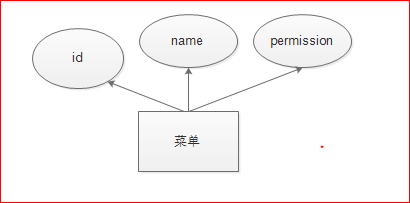


图4-5 菜单实体属性图

（5）菜单-角色实体主要用于存储菜单ID、角色ID等信息。

菜单实体属性图如图4-6：

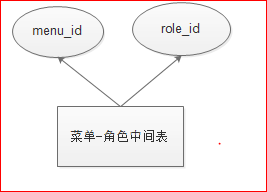


图4-6 菜单-角色实体属性图

（6）产品实体主要用于存储产品ID、菜单名称（name）、大类分类编号(bid)、小类分类编号(sid)、产品单位(units)、数量(count)、预警值(wvalue)、状态(state)、创建时间(createtime)、经办人(annt)等信息。

产品实体属性图如图4-7：

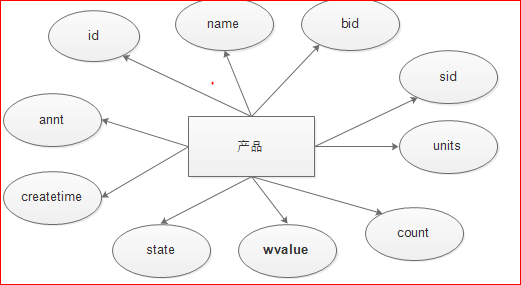


图4-7 产品实体属性图

（7）入库实体主要用于存储入库ID、产品名称（name）、入库数量(pcount)、单价（dprice）、总价（price）、经办人(annt)、供应商(pname)、入库时间(inputtime)、状态(state)等信息。

入库实体属性图如图4-8：

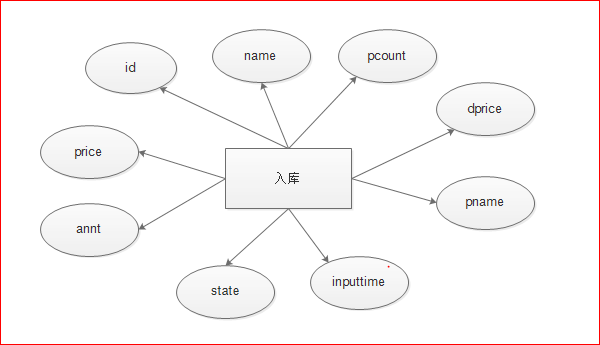


图4-8 入库实体属性图

（8）出库实体主要用于存储出库ID、出库产品名(name)、数量(pcount)、单价（price）、经办人(annt)、出库日期(outputtime)、状态（state）等信息。

出库实体属性图如图4-9：

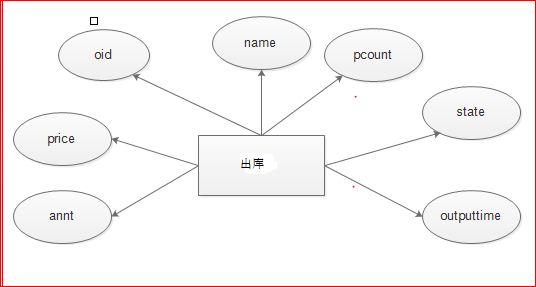


图4-9 出库实体属性图

（9）大类分类实体主要用于存储大类分类ID、分类名(name)、分类状态（state）等信息。

大类分类实体属性图如图4-10：

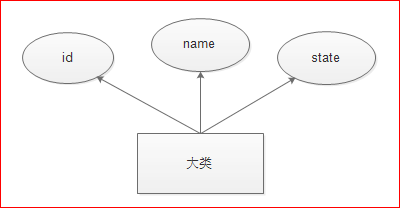


图4-10 大类分类实体属性图

（10）小类分类实体主要用于存储小类分类ID、分类名(name)、大类分类编号（bid）、分类状态（state）等信息。

小类分类实体属性图如图4-11：

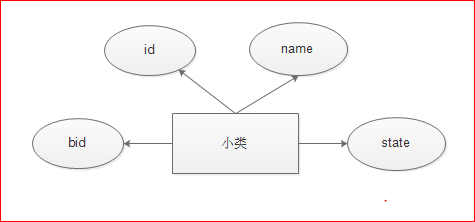


图4-11 小类分类实体属性图

（11）供应商实体主要用于存储供应商pid、供应商名称(pname)、供应商介绍(intro)等信息。

供应商实体属性图如图4-12：

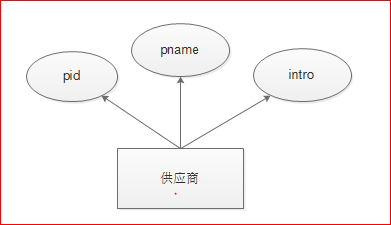


图4-12 供应商实体属性图

## 4.4 数据表设计

在数据库的规范设计前提下，建立以下基于物料库存管理系统需求的数据库表以及对应字段和各字段的介绍

1. 管理员信息表专门用于存储管理员的具体信息，表结构如图4-1所示

表4-1 管理员信息表（t\_admin)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 字段长度 | 是否允许为空 | 是否为主键 | 字段含义 |
| id | int | 11 | 不允许 | 是 | 管理员id号 |
| username | varchar | 30 | 不允许 | 否 | 登录用户名 |
| password | varchar | 32 | 不允许 | 否 | 登录密码（MD5加密处理） |
| state | int | 2 | 允许 | 否 | 状态（0：停用，1：可用） |
| salt | varchar | 20 | 允许 | 否 | 加密盐值 |

1. 角色表专门用来表明当前管理员对应的角色信息，表结构如4-2所示。

表4-2角色信息表（t\_role）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 字段长度 | 是否允许为空 | 是否为主键 | 字段含义 |
| id | int | 10 | 不允许 | 是 | 角色id |
| name | varchar | 20 | 允许 | 否 | 角色名 |

1. 管理员-角色信息表专门用来表明管理员和角色之间多对多的关系，4-3所示。

表4-3 管理员-角色 信息表（t\_admin\_role）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 字段长度 | 是否允许为空 | 是否为主键 | 字段含义 |
| admin\_id | int | 11 | 不允许 | 是 | 管理员id |
| role\_id | int | 11 | 不允许 | 是 | 角色id |

1. 菜单信息表专门用来保存系统中界面操作的菜单，表结构如4-4所示。

表4-4 菜单信息表(t\_menu)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 字段长度 | 是否允许为空 | 是否为主键 | 字段含义 |
| id | int | 11 | 不允许 | 是 | 菜单id |
| name | varchar | 11 | 允许 | 否 | 菜单名 |
| permission | varchar | 11 | 允许 | 否 | 操作菜单的权限 |

1. 菜单-角色信息表专门用来保存菜单和角色间多对多关系的中间表，表结构如4-5所示。

表4-5 菜单-角色信息表(t\_menu\_role)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 字段长度 | 是否允许为空 | 是否为主键 | 字段含义 |
| menu\_id | int | 11 | 不允许 | 是 | 菜单id |
| role\_id | int | 11 | 不允许 | 是 | 角色id |

1. 产品信息表专门用来保存系统各个产品的基本信息，表结构如4-6所示。

表4-6产品信息表(t\_products)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 字段长度 | 是否允许为空 | 是否为主键 | 字段含义 |
| id | int | 11 | 不允许 | 是 | 产品id |
| name | varchar | 50 | 不允许 | 否 | 产品名 |
| bid | int | 5 | 允许 | 否 | 大类id |
| sid | int | 5 | 允许 | 否 | 小类id |
| units | varchar | 32 | 允许 | 否 | 单位 |
| count | int | 11 | 允许 | 否 | 数量 |
| wvalue | int | 8 | 允许 | 否 | 预警值 |
| state | int | 2 | 允许 | 否 | 状态 |
| createtime | datetime | 0 | 允许 | 否 | 入库时间 |
| annt | varchar | 32 | 允许 | 否 | 经办人 |

1. 入库信息表专门用来保存入库产品的基本信息，表结构如4-7所示。

表4-7入库信息表(t\_input)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 字段长度 | 是否允许为空 | 是否为主键 | 字段含义 |
| id | int | 11 | 不允许 | 是 | 入库id |
| name | varchar | 32 | 不允许 | 否 | 入库产品名 |
| pcount | int | 8 | 允许 | 否 | 数量 |
| dprice | double | 10 | 允许 | 否 | 单价 |
| price | double | 15 | 允许 | 否 | 总价 |
| annt | varchar | 32 | 允许 | 否 | 经办人 |
| pname | varchar | 32 | 不允许 | 否 | 供应商 |
| inputtime | datetime | 0 | 允许 | 否 | 入库时间 |
| state | int | 2 | 允许 | 否 | 入库记录状态（1：可以显示，0：记录删除） |

1. 出库信息表专门用来保存产品出库的基本信息，表结构如4-8所示。

表4-8 出库数据表(t\_output)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 字段长度 | 是否允许为空 | 是否为主键 | 字段含义 |
| oid | int | 11 | 不允许 | 是 | 出库id |
| name | varcahr | 32 | 不允许 | 否 | 出库产品名 |
| pcount | int | 8 | 允许 | 否 | 数量 |
| price | double | 10 | 允许 | 否 | 单价 |
| annt | varchar | 32 | 允许 | 否 | 经办人 |
| outputtime | datetime | 0 | 允许 | 否 | 出库时间 |
| state | int | 2 | 允许 | 否 | 出库记录状态（1：可以显示，0：记录删除） |

1. 产品大类分类信息表专门用来保存产品的大分类基本信息，表结构如4-9所示。

表4-9 大类分类数据表(t\_bigtype)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 字段长度 | 是否允许为空 | 是否为主键 | 字段含义 |
| id | int | 5 | 不允许 | 是 | 分类id |
| name | varcahr | 32 | 不允许 | 否 | 分类名 |
| state | int | 2 | 允许 | 否 | 分类状态（1：可用，0：不可用） |

（10）产品小类分类信息表专门用来保存产品的小分类基本信息，表结构如4-10所示。

表4-10 小类分类数据表(t\_smalltype)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 字段长度 | 是否允许为空 | 是否为主键 | 字段含义 |
| id | int | 5 | 不允许 | 是 | 分类id |
| name | varcahr | 32 | 不允许 | 否 | 分类名 |
| bid | int | 5 | 允许 | 否 | 大类id |
| state | int | 2 | 允许 | 否 | 分类状态（1：可用，0：不可用） |

（11）产品供应商信息表专门用来保存库存系统供应商的基本信息，表结构如4-11所示。

表4-11 产品供应商信息表(t\_provider)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 字段长度 | 是否允许为空 | 是否为主键 | 字段含义 |
| pid | int | 11 | 不允许 | 是 | 供应商id |
| pname | varcahr | 50 | 不允许 | 否 | 供应商名称 |
| intro | varchar | 200 | 允许 | 否 | 供应商介绍信息 |

# 项目系统详细数据

## 5.1 系统流程及开发流程

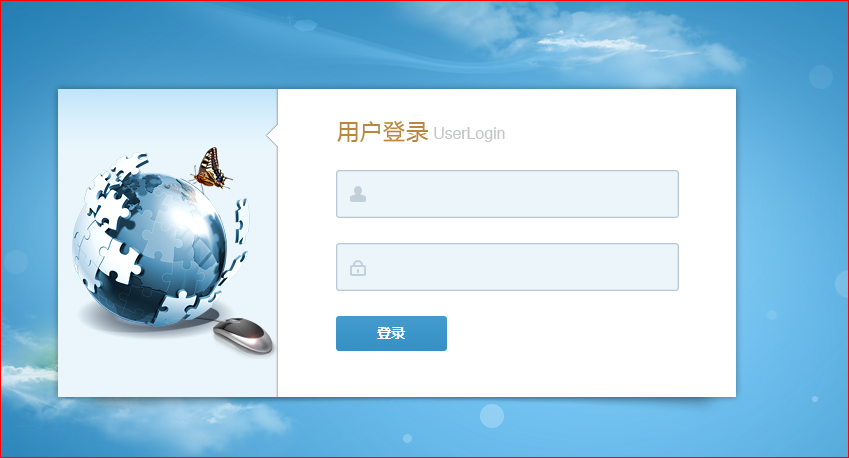
## 5.2 视觉设计

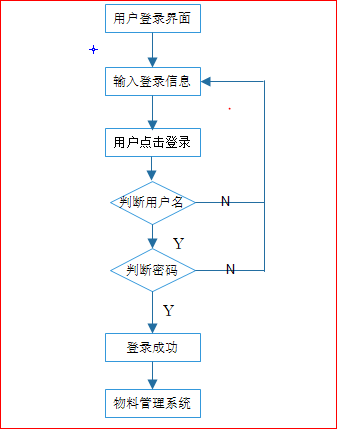
## 5.3 前端交互设计

## 5.4 主要功能模块实现

### 5.4.1 登录板块

输入登录信息验证成功进行首页跳转的同时，判断用户权限，根据用户权限显示不同的菜单列表; 验证失败跳转登录界面，并且提示错误信息

 图5-1 登录界面



1. 后台登录验证核心代码

//获取用户的身份凭证信息

String username = (String) token.getPrincipal();

TAdmin admin = adminServiceImpl.selByName(username);

if(admin==null) {

//用户不存在

return null;

}else if(admin.getState()!=1) {

//代表此时管理员为禁用状态

return null;

}

//此处代表使用了盐值进行加密

SimpleAuthenticationInfo info = new SimpleAuthenticationInfo(admin, admin.getPassword(),ByteSource.Util.*bytes*(admin.getSalt()), getName());

//此处代表使用了盐值进行加密

if(info!=null) {

//将成功的用户信息存储在session中

SecurityUtils.*getSubject*().getSession().setAttribute("loginAdmin", admin);

}

return info;

1. 用户授权核心代码

//得到权限字符串对象

SimpleAuthorizationInfo info = new SimpleAuthorizationInfo();

//获取用户的凭证信息

TAdmin admin = (TAdmin)principals.fromRealm(getName()).iterator().next();

List<TAdminRoleKey> roIdList = roleServiceImpl.selByAid(admin.getId());

if(roIdList!=null&&roIdList.size()>0) {

//存在用户角色

for (TAdminRoleKey tAdminRoleKey : roIdList) {

List<TMenuRoleKey> menuIdList = menuServiceImpl.selByRid(tAdminRoleKey.getRoleId());

if(menuIdList!=null&&menuIdList.size()>0) {

for (TMenuRoleKey tMenuRoleKey : menuIdList) {

TMenu tMenu = menuServiceImpl.selByMId(tMenuRoleKey.getMenuId());

info.addStringPermission(tMenu.getPermission());

}

}

}

}

return info;

1. 登录失败跳转前期提示核心代码

//当用户认证失败时，shiro封装了当前的错误类型

String errorName = (String)request.getAttribute("shiroLoginFailure");

if(IncorrectCredentialsException.class.getName().equals(errorName)) {

System.*out*.println("密码错误");

}else if(UnknownAccountException.class.getName().equals(errorName)) {

System.*out*.println("账号不存在");

}

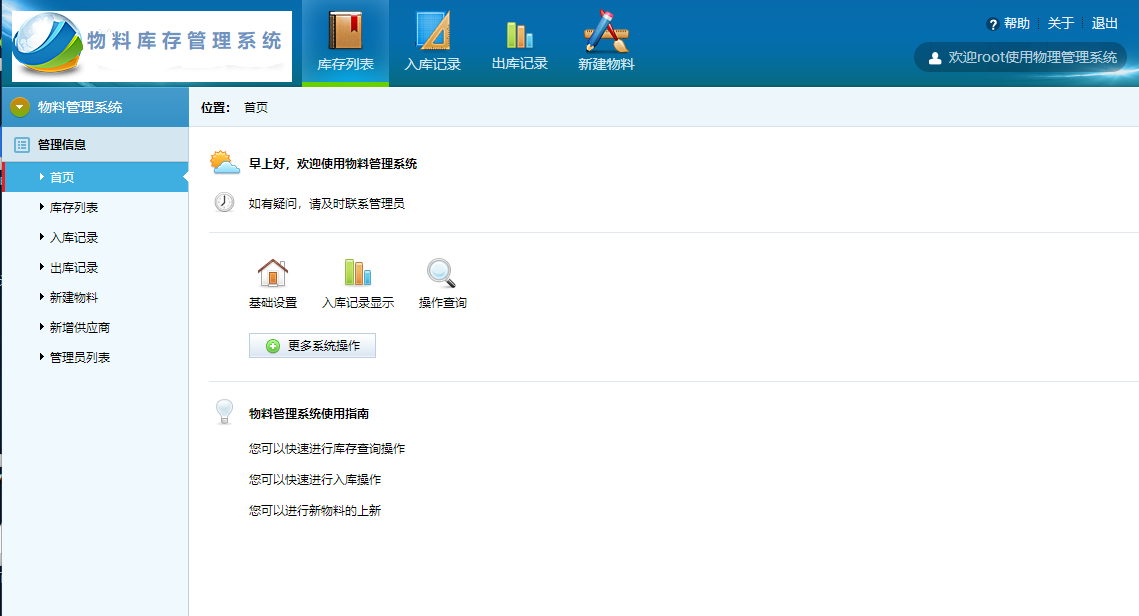
request.setAttribute("error", "用户名或密码错误");

return "login";

5.4.2 首页列表板块

登录成功的用户进入首页列表，根据用户权限显示不同的菜单列表

5.4.2.1 超级管理员对应的菜单列表

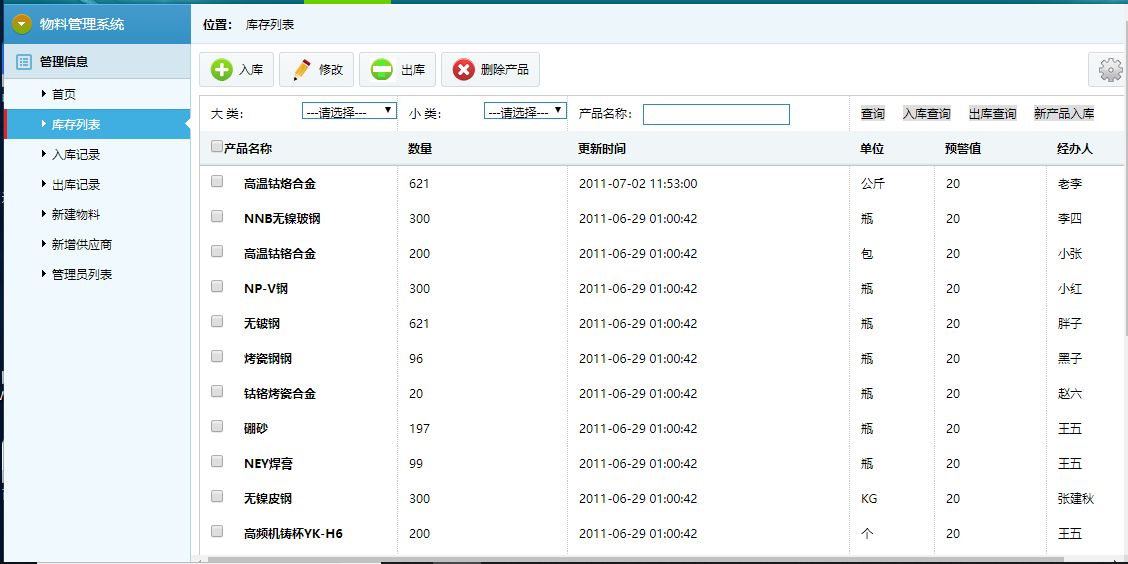


5.4.2.2 普通管理员对应的菜单列表



5.4.3 库存列表板块

5.4.3.1 显示当前从数据库的产品表获取的数据，根据分页显示产品库存信息，根据点击的页进行展示不同页数据



5.4.3.2 可以根据大类分类编号，小类分类编号查询对应的用户信息，根据当前输入的产品信息进行模糊查询；

5.4.3.2.1 根据大类编号查询（级联获取小类信息）

代码：

function loadSmall(obj) {

var bid = obj.value;

//当没有选择大分类时，不请求

if(bid==0) {

return ;

}else {

$.post("/rest/smallType/search",{bid:bid},function(data) {

//直接返回一个list集合。集合存放对应的小分类对象

var options = "";

options += " <option value='0'>---请选择---</option> ";

//返回的json数据是一个list集合数组形式

for(var i=0;i<data.length;i++) {

//拼接数据

options += "<option value="+data[i].id +">"+data[i].name+"</option>"

}

$("#smallType").html(options);

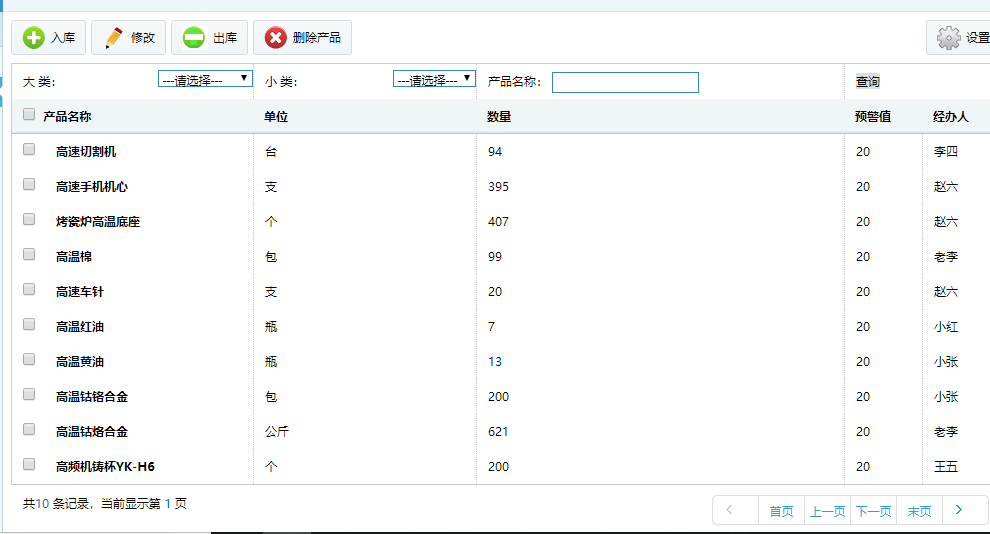
});

}

}

5.4.3.2.2 根据产品名模糊查询

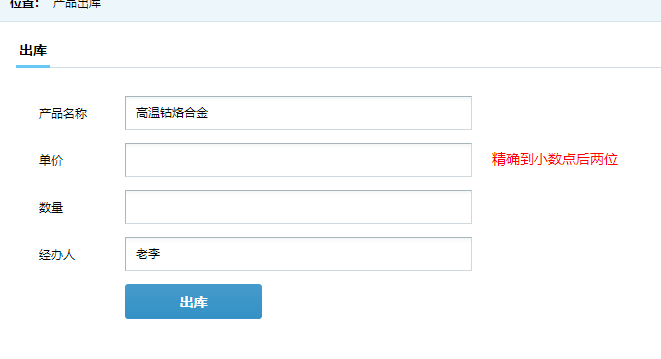
例如：根据“高”进行产品模糊查询

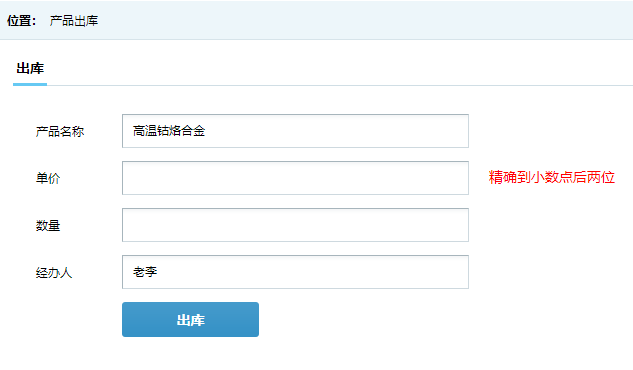


5.4.3.3 对产品进行入库操作



5.4.3.4 对产品进行出库操作，根据点击的产品跳转出库界面，对出库数量与入库数量进行比较；匹配成功允许出库，匹配不成功不允许出库





核心代码：校验出库数量

var id = $("#pid").val();

//进行入库数量和产品数量的比较，发送ajax请求

$.post("/rest/products/query/num",{num:obj.value,id:id},function(data){

if(data.status==200) {

//代表数量符合条件，则允许入库操作

pcount\_msg.innerHTML="";

}else {

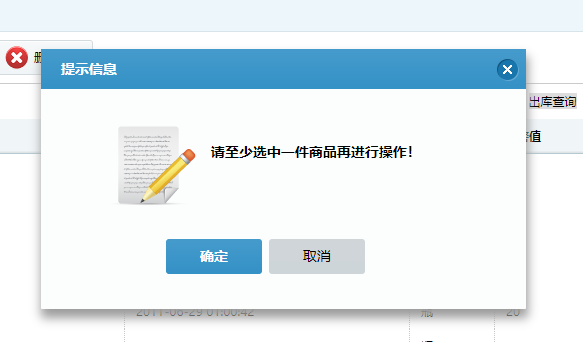
pcount\_msg.innerHTML="当前产品数量为"+data.msg+"，库存不足，请修改出库数量";

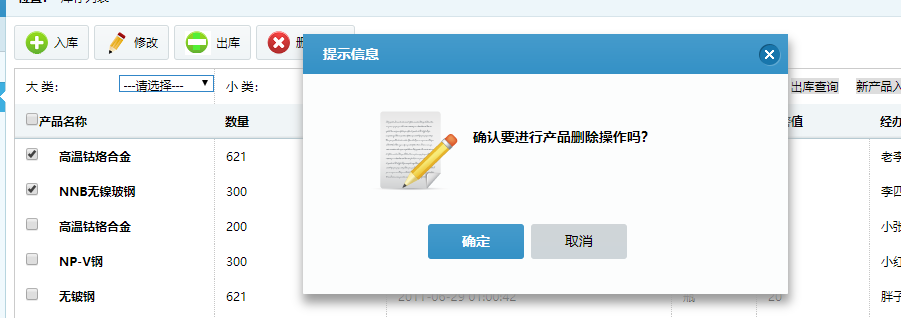
return false;

}

});

5.4.3.5 批量删除当前的产品；当未选中任意商品时，弹出提示框；当选中商品时，弹出提示框“是否确认删除”：确认：直接删除产品，取消：清除提示框





核心代码：

$(（($("#mess\_tip").html("确认要进行产品删除操作吗？");

$(".tip").fadeIn(200);

if($("#sure")[0].click) {

//发送请求进行批量的删除入库记录，物理删除

$.post("/rest/products/delete",{id:ids},function(data) {

//判断是否删除成功

if(data.status==200) {

location.href="/item/kucun\_list";

}else {

$("#mess\_tip").html("删除失败数据回滚，请重新操作");

$(".tip").fadeIn(200);

return false;

}

},"json");

}else if($("#cancel")[0].click){

$(".tip").fadeOut(200);

}

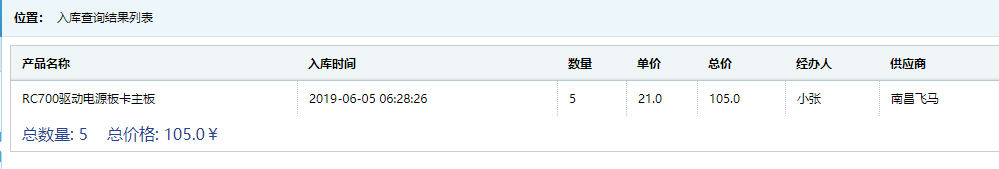
5.4.4 入库记录列表模块

5.4.4.1 显示数据库入库表（t\_input）中产品入库信息



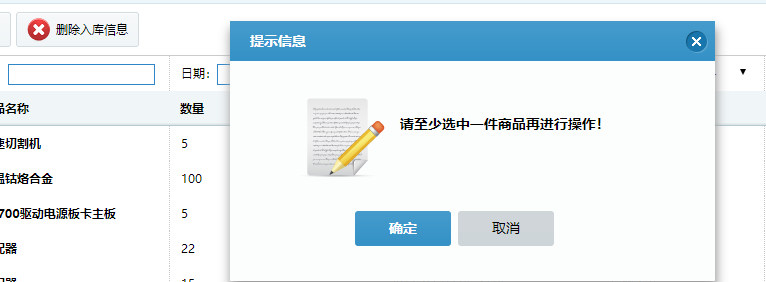
5.4.4.2 根据输入的“产品名称”模糊匹配查询、输入日期、经办人、供应商查询对应入库记录

例如：根据“日期”查询入库记录



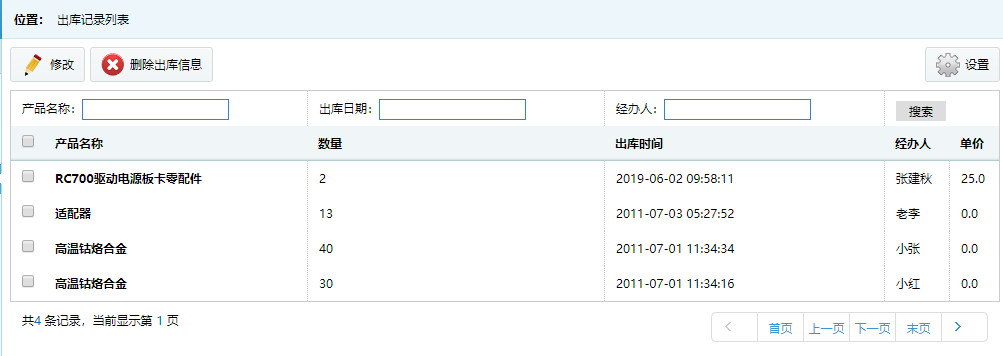
5.4.4.3 批量删除入库记录信息

给出未选中商品的提示信息



5.4.5 出库记录列表模块

5.4.5.1 显示数据库出库表(t\_output)中产品的出库记录



5.4.6 新增物料板快

输入新增产品的信息，对输入的产品信息进行校验（产品名是否唯一校验，其他输入信息是否符合小验格式）、校验都成功的情况下，进行产品的新增操作，插入数据库表中；



核心代码：对新增“产品名”进行唯一校验

$("#pname").blur(function() {

var pro\_name = $(this).val();

if(pro\_name=="") {

$("#pname\_tip").html("产品名称不允许为空，请填写！");

return false;

}else {

//发送ajax请求去判断产品名称是否已经存在

$.post("/rest/products/query/name",{pname:pro\_name},function(data) {

if(data.status==200) {

return true;

}else {

$("#pname\_tip").html("不允许新增同名的产品！");

return false;

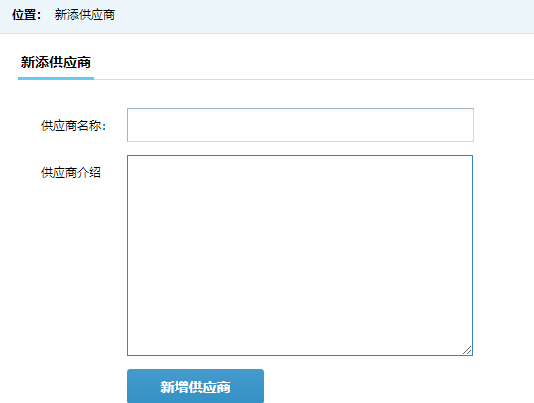
}

});

}

5.4.7 新增供应商板块

输入新增供应商的信息，对输入的信息进行校验（供应商名称是否唯一校验）,校验成功插入数据库表；



核心代码：校验“供应商名称”是否唯一

$("#pname").blur(function() {

var pro\_name = $(this).val();

var reg\_pname = /^[a-zA-Z\u4e00-\u9fa5]+$/;

if(pro\_name=="") {

$("#pname\_tip").html("供应商名称不允许为空，请填写！");

return false;

}else if(!reg\_pname.test(pro\_name)) {

$("#pname\_tip").html("请输入中文字符,或英文字符的名称");

return false;

}else {

//发送ajax请求去判断供应商是否已经存在

$.post("/rest/providers/query/name",{pname:pro\_name},function(data) {

if(data.status==200) {

//代表供应商名称符合规范

return true;

}else {

$("#pname\_tip").html("供应商名称已存在，请重新填写！");

return false;

}

});

}

});

5.4.8 管理员列表板块

5.4.8.1 该模块只对超级管理员开放操作，起始展示管理员列表

5.4.8.2 修改管理员信息操作：

选择管理员信息，跳转信息修改界面如图



对修改后的界面进行校验（密码格式、状态码校验），校验都成功，修改数据库信息，跳转管理员记录列表；

核心代码： 校验的正则表达式

1. 校验密码：var reg\_pwd = /^[a-zA-Z0-9]{4,20}$/;
2. 校验状态：var reg\_state = /^(0|1){1}$/;

5.4.8.3 新增管理员操作：

点击新增按钮，跳转新增界面，如图



输入新增数据，对新增数据进行输入校验（管理员名称唯一、密码格式、状态码格式），校验成功，将数据插入数据库表；

核心代码：校验新增“管理员名称”是否唯一

$.get("/rest/admin/query/name",{userName:obj.value},function(data) {

if(data.status==200) {

return true;

}else {

//表示管理员名称已经存在

name\_msg.innerHTML="用户名已经存在，请更换一个";

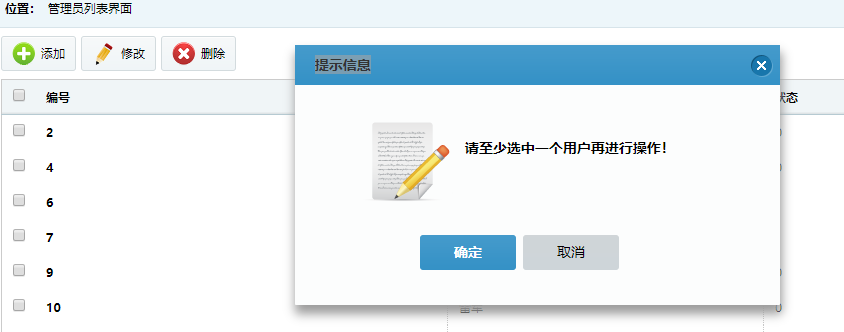
return false;

}

},"json");

5.4.8.4 删除用户操作：

根据选择的数据，进行删除（物理删除，状态修改为0），未选中时，给出提示信息



# 项目系统测试

软件测试是什么？软件测试就是对项目开发过程的产品（编码、文档等）进行差错审查，保证其质量的一种过程。[9]

系统测试毫无疑问是系统开发过程中一个至关重要的阶段，它的重要性体现在它是项目系统正式发布与上线之前保证系统质量和可靠性的最后关口，是整个项目系统开发过程的最后阶段。测试人员在系统成品之初根据需求文档制定测试计划并提交相应的测试用例。系统测试是一个漫长的过程，将伴随着整个系统生命周期，

## 6.1 测试环境

windows10旗舰装机版64位操作系统下进行。

## 6.2 测试方法

### 6.2.1 用户界面测试

根据前端界面的设计样式，测试界面中的基本样式信息是否符合软件开发过程中的美观界面一致化

### 6.2.2 单元测试

对脚本语言中的模块或者封装的方法等最小可验证的单元进行检查以及验证过，单元测试可以尽早发现出问题，但是单元测试具有一定的局限性，测试范围不广泛不全面，无法覆盖所有的执行路径。因此需要进行集成测试。

### 6.2.3 性能测试

性能测试通常包括负载以及压力测试两种。

性能测试通常是用于检测系统在反复使用的情况下是否还能满足需求上要求的性能指标，也能用于检测系统在运行过程中是否会发生内存泄漏之类的问题。

### 6.2.4 回归测试

在系统大概完成之时，要对项目系统依照测试计划完整地测试一遍整个项目并记录和提出bug反馈，在解决相应bug时，便可以进行回归测试了。

### 6.2.5 验收测试

在回归测试之后，当该版本的项目系统准备正式封版之前，进行最后一次整体的至上而下的详细测试，该测试步骤主要是针对用户需求、业务流程进行的正式的测试，以确定系统是否能够满足验收标准。

## 6.3 浏览器测试

（1）测试目的：由于该项目的重心偏向于页面设计和前台交互实现，于是浏览器测试便是测试的重中之重。

（2）测试对象：对目前互联网中的前七款浏览器进行测试

（3）测试结果如表6-1。

表6-1测试结果表

|  |  |  |
| --- | --- | --- |
| 测试对象 | 测试内容 | 测试结果 |
| Google | 系统前后台功能及页面 | 正常 |
| QQ浏览器 | 系统前后台功能及页面 | 正常 |
| UC浏览器 | 系统前后台功能及页面 | 正常 |
| 猎豹安全浏览器 | 系统前后台功能及页面 | 正常 |
| Firefos | 系统前后台功能及页面 | 正常 |
| 搜狗高速浏览器 | 系统前后台功能及页面 | 正常 |
| Internet Explorer11 | 系统前后台功能及页面 | 正常 |

# 结论与展望

在互联网大范围覆盖的加速时代背景下，人们的生活节奏也随之越来越快，这款基于WEB的生活网站不论是设计它之时，还是到现在小出成品，它的初衷与意义都是在符合当代人们生活要求的前提下给人们带来一些创意用心非恶性盈利的交流平台，希望可以为广大网友带来一股不一样的清流，希望该平台可以组建发展为让使用者舒适并且停下脚步记录自己的事迹与他人相互交流学习的场所

## 7.1 结论

从思索自己的项目方向到如今的小有成果，一路过来，这就是个学习的过程。从某种意义上来说，这是第一个从设计理念到需求初稿、项目总体设计、页面设计、后台功能需求，然后逐个去完成再到后台的找bug改bug到最后的总结。一路走来学习到了很多优秀文献，精美的前端页面设计，以及尝试学习和接触不太擅长的后台和数据库。其次，论文的整套流程让我对大学的许多学习课程再次进行了回顾，也让我对不曾涉及的领域有个一个专研并使用的机会。总而言之，这就是个学习的路程，一路下来学习到了很多，自己也扎实了很多。

## 7.2 后期展望

到现在为止我们的项目已经算是有了一个1.0版本了，基本上可以满足用户的需求，但是随着互联网的不断发展以及为了更上这日新月异的时代的前进角度，我们的系统需要不断优化、更新、迭代，不断满足新的需求。这是一个漫长的过程，但是只要我们不忘这款项目的设计初心，我们的项目一定可以越来越好并从中学习到许多。

在未来的时间里我希望以下几个方面可以进行更好的完善与优化:

1. 扩大系统的信息来源，增强自己的数据库，以为用户提供更多更丰富的优质信息和有趣故事。
2. 系统的信息分类更加广泛更加全面，不再仅仅局限于以上几个模块，而是可以涵盖到生活的方方面面。
3. 提高系统的信息搜索效率和准确性，在巨大的信息资源库中，对信息的准确捕捉是一件非常重要的事情，得以迅速给出用户他们想要的反馈信息