需求规约

***仓库管理系统***

东软XX科技有限公司

**(版权所有，翻版必究)**

**更改履历**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 版本 | 更改时间 | 更改人 | 更改章节 | 状态 | 更改描述 |
| 1 | 1.0 | 2023-8-23 | 全体成员 | 全部 | 新建 | 无 |
| 2 | 2.0 | 2023-9-05 | 全体成员 | 全部 | 修改 | 无 |
|  |  |  |  |  |  |  |

状态：新建、增加、修改、删除。

**目录**

[第1章 绪论 4](#_Toc144991648)

[1.1 研究背景 4](#_Toc144991649)

[1.2 研究目的 4](#_Toc144991650)

[1.3 研究内容 4](#_Toc144991651)

[1.4 研究意义 4](#_Toc144991652)

[第2章 系统分析 6](#_Toc144991653)

[2.1 整体应用系统架构 6](#_Toc144991654)

[2.2 功能需求分析 6](#_Toc144991655)

[2.3 非功能需求分析 8](#_Toc144991656)

[第3章 系统设计 10](#_Toc144991657)

[3.1 环境设计 10](#_Toc144991658)

[3.2 用户界面设计 10](#_Toc144991659)

[3.3 系统界面设计 10](#_Toc144991660)

[3.4 应用程序设计 11](#_Toc144991661)

[3.5 数据库设计 11](#_Toc144991662)

[第4章 系统实现 17](#_Toc144991663)

[4.1 系统模块功能实现 17](#_Toc144991664)

[第5章 系统测试 19](#_Toc144991665)

[5.1 测试用例 19](#_Toc144991666)

[5.2 结果分析 20](#_Toc144991667)

[第6章 结论与未来研究方向 22](#_Toc144991668)

[6.1 结论 22](#_Toc144991669)

[6.2 未来研究方向 22](#_Toc144991670)

# 第1章 绪论

## 1.1 研究背景

随着企业业务规模的增长，企业内部存在多个小型仓库，商品种类及数量不断增加，传统的人工管理方式已经无法满足仓储管理的需求。无法及时了解库存情况，经常出现库存不足或过剩的情况，严重影响企业正常运营。为了更好地管理这些分散的仓库并提升整体物流效率，因此，开发一套高效的仓储管理系统势在必行。

## 1.2 研究目的

本研究旨在开发一套适用于企业仓储管理的系统解决方案。该系统基于Spring Boot+Vue架构，实现前后端分离。系统的主要目标是集中管理多个小仓库，优化库存管理，提供数据分析和决策支持，并提供用户友好和易用的界面。通过实现这些目标，企业可以提升仓库管理效率，降低成本，并优化整体供应链运作，也可以精确管理库存，实时监控库存变化情况，并可以生成各类库存报表。使企业能够随时掌握商品库存情况，大大提升仓储管理效率。

## 1.3 研究内容

该系统的主要功能包括:用户管理、角色管理、商品管理、商品分类管理、采购管理、入库管理、出库管理、统计管理以及仓库管理等。旨在提供一个高效、可扩展、易用的仓库管理平台，以满足企业仓储管理的需求。

## 1.4 研究意义

1. 提升仓库管理效率:自动化管理流程、准确记录库存信息，避免人为错误和漏洞。
2. 降低成本：优化库存控制，减少过剩和缺货情况，提供准确的库存数据支持决策制定。
3. 提升数据分析和决策支持能力：该系统可以收集、整理和分析大量的仓储数据，为企业提供全面的数据分析和决策支持，帮助企业做出更加明智和准确的决策。
4. 推动数字化转型：仓储管理系统的研究和应用可以推动企业的数字化转型，从传统的纸质记录和人工操作向数字化、智能化管理方式转变，提升企业的竞争力和创新能力。
5. 促进行业标准化和规范化：通过研究和应用统一的仓储管理系统，可以促进行业内的标准化和规范化，降低行业运作的混乱度，提高整个行业的效率和合作水平。
6. 为其他领域提供借鉴和参考：仓储管理系统的研究和应用涉及到信息技术、业务流程优化、供应链管理等多个领域的结合，对其他领域的管理系统设计和优化提供借鉴和参考价值。

# 第2章 系统分析

## 2.1 整体应用系统架构

本系统采用前后端分离架构，前端使用Vue + ElementUI，后端使用Spring Boot + MyBatis框架；数据库使用MySQL。

1. 前端

* 使用Vue进行模块化开发，提高代码复用率
* 使用ElementUI组件库快速构建界面
* 使用Axios与后端交互，访问RESTful API

1. 后端

* 使用Spring Boot框架，简化配置
* 使用MyBatis进行数据持久层开发
* 使用Spring Security实现权限控制
* 设计RESTful规范的接口
* 使用Redis进行缓存

1. 数据层

* 数据库使用MySQL
* 对数据表进行规范设计

1. 部署

* 后端部署在Tomcat容器中
* 对外提供HTTPS安全访问

## 2.2 功能需求分析

1. 用户管理

* 查询用户:支持条件查询、分页查询
* 添加用户:填写基本信息添加用户
* 修改用户:修改用户信息
* 删除用户:批量/单个删除用户
* 导出数据:Excel导出用户
* 禁用/启用用户:设置用户状态
* 重置密码:重置用户密码
* 分配角色:为用户分配角色

1. 角色管理

* 查询角色:支持条件查询、分页查询
* 添加角色:填写信息添加角色
* 修改角色:修改角色信息
* 删除角色:批量/单个删除角色
* 导出数据:Excel导出角色
* 禁用/启用角色:设置角色状态
* 分配权限:为角色分配菜单、按钮权限

1. 商品管理

* 查询商品:支持条件查询、分页查询
* 添加商品:填写商品信息添加商品
* 修改商品:修改商品信息
* 删除商品:批量/单个删除商品
* 导出数据:Excel导出商品
* 添加采购单:填写采购信息生成采购单
* 生成出库单:新增出库单,填写出库信息
* 商品审核:提交审核,通过后上架销售

1. 商品分类管理

* 添加分类:新增商品分类
* 查询分类:分类列表查询
* 修改分类:编辑商品分类
* 删除分类:批量/单个删除分类

1. 采购管理

* 查询采购单:条件查询采购单
* 生成采购单:采购单审核通过生成
* 修改采购单:编辑采购单信息
* 删除采购单:删除采购单

1. 入库管理

* 查看入库单:查看入库单列表
* 审核入库单:提交入库单审核
* 确认入库:审核通过后确认入库

1. 出库管理

* 查看出库单:出库单列表查询
* 编辑出库单:修改出库单信息
* 审核出库单:出库单提交审核
* 确认出库:审核通过后确认对商品出库

1. 统计管理

* 仓库统计:统计各仓库的库存数量

1. 仓库管理

* 查询仓库:支持条件查询、分页查询
* 添加仓库:填写仓库信息添加仓库信息
* 修改仓库:修改仓库信息
* 删除仓库:删除仓库信息
* 导出数据:Excel导出仓库

## 2.3 非功能需求分析

1. 系统性能

* 系统支持高并发访问,响应时间小于1s
* 高流量下系统服务稳定运行

1. 数据安全

* 对用户访问进行权限控制和认证
* 敏感的业务数据加解密传输
* 数据库备份,防止数据丢失

1. 易用性

* 界面设计简洁易用
* 交互语义化,用户易学易用

1. 可扩展性

* 模块化设计,低耦合度,方便扩展
* 支持分布式部署扩展

# 第3章 系统设计

## 3.1 环境设计

* 操作系统:Windows、Linux
* 数据库:MySQL 5.7+
* 后端:JDK 1.8、Spring Boot、MyBatis
* 前端:Vue 、ElementUI
* 服务器:Tomcat 8.5+
* 开发工具:IDEA、Navicat

## 3.2 用户界面设计

* 前端采用响应式布局，适配不同终端
* 使用ElementUI组件库，样式统一美观
* 首页:数据统计图表，快捷入口
* 导航栏:导航菜单及面包屑
* 左侧菜单:多级分类展示系统菜单
* 内容区:待开发页面和功能

## 3.3 系统界面设计

提供前后端交互的RESTful API接口:

* 用户管理接口:获取用户信息并支持新增、删除、修改、查询等操作
* 角色管理接口:获取角色信息并支持新增、删除、修改、查询等操作
* 商品管理接口:获取商品信息并支持新增、删除、修改、查询等操作
* 商品分类管理接口:获取商品分类信息并支持新增、删除、修改等操作
* 采购管理接口:获取采购信息并支持修改、删除等操作
* 入库管理接口:获取入库信息并支持审核入库单操作
* 出库管理接口:获取出库信息并支持审核出库单操作
* 统计管理接口:获取各个仓库库存数量
* 仓库管理接口:获取各个仓库信息并支持增删改查操作

## 3.4 应用程序设计

* Config层:配置类及相应的方法
* 实现层:负责具体的业务模块逻辑控制
* 实体层:用于封装实体类数据结构
* Service层:调用Mybatis的Mapper接口访问数据库
* 业务层:组合调用Service层接口完成业务
* Utils层:工具类及通用方法
* 日志记录:记录系统关键业务日志

## 3.5 数据库设计

* 用户表（user\_info）:存储系统用户信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **列名** | **中文含义** | **数据类型** | **主外键** | **说明** |
| user\_id | 用户id | Int(11) | 主键 | 自增，非空 |
| user\_code | 用户账号 | Varchar（50） |  | 默认为空 |
| user\_name | 用户名 | Varchar（50） |  | 默认为空 |
| user\_pwd | 用户密码 | Varchar（100） |  | 默认为空 |
| user\_type | 用户类型 | Char(1) |  | 默认为空 |
| user\_state | 用户类型 | Char(1) |  | 默认为空 |
| is\_delete | 删除状态 | Char(1) |  | 默认为空 |
| create\_by | 创建人 | Int(11) |  | 默认为空 |
| create\_time | 创建时间 | datetime |  | 默认为空 |
| update\_by | 修改人 | Int(11) |  | 默认为空 |
| update\_time | 修改时间 | datetime |  | 默认为空 |

* 用户角色表（user\_role）:存储用户角色信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **列名** | **中文含义** | **数据类型** | **主外键** | **说明** |
| user\_role\_id | 用户角色id | Int | 主键 | 自增，非空 |
| role \_id | 角色id | Int |  | 默认为空 |
| user\_id | 用户id | Int |  | 默认为空 |

* 角色表（role）:存储角色信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **列名** | **中文含义** | **数据类型** | **主外键** | **说明** |
| role\_id | 角色id | Int(11) | 主键 | 自增，非空 |
| role \_name | 角色名 | varchar(100) |  | 默认为空 |
| role \_desc | 角色描述 | varchar(300) |  | 默认为空 |
| role \_code | 角色代码 | varchar(100) |  | 默认为空 |
| role\_state | 角色状态 | char(1) |  | 默认为空 |
| create\_by | 创建人 | Int(11) |  | 默认为空 |
| create\_time | 创建时间 | datetime |  | 默认为空 |
| update\_by | 修改人 | Int(11) |  | 默认为空 |
| update\_time | 修改时间 | datetime |  | 默认为空 |

* 角色权限表（role\_auth）:存储角色权限信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **列名** | **中文含义** | **数据类型** | **主外键** | **说明** |
| role\_auth\_id | 角色权限id | Int(11) | 主键 | 非空，自增列 |
| role\_id | 角色id | int(11) |  | 默认为空 |
| auth\_id | 权限id | int(11) |  | 默认为空 |

* 权限表（auth\_info）:存储权限信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **列名** | **中文含义** | **数据类型** | **主外键** | **说明** |
| auth\_id | 权限id | Int(11) | 主键 | 自增列，非空 |
| parent\_id | 父级id | Int(11) |  | 默认为空 |
| auth\_name | 权限名 | Varchar(100) |  | 默认为空 |
| auth\_desc | 权限描述 | Varchar(300) |  | 默认为空 |
| auth\_grade | 权限等级 | Int(11) |  | 默认为空 |
| auth\_type | 权限类型 | Char(1) |  | 默认为空 |
| auth\_url | 权限接口 | Varchar(100) |  | 默认为空 |
| auth\_code | 权限代码 | Varchar(100) |  | 默认为空 |
| auth\_order | 权限顺序 | Int(11) |  | 默认为空 |
| auth\_state | 权限状态 | Char(1) |  | 默认为‘1’ |
| create\_by | 创建人 | Int(11) |  | 默认为空 |
| create\_time | 创建时间 | datetime |  | 默认为空 |
| update\_by | 修改人 | Int(11) |  | 默认为空 |
| update\_time | 修改时间 | datetime |  | 默认为空 |

* 商品表（product）:存储商品信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **列名** | **中文含义** | **数据类型** | **主外键** | **说明** |
| product\_id | 商品id | Int(11) | 主键 | 自增列，非空 |
| store\_id | 仓库id | Int(11) |  | 默认为空 |
| brand\_id | 品牌id | Int(11) |  | 默认为空 |
| product\_name | 商品名 | Varchar(200) |  | 默认为空 |
| product\_num | 商品编号 | Varchar(50) |  | 默认为空 |
| product\_invent | 商品库存 | Int(11) |  | 默认为空 |
| type\_id | 商品类型id | Int(11) |  | 默认为空 |
| supply\_id | 供应商id | Int(11) |  | 默认为空 |
| place\_id | 产地id | Int(11) |  | 默认为空 |
| unit\_id | 单位id | Int(11) |  | 默认为空 |
| introduce | 商品介绍 | longtext |  |  |
| up\_down\_state | 上下架状态 | Char(1) |  | 默认为空 |
| in\_price | 进价 | decimal(10,2) |  | 默认为空 |
| sale\_price | 售价 | decimal(10,2) |  | 默认为空 |
| mem\_price | 会员价 | decimal(10,2) |  | 默认为空 |
| create\_time | 创建时间 | datetime |  | 默认为空 |
| update\_time | 修改时间 | datetime |  | 默认为空 |
| create\_by | 创建人 | int(11) |  | 默认为空 |
| update\_by | 修改人 | int(11) |  | 默认为空 |
| imgs | 图片 | Varchar(500) |  | 默认为空 |
| product\_date | 生产日期 | datetime |  | 默认为空 |
| supp\_date | 保质期 | datetime |  | 默认为空 |

* 商品分类表（product\_type）:存储商品分类信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **列名** | **中文含义** | **数据类型** | **主外键** | **说明** |
| type\_id | 商品类型id | int(11) | 主键 | 自增列，非空 |
| parent\_id | 父级id | int(11) |  | 默认为空 |
| type\_code | 商品类型代码 | varchar(50) |  | 默认为空 |
| type\_name | 商品类型名称 | varchar(100) |  | 默认为空 |
| type\_desc | 商品类型描述 | varchar(500) |  | 默认为空 |

* 品牌表（brand）:存储品牌信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **列名** | **中文含义** | **数据类型** | **主外键** | **说明** |
| brand\_id | 品牌id | int(11) | 主键 | 自增列，非空 |
| brand\_name | 品牌名 | varchar(100) |  | 默认为空 |
| brand\_leter | 品牌字母 | char(1) |  | 默认为空 |
| brand\_desc | 品牌描述 | varchar(1000) |  | 默认为空 |

* 产地表:存储产地信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **列名** | **中文含义** | **数据类型** | **主外键** | **说明** |
| place\_id | 产地id | int(11) | 主键 | 自增列，非空 |
| place\_name | 产地名 | varchar(200) |  | 默认为空 |
| place\_num | 产地编号 | varchar(20) |  | 默认为空 |
| introduce | 产地介绍 | varchar(300) |  | 默认为空 |
| is\_delete | 可用状态 | char(1) |  | 默认为‘0’ |

* 供货商表（supply）:存储供货商信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **列名** | **中文含义** | **数据类型** | **主外键** | **说明** |
| supply\_id | 供货商id | int(11) | 主键 | 自增列，非空 |
| supply\_num | 供货商编号 | varchar(20) |  | 默认为空 |
| supply\_name | 供货商名称 | varchar(100) |  | 默认为空 |
| supply\_introduce | 供货商介绍 | Longtext |  | 默认为空 |
| concat | 供货商名 | varchar(50) |  | 默认为空 |
| phone | 供货商电话 | varchar(20) |  | 默认为空 |
| address | 供货商地址 | varchar(100) |  | 默认为空 |
| is\_delete | 可用状态 | char(1) |  | 默认为‘0’ |

* 规划单位表（unit）:存储单位信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **列名** | **中文含义** | **数据类型** | **主外键** | **说明** |
| unit\_id | 单位id | int(11) | 主键 | 自增列，非空 |
| unit \_name | 单位名称 | varchar(20) |  | 默认为空 |
| unit\_desc | 单位描述 | varchar(20) |  | 默认为空 |

* 采购单表（buy\_list）:存储采购单信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **列名** | **中文含义** | **数据类型** | **主外键** | **说明** |
| buy\_id | 采购id | Int(11) | 主键 | 自增列，非空 |
| product\_id | 商品id | Int(11) |  | 默认为空 |
| store\_id | 仓库id | Int(11) |  | 默认为空 |
| buy\_num | 采购数量 | Int(11) |  | 默认为空 |
| fact\_buy\_num | 实际采购数量 | Int(11) |  | 默认为空 |
| buy\_time | 采购时间 | datetime |  | 默认为空 |
| supply\_id | 供应商id | Int(11) |  | 默认为空 |
| place\_id | 产地id | Int(11) |  | 默认为空 |
| buy\_user | 采购人 | Varchar(20) |  | 默认为空 |
| phone | 采购人电话 | Varchar(20) |  | 默认为空 |
| is\_in | 入库状态 | Char(1) |  | 默认为空 |

* 入库单表（in\_store）:存储入库单信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **列名** | **中文含义** | **数据类型** | **主外键** | **说明** |
| ins\_id | 入库id | Int(11) | 主键 | 自增列，非空 |
| store\_id | 仓库id | Int(11) |  | 默认为空 |
| product\_id | 商品id | Int(11) |  | 默认为空 |
| in\_num | 入库数量 | Int(11) |  | 默认为空 |
| create\_by | 创建人 | Int(11) |  | 默认为空 |
| create\_time | 创建时间 | datetime |  | 默认为空 |
| is\_in | 入库状态 | char(1) |  | 默认为空 |

* 出库单表（out\_store）:存储出库单信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **列名** | **中文含义** | **数据类型** | **主外键** | **说明** |
| outs\_id | 入库id | Int(11) | 主键 | 自增列，非空 |
| product\_id | 商品id | Int(11) |  | 默认为空 |
| store\_id | 仓库id | Int(11) |  | 默认为空 |
| tally\_id | 理货人id | Int(11) |  | 默认为空 |
| out\_price | 出库价格 | decimal(8,2) |  | 默认为空 |
| out\_num | 出库数量 | Int(11) |  | 默认为空 |
| create\_by | 创建人 | Int(11) |  | 默认为空 |
| create\_time | 创建时间 | datetime |  | 默认为空 |
| is\_out | 出库状态 | char(1) |  | 默认为空 |

* 仓库表（store）:存储仓库信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **列名** | **中文含义** | **数据类型** | **主外键** | **说明** |
| store\_id | 仓库id | Int(11) | 主键 | 自增列，非空 |
| store\_name | 仓库名 | varchar(100) |  | 默认为空 |
| store\_num | 仓库编号 | varchar(20) |  | 默认为空 |
| store\_address | 仓库地址 | varchar(100) |  | 默认为空 |
| concat | 联系人 | varchar(50) |  | 默认为空 |
| phone | 电话 | varchar(20) |  | 默认为空 |

# 第4章 系统实现

## 4.1 系统模块功能实现

1. 用户管理

* 完成用户信息的新增、修改以及查询功能
* 实现单个或者批量删除用户
* 支持Excel导出用户
* 完成用户状态的启用禁用切换
* 实现用户密码重置功能

1. 角色管理

* 实现角色信息的新增、修改以及查询功能
* 实现单个或者批量删除角色
* 实现Excel导出角色
* 完成角色状态的启用/禁用
* 实现角色的菜单权限分配

1. 商品管理

* 实现商品信息的新增、修改以及查询功能
* 实现单个或者批量删除商品
* 支持根据分类筛选商品
* 完成商品状态切换、上架/下架
* 实现商品图片的上传及展示
* 完成商品信息的审核功能
* 实现新增采购单，提交采购申请
* 生成出库单,选择商品并填写出库明细
* 支持Excel导出商品

1. 商品分类管理

* 商品分类支持新增、修改以及查询功能
* 支持单个或者批量删除商品分类
* 树形结构显示分类，支持无限级分类

1. 采购管理

* 完成采购单的审批流程
* 支持Excel导出采购单
* 实现采购单的修改与删除

1. 入库管理

* 生成入库单,选择采购单并填写入库明细
* 入库单支持批量导出功能
* 完成入库单的审核与入库操作
* 生成入库报表,更新库存
* 支持Excel导出入库单

1. 出库管理

* 支持出库单的审核与确认出库
* 生成出库报表,更新库存
* 支持出库单的修改与删除
* 支持Excel导出出库

1. 统计管理

* 实现各仓库的库存数量统计

1. 仓库管理

* 实现仓库信息的新增、修改以及查询功能
* 实现单个删除仓库
* 支持Excel导出仓库

# 第5章 系统测试

## 5.1 测试用例

1. 登录功能

|  |  |
| --- | --- |
| 测试用例编号 | TC001 |
| 测试用例名称 | 登录功能验证 |
| 测试目的 | 验证用户登录功能是否正常运行。 |
| 前置条件 | 无 |
| 测试数据 | 用户名：admin  密码：123456 |
| 操作步骤 | 1. 打开系统登录页面。 2. 输入用户名：" admin "。 3. 输入密码："123456"。 4. 点击“登录”按钮。 |
| 预期结果 | 页面跳转至用户首页。  页面显示用户信息和相关功能菜单。 |
| 实际结果 | 页面跳转至用户首页。  页面显示用户信息和相关功能菜单 |
| 验证通过 | 通过 |

1. 添加商品

|  |  |
| --- | --- |
| 测试用例编号 | TC002 |
| 测试用例名称 | 添加商品信息 |
| 测试目的 | 验证添加商品功能是否正常运行。 |
| 前置条件 | 用户已成功登录系统。 |
| 测试数据 | 商品名称：华为Mate  商品数量：100  售价：7399 |
| 操作步骤 | 1. 从菜单中选择 "商品管理" 功能 2. 点击“添加”按钮 3. 输入商品名称："华为Mate "。 4. 输入商品数量："100"。 5. 输入商品售价："7399"。 6. 点击“保存”按钮。 |
| 预期结果 | 弹出提示框，提示商品添加成功。  商品列表中显示新增的商品信息。 |
| 实际结果 | 弹出提示框，提示商品添加成功。  商品列表中显示新增的商品信息。 |
| 验证通过 | 通过 |

1. 修改商品

|  |  |
| --- | --- |
| 测试用例编号 | TC003 |
| 测试用例名称 | 修改商品信息 |
| 测试目的 | 验证修改商品信息功能是否正常运行。 |
| 前置条件 | 用户已成功登录系统且存在可修改的商品。 |
| 测试数据 | 商品名称：华为Mate  新售价：7099  新会员价：6899 |
| 操作步骤 | 1. 从菜单中选择 "商品管理" 功能 2. 在商品列表中找到商品"华为Mate"。 3. 点击商品右侧的“修改”按钮 4. 输入新售价：" 7099"。 5. 输入新会员价："6899"。 6. 点击“保存”按钮。 |
| 预期结果 | 弹出提示框，提示商品价格修改成功。  商品列表中对应商品的价格更新为新价格。 |
| 实际结果 | 弹出提示框，提示商品价格修改成功。  商品列表中对应商品的价格更新为新价格。 |
| 验证通过 | 通过 |

## 5.2 结果分析

1. 单元测试

* 使用JUnit框架编写单元测试用例
* 编写UserService、RoleService等接口的单元测试
* 编写数据访问层的单元测试
* 单元测试覆盖率达到80%以上

1. 功能测试

* 登录流程:多种角色验证登录功能
* 用户管理:测试增删改查和复杂查询功能
* 角色管理:测试角色的增删改查和权限分配
* 商品管理:测试商品管理各功能点
* 分类管理:验证分类树形结构显示
* 采购管理:测试采购单全流程功能
* 入库管理:验证入库单生成及入库操作
* 出库管理:测试出库单生成及确认出库
* 统计管理:验证各类统计图表
* 仓库管理:验证增删改查和复杂查询功能

1. 性能测试

* 使用JMeter压力测试所有接口
* 加压到500并发请求,响应时间小于1s

1. 数据库测试

* 使用MySQL Tuner等工具分析数据库性能
* 压力测试数据库,TPS达到100+
* 所有SQL执行时间小于1s

# 第6章 结论与未来研究方向

## 6.1 结论

通过本系统的设计与实现，达成了打造一套适用于企业仓储管理的解决方案的目标。

系统实现了用户管理、角色管理、商品管理、入库出库管理等核心功能，能够满足仓库信息化管理的需求。系统交互流畅、响应快速,数据处理能力强,可以适应高并发场景。

经过测试，系统功能运转正常,各项指标达到预期。通过本系统的应用，可以有效优化企业仓储管理，提升商品管理效率。

## 6.2 未来研究方向

1. 移动端支持

开发Android和iOS移动应用，扩大系统应用场景。

1. RFID技术运用

结合射频identification技术，实现仓库自动化管理。

1. 业务智能化

使用大数据、人工智能技术，Enable 智能化决策。

1. 功能拓展

根据客户需求，持续扩展系统功能模块。

1. 第三方集成

与企业其他系统集成，提供更强大的支撑。

通过不断更新迭代，我们将使系统在可靠性、可用性、扩展性方面更上一层楼。