

论文检测报告

报告编号：1BBE6B64F5F449B5B11EC7CAAB0D06CA [真伪查询](#)

送检文档：2014263040105-程兆祥-基于JavaWeb的小型超市管理系统

论文作者：程兆祥

文档字数：25736

检测时间：2018-05-17 22:49:25

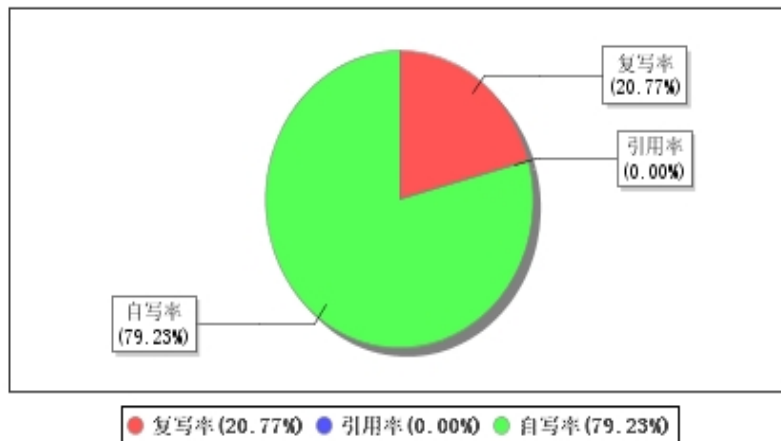
检测范围：互联网，中文期刊库（涵盖中国期刊论文网络数据库、中文科技期刊数据库、中文重要学术期刊库、中国重要社科期刊库、中国重要文科期刊库、中国中文报刊报纸数据库等），学位论文库（涵盖中国学位论文数据库、中国优秀硕博论文数据库、部分高校特色论文库、重要外文期刊数据库如Emerald、HeinOnline、JSTOR等），共享资源库，参考文献库。

一、检测结果：

总相似比：20.77% [即复写率与引用率之和]

检测指标：自写率 79.23% 复写率 20.77% 引用率 0.00%

其他指标：页数 (23) 图片数 (23) 表格数 (17) 引用句子数 (0) 公式 (0)



二、相似文献汇总：

序号	标题	文献来源	作者	出处	发表时间
1	网上书店管理信息系统.doc-max上传文档投稿赚钱-文档C2C交易模式...	互联网		互联网	0
2	...多对多,用户新增,只增加中间表对应角色关系,不增加角色role-...	互联网		互联网	0
3	mvc软件开发模式,ssh_云朵网络_www.02521.com	互联网		互联网	
4	Volley源码分析之自定义GsonRequest(带header,cookie,..._CSDN博客	互联网		互联网	0
5	...@RequestMapping注解后,在jsp页面的form的action属..._百度知道	互联网		互联网	0

6	Jsp和Servlet有什么区别?-知乎	互联网		互联网	0
7	...springmvc使用注解方式,为什么实体bean需要在程序..._CSDN论坛	互联网		互联网	0
8	超市，城市的供给站——超市与城市形象传播策略研究	学术期刊	无	广告大观：综合版	2013
9	jsp与servlet的区别与联系-gsyabc-博客园	互联网		互联网	0
10	java权限设计大纲-CSDN博客	互联网		互联网	0
11	基于单片机的电子密码锁控制电路设计 - 期刊论文 - 道客巴巴	互联网		互联网	
12	...2017值得买吗?2017款ThinkPadX1Carbon笔记本全面评..._脚本之家	互联网		互联网	0
13	大学毕业设计论文 - 实施方案 - 道客巴巴	互联网		互联网	
14	C++开发人脸性别识别教程(12)——添加性别识别功能-CSDN博客	互联网		互联网	0
15	Ihavenoideawhywehavetogotothesupermarkettonight.YouknowIhave...	互联网		互联网	0
16	基于WEB的建筑施工技术知识管理系统研究与实现	学位论文	尹汉东	硕博学位论文	2005
17	示例解读Java的跨平台原理-CSDN博客	互联网		互联网	0
18	网上商品订购系统论文 - 讲义教程 - 道客巴巴	互联网		互联网	
19	基于JSP的电子商城网站的设计与实现论文 - 讲义教程 - 道客巴巴	互联网		互联网	
20	超市_互动百科	互联网		互联网	0
21	基于MVC设计模式的舰船信息管理系统的设计与实现	学术期刊	李振华	软件	2012
22	基于Servlet的线程级安全的分析与实现	学术期刊	鲍考明 程耕国 陈新国	计算机与现代化	2004
23	管理系统设计与开发技术综述	学术期刊	金行;吕冬梅	民营科技	2017
24	设有员工实体Employee(员工号,姓名,性别,年龄,电话,家..._百度文库	互联网		互联网	0
25	动态WEB技术研究——基于WEB的办公自动化系统的研究与实现	学位论文	熊姣姣	硕博学位论文	2005
26	C#ABP-创建自己的模块	互联网		互联网	0
27	基于“角色-模块-页面-功能点”的分级细粒度权限访问控制方法 - ...	互联网		互联网	
28	测试用例实例 - Test Dancer的个人空间 - 51Testing软件测试网 51...	互联网		互联网	
29	中美大学生论说文语篇连贯和衔接性对比分析	学位论文	皇甫伟	硕博学位论文	2004
30	基于J2EE的高校后勤采供管理系统的研究和实现	互联网		互联网	0
31	【Toeathealthily,moreandmorepeopleareinterestedinorganic(有机...]	互联网		互联网	0
32	MVC设计模式—商城项目的详细介绍-CSDN博客	互联网		互联网	0
33	服务营销在我国超市零售业中的应用研究	学位论文	宋安玲	硕博学位论文	2008

34	商业零售企业员工工作满意度与积极应激的关系研究	学术期 刊	张淑敏 王 琼	经营与管理	2012
35	类型3JDBC驱动程序原理及实现	学术期 刊	程学先 龚 晓明 王富 强	武汉理工大 学学报：交 通科学与工 程版	2004
36	基于JSP的发印管理系统设计与实现	学术期 刊	窦立君;张 镜;赵飞	电脑编程技 巧与维护	2016
37	201210年石家庄振头沿街商业招商营销策略92P_百度文库	互联网		互联网	0
38	阿里云服务器centos7.4操作系统Apache配置https服务-CSDN博客	互联网		互联网	0
39	天津大学成人高等教育专科生毕业设计(论文).doc	互联网		互联网	0
40	网络分销渠道的构成与结构分析	学术期 刊	杨赋立	商业时代	2004
41	Spring中IOC配置xml实现和IOC注解实现-CSDN博客	互联网		互联网	0
42	基于JSP技术的婚庆公司管理系统的开发开题报告 - 豆丁网	互联网		互联网	

三、全文相似详情：（红色字体为相似片段、浅蓝色字体为引用片段、深蓝色字体为可能遗漏的但被系统识别到与参考文献列表对应的引用片段、黑色字体为自写片段）

分类号 论文选题类型 非师范型应用研究 U D C 编号 本科毕业论文（设计） 题 目 基于SSM框架的超市 管理系统设计与实现 学 院 计算机学院 专 业 软件工程 年 级 201401 学生姓名 程兆祥 学 号 2014263040105 指导教师 关玉蓉 二 0 1 8 年 5 月 黄冈师范学院学位论文原创性声明 本人郑重声明：所呈交的学位论文是本人在导师指导下独立进行研究工作所取得的研究成果。除了文中特别加以标注引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写的成果作品。本人完全意识到本声明的法律后果由本人承担。 学位论文作者签名： 日期： 年 月 日 学位论文版权使用授权书 本学位论文作者完全了解学校有关保障、使用学位论文的规定，同意学校保留并向有关学位论文管理部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权省级优秀学士学位论文评选机构将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。 本学位论文属于 1、保密 ，在_____年解密后适用本授权书。 2、不保密 。 （请在以上相应方框内打“ ”） 学位论文作者签名： 日期： 年 月 日 导师签名： 日期： 年 月 日

目录

内容摘要 PAGeref_Toc22953 5

关键词：5

Abstract6

Keywords7

一、绪论1

(一) 导论1

1. 前言1

2. 选题背景1

3. 课题内容2

4. 可行性分析3

(二) 系统概述3

1. 系统描述：3

2. 系统开发环境4

(三) 本章小结4

二、系统需求分析与总体架构5

(一) 需求分析5

1. 需求描述5

2. 模块需求分析5

(二) 系统架构和运行环境6

1. MVC架构介绍6

2. 系统运行环境8

3. 系统组织结构图8

(三) 数据库设计8

1. 数据库表结构8

2. 数据库结构图：13

(四) 本章小结13

三、系统运行效果展示14

(一) 系统相关14

1. 用户登录14

2. 修改密码14

(二) 商品管理15

1. 商品列表15

2. 添加商品16

3. 商品分类16

4. 商品回收站17

5. 订单列表17

(三) 员工管理18

1. 员工列表18

2. 新增员工18

(四) 会员管理19

1. 会员列表19

2. 添加会员20

(五) 权限管理20

1. 管理员列表20

2. 添加管理员21

3. 角色列表21

4. 添加角色22

(六) 统计图表22

1. 商品统计22

(七) 本章小结23

四、后端代码示例24

(一) MemberAction.java24

(二) MemberService.java25

(三) MemberServiceImpl.java26

(四) MemberDao.java27

(五) Member.java27

(六) 本章小结29

五、系统测试30

(一) 测试说明30

(二) 测试用例30

(三) 部分测试结果示例33

1. 商品详细信息33

2. 删除商品34

3. 添加商品34

结束语35

参考文献36

致 谢36

内容摘要：随着社会的发展和进步，目前越来越多的超市出现在人们生活中，这些超市为人们提供了各式各样的生活必需品，去超市购物已经成为人们生活中必不可少的一环。随着超市的不断发展，商品数量增多，一个超市需要处理大量的关于商品，订单，库存等的信息，还需要对各种实时的数据进行统计分析已选择更好的管理模式。

本系统是基于WEB开发，参考现今开放的一些超市管理系统，进行一系列的系統结构分析，设计后制作出来，专为各种中小型超市服务的超市管理系统。系统已当今最流行的J2EE为基础，Spring，SpringMVC和Mybatis三大框架为工具进行开发。使用到了MySQL数据库作为后端数据存储。系统严格遵循MVC设计模式，分为表现层（View），业务层（Service）和数据访问层（DAO）。采用B/S模式，用户不需要对系统进行下载安装，只要一台安装了浏览器的电脑就可以进行访问和使用，相对于C/S模式，不需要进行软件的更新和升级，十分简洁高效，并具有很高的可维护性和可扩展性。

结合以上的软件和相关功能与技术要求，本系统的前台页面具有用户登录、商品列表、添加商品、商品分类、商品回收站、订单列表、员工列表、添加员工、会员列表、添加会员、折扣设置、管理员列表、添加管理员、角色列表、添加角色、商品统计、会员统计、员工统计、订单统计等功能，后台能够完成商品增加、商品信息修改、员工信息修改、订单列表查询、权限管理、实时统计、修改信息等一系列的功能。通过前端页面、后端程序和数据库之间的数据访问和交互，构成了本设计的系统基本架构。

本论文主要目的是介绍课题的开发背景，需要完成的功能以及开发过程。给出了系统的各项分析报告，罗列了系统开发的难点技术以及相关的解决方案。本系统可以完成日常超市管理中大部分所需求的功能。

关键词：超市管理，Spring，SpringMVC，Mybatis，B/S，J2EE，MySQL，Linux服务器

Abstract：Along with society's development and progress, currently, more and more supermarkets appear in people's life, the supermarket provides people with a wide variety of life necessities, go to the supermarket shopping has become a indispensable part of people's life. With the continuous development of the supermarket, the quantity increased, a supermarket to deal with a lot of goods, orders, inventory and other information, also need to all kinds of real-time data statistical analysis has chosen a better management model.

This system is based on WEB development, reference current open some of the supermarket management system, a series of system structure analysis, made after the design, specially designed for all kinds of small and medium-sized supermarkets service supermarket management system. The system is now the most popular J2EE based, Spring, SpringMVC and Mybatis three frameworks for tool development. Use the MySQL database as the back-end data store. The system strictly follows the MVC design pattern, which is divided into presentation layer (View), business layer (Service) and data access layer (DAO). Using B/S

model, users don't need to download and install the system, as long as a computer installed the browser can access and use, compared with the C/S mode, don't need to do a software update and upgrade, very concise and efficient, and has the very high maintainability and extensibility.

In combination with software and related functions and technical requirements of the above, the front page of this system has the user login, product list, add commodities, commodity classification, the recycle bin, order list, the staff list, add the staff, member list, add, discount is set, the administrator list, add the administrator, role list, adding roles, commodity statistics, member statistics, statistics, order statistics, and other functions, the background can finish goods increase, commodity information modify query, modify staff information, order list, rights management, a series of functions such as real-time statistics, modify the information. Through the data access and interaction between the front-end page, back-end program and database, the basic architecture of the system is formed.

The main purpose of this paper is to introduce the development background of the project, the functions to be completed and the development process. The paper gives the analysis report of the system, lists the difficult technology of system development and the related solution. The system can fulfill most of the functions of daily supermarket management.

Keywords: SuperMarket managerSystem , Spring , SpringMVC , Mybatis , B/S, J2EE, MySQL , Linux

一、绪论

(一) 导论

1. 前言

在计算机技术不断发展的今天, IT产业已经渗透到了各个行业当中, 用于对数据进行处理和分析。随着各种超市的不断发展, 每天需要处理的商品数量越来越多, 靠着传统的人工方式去进行管理和分类已经不太现实了。目前很多的大型超市都开始使用管理系统进行商品管理, 而一些中小型超市, 他们并没有必要去使用大型的超市管理系统, 但对于信息的处理却也是存在迫切的需要。小型超市管理系统面向的就是这样的群体, 为他们提供一个高效的, 可行的商品管理方案。也符合当今时代的发展要求。

2. 选题背景

互联网发展至今, 技术已经相当的成熟, 随着网络和科技的发展, 各种各样的管理系统应运而生, 并且被各种大小企业所使用, 而超市的管理目前也开始由管理系统慢慢代替人工管理。

超市即超级市场, 一般是指商品开放陈列, 顾客自我服务, 货款一次结算, 以经营生鲜食品、日杂用品为主的商店。超级市场是一种消费者自我服务、敞开式的自选售货的零售企业。它是二次大战后发展起来, 最先在欧美兴起, 现在在欧美十几个国家中已有超级市场20万个。在我国, 超市形成在20世纪90年代初期, 现今已成为我国零售业的一种重要形态, 为普通老百姓提供了极大的便利。在超市的不断发展过程中, 商品越来越多, 其经营模式也越来越复杂, 早期的一些管理方式在这种情况下显得很吃力, 已经不能满足现今零售行业的发展了。

超市的形态具有很多的优点, 这也是它发展如此迅速的重要原因, 然而, 它依旧存在着很多传统零售行业共同的缺点: 商品种类繁多, 不能进行合理的分类和规划, 盘点效率很低。日常所需处理的数据量日益增大, 让人工进行处理变得非常困难, 因此, 目前的中小型超市迫切的需要引进新的技术来对各种数据进行统一和集中的管理, 并根据统计数据对今后的发展进行规划。

3. 课题内容

经过一系列的研究和分析, 超市管理系统决定使用Spring+SpringMVC+Mybatis框架开发, 考虑到小型超市的经济能力, 使用Oracle的免费开源数据库MySQL来进行数据的存储。项目开发环境使用IntelliJ IDEA集成开发环境+JDK1.8, 使用Apache Tomcat8.0作为WEB引用服务器。并使用到了Maven作为项目管理工具, Git作为系统管理工具, GitHub作为远程仓库。主要研究以下几个方面:

参考显示超市的销售流程和管理方式, 分析其优势和不足, 考虑相对应的超市管理系统替代手段和实现方式。

使用目前被企业大量采用的Spring+SpringMVC+Mybatis三大开源框架进行系统整合, 使用MySQL作为后端数据存储。

严格遵循软件工程开发流程和MVC设计架构, 对系统进行详细的需求分析、概要设计、软件测试等等。

对系统的安全性有一定要求, 防止用户恶意输入, SQL注入攻击等等, 并对于用户输入的各项数据都进行验证。

本课题最终将实现以下的目标:

功能完善, 可以满足超市的需求

界面美观易操作, 用户体验良好

具有较强的交互性
便于维护和优化，有较强的可移植性
系统的部署简单方便，运行容易
没有会影响业务流程的bug
可以适配不同的操作平台
可以适配多种数据库

4. 可行性分析

超市管理系统项目可行性分析通过对项目的市场需求、建设规模、开发流程、环境选择、系统架构等方面的研究，从技术、经济、工程等角度对项目进行调查研究和比较分析，为项目决策提供公正、可靠、科学的可行性分析报告。具体而言，本报告体现如下：

经济可行性：只需要一台PC就可以进行开发，不需要其他设备，并且使用的都是开源免费软件，不需要进行额外的投入。

技术可行性：JAVA发展至今已经20余年，J2EE已经成为使用最多的服务端开发语言，而Spring，SpringMVC和Mybatis也是现在最流行的开发框架，被各大公司所使用。

运行可行性：本系统使用JDK1.8开发，Tomcat作为应用服务器，只需要安装了jdk和tomcat的服务器都可运行，并且由于Java的跨平台特性，可以运行在多个操作系统上。

法律可行性：所有技术资料都为合法，开发过程中不存在知识产权问题，未抄袭任何网站，不存在侵犯版权问题，符合国家法律和软件法律法规。

（二）系统概述

1. 系统描述

SMS小型超市管理系统是一个基于J2EE的B/S架构的在线管理系统，使用软件工程的开发流程和面向对象的程序设计思想进行开发，拥有商品管理，员工管理，订单管理，权限管理，统计图表等多个应用模块，实现对超市的商品统一分类，订单的查询，员工的管理，各种数据的展示等功能。

系统开发环境

硬件：

联想 ThinkPad X1 Carbon 笔记本电脑

操作系统Windows 10 专业版 64位 (DirectX 12)

处理器英特尔 第三代酷睿 i7-3667U

内存 8GB

显卡 英特尔 HD Graphics 4000

软件：

JAVA开发版本 JDK1.8

Tomcat版本 8.0

IDE IntelliJ Idea 2018.1.2

数据库 MySQL 5.0.22

浏览器 Chrome 66.0.3359.139

远程服务器 阿里云 CentOS 7.4

SSH连接工具 XShell 6 , Xftp 6

项目管理工具 Maven , Git , GitHub

其他工具 NaviCat 9.1.8 , PostMan 6.0.10

（三）本章小结

本章主要介绍了选题的背景和意义，阐述了对小型超市管理系统的市场调研结果，对系统的需求和可行性进行了系统的分析和说明。对于小型超市管理系统的概况做了论述，并合理分析与设计了系统的架构和开发过程中需要使用到的工具，这对于系统的开发是非常重要的，一个好的系统设计可以让程序在开发过程中事半功倍。

二、系统需求分析与总体架构

（一）需求分析

1. 需求描述

通过对显示生活中的超市进行调研，提出在线超市管理系统的需求，对要实现的功能进行大体的描述，包括对

商品信息, 商品分类, 员工信息, 订单信息, 会员信息, 权限信息的管理。要求实现对各项数据的增删改查能力, 并要求以统计图表的方式对信息进行展示, 方便用户查看和分析。

2. 模块需求分析

(1) 商品管理: 主要负责对商品信息的展示, 增删改查等等, 支持条件查询和模糊匹配, 对新增的商品信息要进行表单验证。点击商品名称弹窗显示商品的详细信息, 并可以直接在弹窗中对商品进行修改操作。对于商品的添加进行严格的表单验证, 已经存在的商品名称无法使用。商品分页进行显示, 每页15条数据。删除商品不会直接对数据库中的数据进行删除, 而是放入商品回收站。

(2) 商品分类: 系统商品类别分为大类和细类, 大类即细类的父类, 细类则是商品的具体类别。分为三个表格, 进入页面后显示所有的大类, 点击大类表格后显示相应的细类表格, 点击细类表格后显示相应的商品信息表格。

(3) 商品回收站: 已经被删除的商品可以在商品回收站之中进行恢复。回收站支持条件查询和模糊匹配, 分页进行显示, 每页15条数据。

(4) 订单列表: 显示所有的订单信息, 要包含订单的商品名称, 下单人名称等信息, 可以对商品名称, 下单人名称和订单状态进行查询, 支持模糊匹配。分页进行显示, 每页15条数据。

(5) 员工管理: 对员工的信息进行增删改查的管理, 可以对员工的姓名, 年龄, 性别, 工种等进行条件查询。工种分为店长、经理、促销员、收银员、收获员、理货员、保洁, 其中店长的信息不可修改。支持模糊匹配。新增员工时要进行表单验证, 并且相同名称的会员无法新增。分页进行显示, 每页15条数据。

(6) 会员管理: 对会员的信息进行增删改查的管理, 可以对员工的姓名, 电话和等级进行条件查询, 点击员工姓名弹窗显示员工的详细信息, 并可以直接在弹窗中对商品进行修改操作。对于员工的添加进行严格的表单验证, 已经存在的员工名称无法使用。分页进行显示, 每页15条数据。可以对不同会员等级的折扣进行设置, 不同的折扣会影响会员购物时的价格。

(7) 权限管理: 权限系统包含管理员, 角色和权限, 一个管理员可以有多个角色, 一个角色可以有多个权限, 不同的权限影响不同的系统菜单, 没有权限的管理员无法进入相对应的系统菜单。可以对管理员和角色进行增删改查, 支持模糊匹配, 角色界面要显示该角色的所有权限, 管理员界面要显示该管理员的所有角色和所有权限, 分页进行显示, 每页12条数据。新增数据时进行验证, 相同名称的管理员和角色无法添加。

(8) 统计图表: 包含商品统计, 会员统计, 员工统计和订单统计, 要求已柱状图和饼图的方式对统计信息进行显示。并显示每个信息对应的数据库中数据条数。

(9) 系统设置: 包含修改密码和退出登录, 修改密码是要求两次输入密码一致才可修改, 修改完密码后要求重新进行登录。点击退出登录则会直接退出系统到登录界面。

(二) 系统架构和运行环境

1. MVC架构介绍

MVC模式(Model-View-Controller)是软件工程中的一种软件架构模式, 把软件系统分为三个基本部分: 模型(Model)、视图(View)和控制器(Controller)。最早由Trygve Reenskaug在1978年提出, 并应用在Smalltalk系统中。

图 1.2.1 MVC架构模式图

Model: 封装领域数据及逻辑。用于管理应用程序域的行为和数据, 并响应为获取其状态信息(通常来自视图)而发出的请求, 还会响应更改状态的指令(通常来自控制器)。

View: 查询领域数据并展现给用户。用于管理信息的显示。

Controller: 截获用户请求并改变领域数据。用于解释用户的鼠标和键盘输入, 以通知模型和/或视图进行相应的更改。

系统的MVC处理流程如下:

Web浏览器发送HTTP请求到服务端, 被Controller(Servlet)获取并进行处理(例如参数解析、请求转发)

(1) Controller(Servlet)调用核心业务逻辑——Model部分, 获得结果

(2) Controller(Servlet)将逻辑处理结果交给View(JSP), 动态输出HTML

(3) 动态生成的HTML内容返回到浏览器显示

MVC模式在Web开发中的好处是非常明显的, 它规避了JSP与Servlet各自的短板, Servlet只负责业务逻辑而不会通过out.append()动态生成HTML代码; JSP中也不会充斥着大量的业务代码。这大大提高了代码的可读性和可维护性。

2. 系统运行环境

本系统采用Java语言开发，具有“一次编译，处处执行”的特性，理论上来说，在任何一台安装了jdk和Tomcat的服务器上都可运行，不论是liunx，unix还是windows操作系统。目前，本系统采用的是阿里云的服务器，操作系统为CentOS7.4，JDK版本为1.8，Tomcat版本为8.0。

系统组织结构图

图 1.2.2 系统组织结构图

数据库设计

1. 数据库表结构

员工信息表employee：本表保存所有员工的信息，包括员工姓名、年龄、性别、薪水、电话等基本信息，empid是员工信息表的主键。其中性别使用int类型来存储，1代表男性，2代表女性。工种也是使用int类型存储，数字和工种对应关系：1店长，2经理，3促销员，4收银员，5收货员，6理货员，7保洁。

表1.3.1 员工信息表

字段名

数据类型

描述

empid

int

Id

ename

varchar

姓名

esex

int

性别

eage

int

年龄

etime

datetime

入职时间

etel

long

电话

epro

int

工种

email

varchar

邮箱

esal

double

薪水

商品信息表commodity：本表负责保存所有商品的主要信息，包括商品的名称、分类、保质期、价格等等。其中comid为主键，finid为商品分类表的外键，对应该商品所在的细类。Cstatus代表商品的状态，0代表已进入回收站的商品，1代表正常的商品。

表1.3.2 商品信息表

字段名

数据类型

描述

comid
int
ld
cname
varhar
名称
finid
int
分类
month
int
保质期
cweight
int
重量
cprice
double
价格
cproder
varchar
生产商
cprodate
datetime
生产日期
cstock
int
库存
cdesc
varchar
描述
cstatus
int
状态

会员信息表member：本表主要保存会员的信息，包括会员的名称、电话、余额等等。Memid为表主键，mlevel代表该会员的会员等级。

表1.3.3 会员信息表

字段名
数据类型
描述
memid
int
ld
mname
varchar
名称
mtel
long
电话

mlevel
int
等级
mdate
datetime
时间
mcount
double
消费总额
mrem
double
余额

商品大类表class：存储商品大类的信息，claid为表主键，cname是大类的名称，ccode为大类的编号，主要用来负责在系统中对类别进行标识，方便管理。

表1.3.4 商品大类表

字段名
数据类型
描述
claid
int
Id
cname
varchar
名称
ccode
varchar
编号

商品细类表fine：存储商品细类的信息，finid为表主键，claid是商品大类外键，对应商品大类表的主键。fcode为大类的编号，主要用来负责在系统中对类别进行标识，方便管理。

表1.3.5 商品细类表

字段名
数据类型
描述
finid
int
Id
fname
varchar
名称
fcode
varchar
编号
claid
int

大类id

订单信息表form：存储所有订单的信息，包括订单商品名称，会员名称，下单数量和总价等等。

表1.3.6 订单信息表

字段名

数据类型

描述

forid

int

ld

comid

int

商品id

finid

int

分类id

cprodate

datetime

生产日期

month

int

保质期

cprice

double

价格

ccount

int

数量

memid

int

会员id

mname

varchar

会员名称

createtime

datetime

创建时间

fstatus

int

状态

管理员信息表admin：存储所有管理员账户的信息，aid是表主键，createtime代表管理员的创建时间，在管理员被创建的时候会自动被写入数据库，lastlogintime代表管理员上次登录时间，每次管理员进行登录的时候系统都会更新数据库中的该字段数据。

表1.3.7 管理员信息表

字段名

数据类型

描述

aid

int

ld

aname

varchar

名称

apassword

varchar

密码

createime

datetime

创建时间

lastlogintime

datetime

上次登录时间

角色信息表role：存储所欲角色的信息，rid是表主键，createime是角色创建的时间，在角色被创建的时候系统会自动记录下当前时间并写入数据库。

表1.3.8 角色信息表

字段名

数据类型

描述

rid

int

Id

rname

varchar

名称

createime

datetime

创建时间

权限信息表power：代表所有的权限，pid是表主键，pname是权限的名称，purl则是该权限对应的url地址，只有管理员拥有的权限和请求的权限一致时，管理员才可以访问对应的页面。

表1.3.9 权限信息表

字段名

数据类型

描述

pid

int

Id

pname

varchar

名称

purl

varchar

地址

角色权限中间表rolepower：角色和权限对应的中间表，rid是角色表主键对应的外键，pid是权限表主键对应的外键。

表1.3.10 角色权限中间表

字段名

数据类型

描述

rid

int

角色Id

pid
int
权限Id

角色管理员中间表adminrole：管理员和角色对应的中间表，rid是角色表主键对应的外键，aid是管理员表主键对应的外键。

表1.3.11 角色管理员中间表

字段名
数据类型
描述
rid
int
角色Id
aid
int
管理员Id

2. 数据库结构图：

图 1.3.3 数据库结构图

（四）本章小结

本章主要介绍了系统的整体需求，对需求进行分析，然后思考实现需求所需要使用到的系统架构模式和系统运行环境，这里选择了使用MVC模式来进行开发。MVC分层设计模式将系统分成Model层，Controller层和View层，从而实现对系统多层之间的统一管理，降低系统耦合度，提交系统的可用性和可移植性。对数据库的表进行了设计，并展示各个字段的意义和设计思路，以及所有表格之间的对应主外键关系。

三、系统运行效果展示

（一）系统相关

1. 用户登录

描述：系统的入口，输入正确的用户名和密码后点击按钮即可登录，登录成功和失败都有相应的文字提示，并且登录失败是界面有抖动效果提示。

图 3.1.1 用户登录

2. 修改密码

描述：管理员可以在这里修改自己的密码，修改密码需要重复输入两次密码，两次密码相同才可修改，修改成功之后系统会强制退出，要求管理员重新登录。

图 3.1.2 修改密码

（二）商品管理

1. 商品列表

描述：实现商品的删改查功能，点击商品名称会弹窗显示商品详细信息，同时可以在弹窗中对商品信息进行修改和提交。该界面支持模糊查询功能，可以根据名称，分类，重量和保质期进行查询，点击右下角的导出按钮可以将html表格保存为本地excel文件。

图 3.2.1 商品列表

2. 添加商品

描述：添加商品界面，在这里可以新增商品的数据，会对输入的数据进行验证，只有验证成功才能进行添加，对于商品的名称采用ajax进行后台验证，已存在的商品名称不能通过验证。添加成功会有弹窗提示。

图 3.2.2 添加商品

3. 商品分类

描述：商品分类查询，进入页面后只显示所有的商品大类列表，点击对应大类的名称会根据大类查询到相应的细类并展示，点击细类则会查询到该细类下所有的商品进行展示。

图 3.2.3 商品分类

4. 商品回收站

描述：商品列表中被删除的商品不会直接从数据库中删除，而是进入商品回收站。在这里可以选择对商品信息

进行恢复或者是将商品信息永久删除，这里删除的商品将会从数据库中删除，无法恢复，点击右下角的导出按钮可以将html表格保存为本地excel文件。

图 3.2.4 商品回收站

5. 订单列表

描述：显示所有的订单，对订单进行模糊查询。可以修改订单的状态，订单状态分为四种：已下单，已发货，已收货和已关闭。对应不同的订单状态，右侧的操作栏会发生相应的改变，点击右下角的导出按钮可以将html表格保存为本地excel文件。

图 3.2.5 订单列表

(三) 员工管理

1. 员工列表

描述：实现员工的删改查功能，点击员工名称会弹窗显示员工详细信息，同时可以在弹出的窗口中对员工信息进行修改和提交。该界面支持模糊查询功能，可以根据员工的姓名，年龄，性别和工种进行查询，点击右下角的导出按钮可以将html表格保存为本地excel文件。

图 3.3.1 员工列表

2. 新增员工

描述：在本页面可以对员工数据进行新增操作，员工的新增会进行严格的表单验证，对于员工的姓名采用ajax进行后台验证，已存在的姓名不能通过验证。添加成功会有弹窗提示。

图 3.3.2 添加员工

(四) 会员管理

1. 会员列表

描述：会员管理列表显示所有的会员信息，进行删改查的操作。点击会员名称会弹窗显示详细信息，可以在弹窗中对会员的数据进行修改和提交。该页面支持模糊匹配，可以对会员的名称，电话和会员等级进行查询，点击右下角的导出按钮可以将html表格保存为本地excel文件。

图 3.4.1 会员列表

2. 添加会员

描述：对会员进行添加操作，通过验证的会员信息才可以添加，姓名使用ajax进行后台验证，已经存在的姓名无法添加。

图 3.4.2 添加会员

(五) 权限管理

1. 管理员列表

描述：管理员列表，显示所有的管理员信息并可进行增删改查操作，一个管理员账号可以拥有多个角色，这里会显示管理员所有拥有的角色和权限。点击管理员名称会进入修改页面，可以对管理员的账号和密码进行修改，也可以更改管理员所拥有的角色。

图 3.5.1 管理员列表

2. 添加管理员

描述：可以再此页面对管理员的数据进行新增，进入此页面时，系统会查询出所有可以使用的角色并列供用户选择，一个管理员可以拥有多个角色。管理员的姓名进行ajax后台验证，已经存在的管理员姓名不能使用。添加管理原始需要输入两次密码，两次密码相同才可以进行添加。

图 3.5.2 添加管理员

3. 角色列表

描述：角色列表，一个角色可以拥有多个权限，每个权限对应同名的菜单列表，没有权限的话将无法进入对应菜单。这里会显示角色的名称以及该角色拥有的所有权限。

图 3.5.3 角色列表

4. 添加角色

描述：可以在此页面对角色的数据进行新增，进入页面时会查询出所有可以使用的权限供用户选择，一个角色可以拥有多个权限。角色的名称使用ajax进行后台验证，已经存在的角色名称不能使用。

图 3.5.4 添加角色

(六) 统计图表

1. 商品统计

描述：统计图表页面，包含了商品统计，员工统计，会员统计，订单统计，每个页面都会以柱状图和饼图的形式将对应的数据进行展示，使用百度的开源h5插件ECharts来实现。点击下载按钮可以以图片的形式导出图表。除了商品统计之外，统计图表模块还包括员工统计，会员统计，订单统计等3个功能，每个功能展示对应模块的图表数据信息。

图 3.6.1 商品统计

(七) 本章小结

本章主要介绍了系统的各个模块的运行效果展示，系统分为商品管理、员工管理、会员管理、权限管理和统计图表等几个模块，本章主要对各个模块的详细使用流程进行分析，介绍界面的使用方法和一些特殊的功能，如后台验证、导出图表的实现方式。

四、后端代码示例

(一) MemberAction.java

此类是会员管理模块的Controller层控制器类，主要负责接受前端发送的请求，经过SpringMVC映射后找到请求中对应的方法，然后调用Service层进行业务处理操作，之后将操作后的数据返回到前台jsp页面进行展示。

```
@Controller@RequestMapping(value = "/member")public class MemberAction { @Resource MemberService memberService; @RequestMapping(value = "/checkmname.do", method = RequestMethod.POST) @ResponseBody public String checkMName(String mname){ Gson gson = new Gson(); Map String, Boolean map = new HashMap (); map.put("valid", memberService.checkMName(mname)); String str = gson.toJson(map).toString(); return str; } @RequestMapping(value = "/qrymember.do", method = RequestMethod.GET) public String getMembers(HttpServletRequest request, String page, String pageSize, Member member){ getMembers(member, page, pageSize, request); return "hylb"; } @RequestMapping(value = "/delmember.do", method = RequestMethod.GET) public String delMembers(HttpServletRequest request, String page, String pageSize, Member member){ Map String, Object delParamMap = new HashMap (); delParamMap.put("member", member); Integer i = memberService.delMember(delParamMap); getMembers(member, page, pageSize, request); return "hylb"; } @RequestMapping(value = "/updmember.do", method = RequestMethod.GET) public String updMembers(HttpServletRequest request, String page, String pageSize, Member member){ if(!MyUtil.isEmpty(member)){ memberService.updMember(MyUtil.beanToMap(member)); } getMembers(member, page, pageSize, request); return "hylb"; } @RequestMapping(value = "/addmember.do", method = RequestMethod.POST) @ResponseBody public Integer addMembers(Member member){ try { if (!MyUtil.isEmpty(member)) { memberService.addMember(MyUtil.beanToMap(member)); } } catch (Exception e){ e.printStackTrace(); return 0; } return 1; } private void getMembers(Member member, String page, String pageSize, HttpServletRequest request){ Map String, Object paramMap = PageUtil.getParamMap(member, page, pageSize); paramMap.put("mname", member.getMname()); paramMap.put("mtel", member.getMtel()); paramMap.put("mlevel", member.getMlevel()); List Member members = memberService.qryMember(paramMap); request.setAttribute("members", members); request.setAttribute("page", page); request.setAttribute("pageSize", pageSize); request.setAttribute("pageCount", getPageCount(member, pageSize)); } private Integer getPageCount(Member member, String pageSize){ List Member members = memberService.qryAllMember(); return PageUtil.getPageCount(members.size(), pageSize); }}
```

(二) MemberService.java

此类是会员管理模块的业务层接口，定义了一些要在控制层中使用的方法，接口并不给出具体实现，而是由其实现类MemberServiceImpl来对方法进行实现。

```
public interface MemberService { List Member qryMember(Map String, Object paramMap); Integer delMember(Map String, Object paramMap); Integer updMember(Map String, Object paramMap); Integer addMember(Map String, Object paramMap); List Member qryAllMember(); boolean checkMName(String mname);}
```

(三) MemberServiceImpl.java

此类是MemberService接口的实现类，实现了接口中的所有方法，主要职责是调用数据访问层的接口进行数据库访问操作，然后对操作的结果进行一些业务处理之后返回到控制层。


```

@Service("memberServiceImpl")public class MemberServiceImpl implements MemberService{ @Resource private
MemberDao memberDao; @Override public List Member qryMember(Map String, Object paramMap) {
return memberDao.qryMember(paramMap); } @Override public Integer delMember(Map String, Object
paramMap) { Map String, Object map = new HashMap (); map.put("memid",
((Member)paramMap.get("member")).getMemid()); return memberDao.delMember(map); } @Override public
Integer updMember(Map String, Object paramMap) { return memberDao.updMember(paramMap); } @Override
public Integer addMember(Map String, Object paramMap) { return memberDao.addMember(paramMap); }
@Override public List Member qryAllMember() { return memberDao.qryAllMember(); } @Override public
boolean checkMName(String mname) { List Member commoditys = memberDao.checkMName(mname);
if(commoditys.size() == 0) return true; return false; }}

```

(四) MemberDao.java

此类是会员管理的数据访问层接口，通过Mybatis的映射Mapper文件来执行对应的SQL语句，对数据库进行增删改查操作并将结果返回业务层。

```

public interface MemberDao { List Member qryMember(Map String, Object paramMap); Integer
delMember(Map String, Object paramMap); Integer updMember(Map String, Object paramMap); Integer
addMember(Map String, Object paramMap); List Member qryAllMember(); List Member
checkMName(String mname);}

```

(五) Member.java

此类是系统中对应会员表（member）的实体类，主要封装了会员表中所有的字段，在进行前后端和mvc各层中的数据传输的时候负责对数据进行封装的左右。

```

public class Member { private Integer memid; private String mname; private Long mtel; private String mcode; private
Integer mlevel; private Double mrem; private Date mdate; private Double mcount; public Integer getMemid() {
return memid; } public void setMemid(Integer memid) { this.memid = memid; } public String getMname() {
return mname; } public void setMname(String mname) { this.mname = mname; } public Long getMtel() { return
mtel; } public void setMtel(Long mtel) { this.mtel = mtel; } public String getMcode() { return mcode; } public
void setMcode(String mcode) { this.mcode = mcode; } public Integer getMlevel() { return mlevel; } public void
setMlevel(Integer mlevel) { this.mlevel = mlevel; } public Double getMrem() { return mrem; } public void
setMrem(Double mrem) { this.mrem = mrem; } public Date getMdate() { return mdate; } public void
setMdate(Date mdate) { this.mdate = mdate; } public Double getMcount() { return mcount; } public void
setMcount(Double mcount) { this.mcount = mcount; } @Override public String toString() { return "Member{" +
"memid=" + memid + ", mname=" + mname + ", mtel=" + mtel + ", mcode=" + mcode +
", mlevel=" + mlevel + ", mrem=" + mrem + ", mdate=" + mdate + ", mcount=" + mcount
+ '}'; } public Member(Integer memid, String mname, Long mtel, String mcode, Integer mlevel, Double mrem,
Date mdate, Double mcount) { this.memid = memid; this.mname = mname; this.mtel = mtel; this.mcode =
mcode; this.mlevel = mlevel; this.mrem = mrem; this.mdate = mdate; this.mcount = mcount; } public
Member() { }}

```

(六) 本章小结

本章主要展示了系统会员管理模块的后端逻辑代码，包括控制层的MemberAction、业务层的MemberService和其实现类MemberServiceImpl、数据访问层的MemberDao和数据库表实体类Member的代码实现。由于系统使用SpringMVC的注解方式进行开发，不需要配置Spring文件中的bean，只要在对应的类上添加注解即可，如MemberService上的@Service、MemberAction上的@Controller和其内部内部MemberService对象上的@Resource注解。通过这些注解，SpringMVC和Spring和自动的对对应的对象进行注入，而不用我们手动去创建新对象，从而实现了控制反转的效果，降低系统耦合度。

系统测试

测试说明

本系统采用白盒测试模式，通过对系统前端页面的多次操作，查看业务逻辑是否正确，以及后端数据库是否生成了对应的数据。

测试用例

表5.1.1 测试用例

模块名称

测试项

操作步骤

预期结果

实际结果

系统相关

用户登录

输入用户名

输入用户密码

控件是否正确

控件正确，登录成功

跳转主页

与预期结果一致

系统相关

退出登录

1.点击退出登录

退出登录，回到登录界面

Session内容清空

与预期结果一致

商品管理

商品列表

点击商品列表

点击商品名称

点击修改商品

点击删除商品

点击导出按钮

正确显示商品列表

正确显示商品详细信息

修改商品成功

弹窗提示，确认后删除

导出excel成功

与预期结果一致

商品管理

添加商品

输入正确的商品信息，点击提交

输入错误的商品信息，点击提交

弹窗显示提交成功，页面刷新清空数据

页面控件显示验证失败，无法进行插入

与预期结果一致

商品管理

商品分类

点击任意一个商品大类

点击任意一个商品细类

显示大类对应的所有细类

显示细类对应的所有商品名称

与预期结果一致

商品管理

商品回收站
点击恢复按钮
点击删除按钮
点击导出按钮
恢复商品成功
弹窗提示，确认后删除成功
导出excel成功
与预期结果一致
商品管理
订单列表
修改订单的状态
点击关闭按钮
点击导出按钮
修改成功
关闭成功
导出excel成功
与预期结果一致
员工管理
员工列表
点击员工姓名
点击删除按钮
点击导出按钮
弹窗显示员工详细信息
成功删除员工
导出excel成功
与预期结果一致
员工管理
添加员工
输入正确的员工信息，点击提交
输入错误的员工信息，点击提交
显示提交成功
验证不通过，提交失败
与预期结果一致
会员管理
会员列表
点击删除按钮
点击导出按钮
输入查询条件后点击查询按钮
数据删除成功
导出excel成功
页面显示条件查询之后的结果
与预期结果一致
会员管理
添加会员
输入正确的会员信息，点击提交
输入错误的会员信息，点击提交
弹窗显示提交成功，页面刷新清空数据
页面控件显示验证失败，无法进行插入

与预期结果一致

权限管理

管理员列表

点击账号名称

点击删除按钮

点击导出按钮

跳转到修改管理员界面

删除管理员成功

导出excel成功

与预期结果一致

权限管理

添加管理员

输入正确的会员信息，点击提交

输入错误的会员信息，点击提交

弹窗显示提交成功，页面刷新清空数据

页面控件显示验证失败，无法进行插入

与预期结果一致

权限管理

角色列表

点击删除按钮

删除角色成功

与预期结果一致

权限管理

添加角色

输入正确的会员信息，点击提交

输入错误的会员信息，点击提交

弹窗显示提交成功，页面刷新清空数据

页面控件显示验证失败，无法进行插入

与预期结果一致

统计图表

商品统计

点击菜单栏对应列表

显示商品统计对应的饼图和柱状图

与预期结果一致

统计图表

会员统计

点击菜单栏对应列表

显示会员统计对应的饼图和柱状图

与预期结果一致

统计图表

员工统计

点击菜单栏对应列表

显示员工统计对应的饼图和柱状图

与预期结果一致

统计图表

订单统计

点击菜单栏对应列表

显示订单统计对应的饼图和柱状图

与预期结果一致

(三) 部分测试结果示例

1. 商品详细信息

图 5.3.1 商品详细信息

2. 删除商品

图 5.3.2 删除商品

3. 添加商品

图 5.3.3 添加商品

结束语

毕业设计是对大学思念学习的一个总结和检测,也是对自己能力的一个肯定,通过这几个月制作毕业设计和写论文,我学习到了很多,在中途遇到了很多问题,也都通过自己的方法解决了,这让我的个人能力有了极大的提升。

刚开始进行毕业设计的时候,由于急着早点开始工作,所以在编程的过程中忽略了很多设计方面的问题,导致之后的开发过程十分困难,在进行一段时间的开发后我意识到这个问题的严重性,所以直接将整个项目推翻重做,从头开始对项目的需求进行分析,并严格按照软件工程的开发流程对系统进行开发,虽然这浪费了我不少的时间,当时通过这样做,我在之后的开发流程中方便了很多。

项目的开发过程中遇到了各式各样的问题,其中有后端数据的问题,也有前端页面的问题,比如前端ajax无法向后端传值,数据库连接出现问题,还有springmvc无法将请求映射到对应的controller等等,这些问题我通过不断的debug和上网查找资料,最终都得以解决。而在解决问题的过程中,我也不断的在学习着,现在我的编程能力相对于之前也有了很大提升。

本次毕业设计,实现了超市管理的大部分功能,包括对于各种商品的管理和分类,对员工和会员的管理,以及各种统计图表的展示功能,可以说基本上达到了预期的效果。不过本系统仍然存在大量的不足之处,相对于真正的超市管理系统来说还差的很远,一个最严重的地方就是在设计的时候没有考虑到并发的的问题,在并发量低的情况下使用可能没有问题,然而一旦并发量高了,就可能出现数据不同步或者系统崩溃的问题。这些问题现在的我还没有能力解决,不过在以后我仍然会不断的去完善系统这当面的问题。

参考文献

- [1]王延翔,杨金民.基于分层分类的J2EE应用系统异常处理方法[J].计算机应用研究,2015,32(03):776-780.
- [2]王令群,何世钧,袁小华,张术台.基于J2EE和云计算的智慧社区架构设计[J].实验室研究与探索,2014,33(01):123-127.
- [3]杜长霄,李晓红,石红,冯志勇.J2EE应用软件的架构安全评估方法[J].计算机科学与探索,2014,8(05):572-581.
- [4]唐倩,张伟.轻量级J2EE中SSH框架的研究及其应用[J].物联网技术,2013,3(12):52-55.
- [5]吕雪,凌捷.基于J2EE架构的信息安全应急预案管理系统研究与实现[J].计算机工程与设计,2013,34(04):1197-1201+1237.
- [6]王俊艳,田小龙.中小型超市管理系统的分析与设计[J].装备制造技术,2012(12):57-59.
- [7]高航.基于会员的超市管理系统分析与设计[J].电子设计工程,2012,20(13):47-49+53.
- [8]李洋,孙永维,许冰,王英双.基于Ajax,Struts,Hibernate和Spring的J2EE架构[J].吉林大学学报(信息科学版),2011,29(06):576-584.
- [9]万东.基于Struts+Hibernate+Spring的轻量级J2EE框架[J].现代电子技术,2011,34(16):39-41.
- [10]田娟,徐钊.基于J2EE的MVC设计模式的分析与思考[J].计算机与现代化,2010(10):54-58.
- [11]闫俊伢,安俊秀.J2EE技术体系的探讨与研究[J].实验室研究与探索,2010,29(07):83-86.
- [12]林卓.基于JSP的图书超市管理系统设计与实现[J].山西大同大学学报(自然科学版),2009,25(05):16-17+20.
- [13]赵洋,张丽,王恩东,张素宁.基于Struts,Hibernate和Spring的J2EE架构研究[J].现代电子技术,2009,32(02):107-110.
- [14]胡启敏,薛锦云,钟林辉.基于Spring框架的轻量级J2EE架构与应用[J].计算机工程与应用,2008(05):115-118+133.
- [15]彭晓川.基于Flex的RIA与J2EE应用的整合[J].电脑与电信,2008(02):37-39.
- [16]刘泽轩,江春华.基于J2EE架构的科研管理系统的设计与实现[J].计算机工程与设计,2007(21):5218-5220+5247.
- [17]陶以政,吴志杰,唐定勇,席传裕,姜龙,李兴兰.基于J2EE的应用框架技术研究[J].计算机工程与设计,2007(04):826-828.
- [18]吕毅,葛玮,郝克刚.Spring轻量级J2EE开发框架的研究[J].计算机技术与发展,2006(12):208-211.

- [19]雷钧,徐洪胜,付勇智.MVC设计模式在J2EE平台上的应用[J].微计算机信息,2006(21):45-48.
- [20]林泊,周明辉,刘天成,黄昱,梅宏.一个J2EE应用服务器的Web容器集成框架[J].软件学报,2006(05):1195-1203.
- [21]王卫平,王松涛,王名茗.一种基于J2EE、Spring和Hibernate的轻量级EAI构架[J].计算机系统应用,2005(11):38-41.
- [22]夏天.大型超市管理系统的方案设计[J].华中农业大学学报(社会科学版),2003(03):39-43.
- [23]李焕哲,刘晓亮,郭大权,王凡,曹强,马书敏.J2EE体系结构探讨[J].河北省科学院学报,2003(03):152-156.
- [24]张珍宝,刘升.超市管理系统的分析与设计[J].高等函授学报(自然科学版),2002(03):49-51.

致 谢

这次毕业设计的完成,有很多人给了我帮助,在这里对他们表示真心的感谢。首先要感谢的就是我的指导老师关玉蓉老师,在我写论文的过程中她给了我很多的帮助,也为我指出了很多不足之处,并给出了好的解决方案。其次要感谢的同学和朋友们,在项目的制作过程中,他们给我提出了很多的建议,帮我想了很多的好点子,通过他们的帮助我的项目变得越来越好,功能也越来越完善。在这里对所有帮助过我的人表示衷心的感谢!

四、指标说明:

1. 总相似比即类似于重合率。总相似比即送检论文中与检测范围所有文献相似的部分(包括参考引用部分)占整个送检论文的比重,总相似比=复写率+引用率。
2. 引用率即送检论文中被系统识别为引用的部分占整个送检论文的比重(引用部分一般指正确标示引用的部分)。
3. 自写率即送检论文中剔除雷同片段和引用片段后占整个送检论文的比重,一般可用于论文的原创性和新颖性评价,自写率=1-复写率-引用率。
4. 复写率即送检论文中与检测范围所有文献相似的部分(不包括参考引用部分)占整个送检论文的比重。
5. 红色字体代表相似片段;浅蓝色字体代表引用片段、深蓝色字体代表可能遗漏的但被系统识别到与参考文献列表对应的引用片段;黑色字体代表自写片段。

五、免责声明:

鉴于论文检测技术的局限性以及论文检测样本库的局限性,Gocheck.cn网站不保证检测报告的绝对准确,相关结论仅供参考,不做法律依据。

Gocheck论文检测服务中使用的论文样本,除特别声明者外,其著作权归各自权利人享有。根据中华人民共和国著作权法相关规定,Gocheck网站为学习研究、介绍、评论、教学、科研等目的引用其论文片段属于合理使用。除非经原作者许可,请勿超出合理使用范围使用其内容和本网提供的检测报告。