成 绩:\_\_\_\_\_\_\_\_\_\_\_\_

教师签名:\_\_\_\_\_\_\_\_\_\_\_\_



**需求分析说明书**

**项目名称:** 超市管理系统项目

教 师: 程汇利

姓 名: 王东阳

学 号: 201710121112

班 级: 17gb计算2班

日 期: 2019年12月27日

目录

[一、前言](#_Toc7184) 1

[1.项目前言 1](#_Toc31335)

[2.编写目的 1](#_Toc26921)

[二、可行性研究 1](#_Toc9250)

[1.管理可行性 1](#_Toc31442)

[2.经济可行性 1](#_Toc11303)

[3.技术可行性 2](#_Toc5415)

[三、系统需求分析 2](#_Toc2177)

[1.系统功能概述 2](#_Toc18720)

2.系统功能需求.....................................2

3.需求功能详细介绍.................................3

[四、系统设计](#_Toc176)

1.系统功能流程图..................................4

2.系统设计的原则..................................5

3.业务流程分析....................................5

4.数据流图........................................7

5.数据库设计与实现................................7

[五、应用程序设计 15](#_Toc8425)

[1.Servlet的详细应用.............................15](#_Toc13719)

[2.特定模块的详细介绍.............................15](#_Toc14574)

六、附：页面前端程序代码.......................27

1. 登录入口页面................................27
2. 登录页面....................................27
3. 系统主界面..................................28
4. 修改密码....................................29

七、结束语.......................................31

**一．前言**

**1.项目前言**

**超市随着市场经济和现代信息技术的发展，不可必要的要卷入信息现代化的时代浪潮，如何使用现代化的崭新工具，让企业和经营者个人在最小的资金投入下获得最大的回报，是每个人都愿意追求的梦想。因而，在超市管理中引进现代化的办公软件，就成为当下最好的解决方法。**

**使用软件办公，不仅能使经营者的劳力资本大量减少，同时能使超市的管理更规范更合理，解决了超市中用于管理进出货以及购买，库存等众多繁琐，工作量大从而导致耗费人员过多，超市经营絮乱等众多问题问题。**

**2.编写目的**

1. **使广大的超市工作者能够脱离纯手工的原始的劳动，可以通过计算机让超市货物管理信息化。**
2. **提高超市企业的市场竞争力，保持小中型超市持续发展的优势。并且以此可以为工作者提供灵活，方便的操作。**
3. **节约商品管理的成本，提高商品管理的效率。**
4. **为企业实现整体信息化的其他相关系统提供必要的数据支持。**
5. **满足当今市场的销售进度，能最大程度均衡管理者与消费者的供求关系，做出满足于消费者与商家满意的平台。**

**二．可行性研究**

**1.管理可行性**

**随着超市管理水平的不断提高，在管理水平方面积累了丰富的经验方法，并建立了成熟的管理机制，齐全的规章制度。有正确的原始资料的支持，使管理工作有了制度支持和信息支持。**

**根据以往在经营中出现的困难与不便，超市的高层管理人员对新系统的开发持积极的态度，肯定新系统的开发，并且为新系统的开发与实施创造了各种良好的条件，制定了有利于系统实施的管理方法。目前管理方面的条件已经能够满足超市管理系统的开发以及日后的实施。因而在管理上是可行的。**

**2.经济可行性**

**本系统需要超市负担的费用主要是计算机硬件设备与软件技术，包括本系统的开发及维护费用，对超市人员使用本系统的培训费用，以及本系统开发完成后-，由于工作方式的改变增加的其他开支。**

**该系统带来的经济效益是明显的，其最主要的表现是减少了企业管理费用和人力开支，不仅方便对进货信息的管理，也减少了由于人工的出错的几率，降低了成本，节省了大量的时间，还为企业的各项决策提供了宝贵的资料，为企业带来了巨大的经济效益。所以经济上也是可行的。**

**3.技术可行性**

**在IT行业中从业的工作人员一般都要熟悉要求掌握计算机技术，具有一定的软硬件基础，会使用各种管理软件，熟悉IT产品。因为，有的超市对员工的素质要求比较高，从管理层到下面的销售人员，都要求具有一定的计算机基础，而本系统简单易用，所以在新系统投入使用时，只要对员工进行少量的培训，系统的功能和使用方法就基本上能够使系统顺利运行。所以在技术上是可性的。**

**综合以上三个方面，该系统具有很高的开发可行性，无论是从技术上或者经济上还是管理上。因此，可以设计该系统的数据流程图，建立数据字典。**

**三．系统需求分析**

**1.系统功能概述**

**本系统主要完成超市管理（商品信息管理、供应商管理、进货管理、退换货管理、客户管理、客户退货管理、仓库设置、库存管理）等功能。通过以上功能实现简单、便捷的超市管理系统。**

**2.系统功能需求**

**商品信息管理：管理员可以对商品进行添加、修改和查询操作。添加库存商品时侯要填入：商品名称、入库价格（即进货价格，进货价格的查询只能是以管理员身份登陆才能查看）、商品卖价（同入库价格一样）、单位预警量（及库存下限）。最主要的是对商品信息进行查询：商品信息查询可以根据商品类型查询，还可以精确查找，也可以模糊查找，查询结果包括：商品编号、商品名称、商品类型、商品售价、商品数量。**

**供应商管理：供应商管理可以对厂商信息进行查询。供应商信息查询可以根据厂商名称查询、负责人查询和最近联系厂商查询。查询结果：供应商。**

**商品销售管理：销售登记需要登记：（单据信息）：单号、客户名称、出货仓库、销售日期、总金额、应收金额、实收金额、`经办人备注（订单信息）:商品编号、商品名称、单位、规格型号、颜色、单价、数量。需要出库登记：商品名称、出库数量、经办人、出库时间。**

**客户（用户）管理：客户管理主要是对于本超市的有过交易记录的人群的信息进行登记和查询。登记客户信息包括：客户名称、负责人、联系地址、联系电话。查询可以通过时间来查询（查询最近联系的客户）、客户名，查询最近联系的客户。**

**退换货管理：退货方式有两种：一种是整单退货，一种是部分退货。登记内容包括：（订单信息）：单号（随机产生流水号，是唯一的）、出货仓库、退货时间、供应商名称、总金额应退金额实退金额经办人备注（订单信息）：商品编号、商品名称、单位、规格型号、颜色、单价、数量。退货查询可以根据单据号、供应商和开单时间查询。查询结果包括：单号、开单时间、供应商名称、仓库名称、经办人、总金额（单据详细信息）：商品编号、商品名称、单位、规格型号、颜色、单价、数量。**

**商品销售管理：销售登记需要登记：（单据信息）：单号、客户名称、出货仓库、销售日期、总金额、应收金额、实收金额、经办人、备注（订单信息）：商品编号、商品名称、单位、规格型号、颜色、单价、数量。需要出库登记：商品名称、出库数量、经办人、出库时间。**

**客户管理：客户管理主要是对本超市的有过交易记录的人群的信息进行登记和查询。登记客户信息包括：客户名称、负责人、联系地址、联系电话。查询可以通过时间来查询（查询最近联系的客户）、客户名，查询最近联系的客户。**

**仓库设置：仓库设置包括增删改的操作：仓库名称、负责人、联系电话、仓库地址、仓库信息备注。**

**库存管理：库存出入登记及商品在进出仓库的时候都需要登记。入库的情况包括：商品的采购、客户退货。出库的情况包括：商品销售、采购退货。库存的出入查询可以根据：单据号、时间来查询。查询结果包括：（单据信息）单号、出入的仓库、出入仓库的日期、经办人。添加库存商品及添加商品种类，添加时需要登记：商品名称、入库时间、商品卖价、单位、及库存下限。同时还具有库存商品预警查询功能。**

**客户退货管理：客户退货管理是在本超市购买商品之后，但产品出现质量问题、货物迟到等原因超市和客户协商要求退货或换货的制度。客户退货需要登记：单号、（订单信息）客户名称、收货仓库、收货日期、总金额、应退金额、实退金额、经办人、备注（订单信息）：商品编号、商品名称、单位、规格型号、颜色、单价、数量。销售退货查询可以根据：客户名称、单据号、开单日期查询。需要注意的是：客户退货之后是否退回到仓库，所以要进行入库登记。**

**3.需求功能详细介绍**

**3.1客户界面部分**

**登陆界面**

登陆界面验证用户名、密码,用户名和密码不能为空。信息不匹配通过弹窗给出对应错误信息，并将验证码重置。点击重置按钮可将所有信息重置。

**主界面**

通过卡片布局对主界面进行布局，进到主界面之后默认是欢迎界面，点击相应的按钮能进入相应的卡片。将当前登录的用户显示在界面中间。界面右上角显示实施用户名，通过多线程完成。若用户级别为项目经理，可是使用界面中所有功能。若等级为普通用户，则只能使用账单管理和退出系统功能。

**3.2管理界面部分**

**用户、供应商、账单的增删改查**

**用户管理**

用户管理界面可以实现增删改查的功能，其中修改功能，部门经理只能修改普通员工和自己，删除功能，部门经理只能删除普通员工。查询按钮可以通过用户名查询指定用户的信息，若用户名称为空，则查询所有。

**供应商管理**

供应商管理界面可以实现增删改查的功能，其中删除按钮，如果账单表中有此供应商信息根据主外键关系，则不能删除，给出相应提示。点击导入按钮，选择对应的文件可以将文件信息导入。查询按钮可以通过供应商名称查询指定供应商的信息，若供应商名称为空，则查询所有。

**账单管理**

账单管理界面可以实现增删改查的功能，点击导出按钮，选择对应的行数可以将信息导出到指定路径。查询按钮可以通过商品名称和是否付款查询指定账单的信息，若两个都为空，则查询所有。

**报表**

**供应商分组查询报表**

按照供应商进行分组查询相应的信息

**商品分组查询报表**

按照商品进行分组查询相应的信息

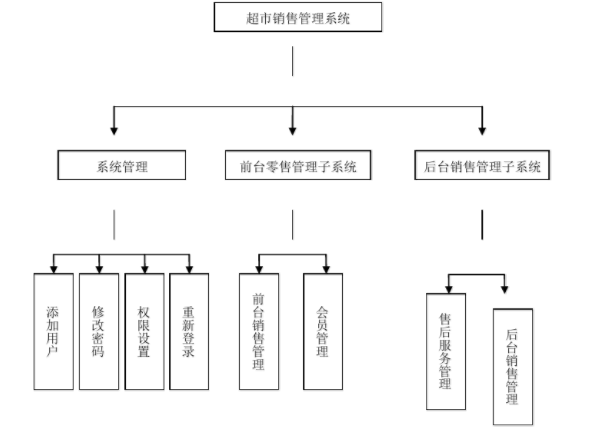
**退出系统**

点击退出系统按钮会弹出一个确认提示框，点击是退出系统，点击否或者取消返回界面。

**四．系统设计**

**1. 系统功能流程图**

3

****

**2.系统设计的原则**

**①易用性原则：为了更直接的面向非IT专业的用户，符合用户的操作使用习惯，系统界面应该突出清晰美观，简单易懂的特点。**

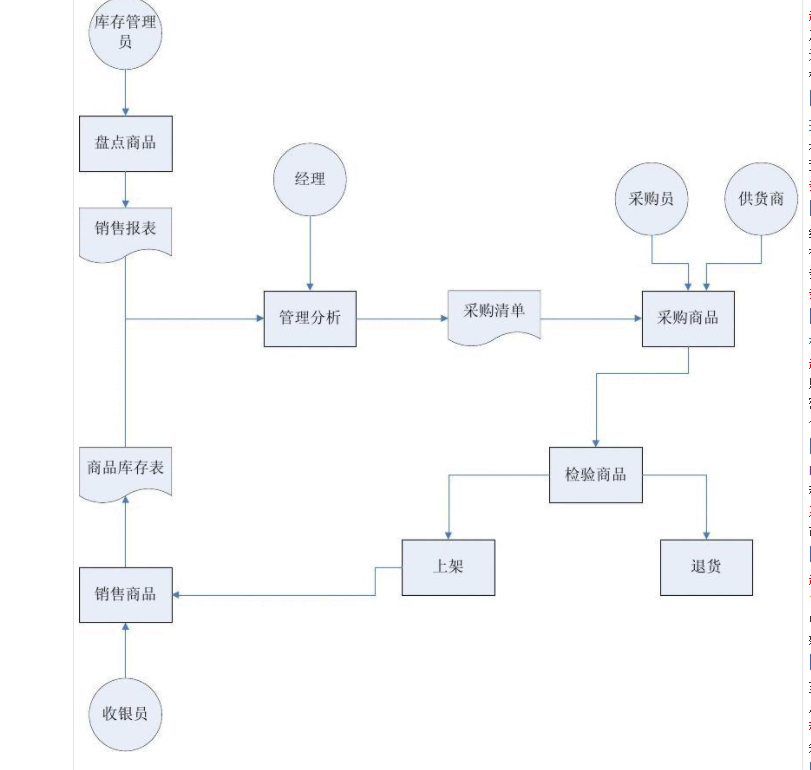
**②实用性原则：设计出的系统不仅要求满足用户以及管理者供应商的网上交易的正常流程，还要有一定的实用性，在用户人群中，让人们都能认为此系统比较适合大众群体广泛使用。**

**3. 业务流程分析**

**超市管理系统主要是以先进的管理理念和科学的计算机技术来取代原始的手工操作。在新系统开发前，我们对这个超市管理系统业务流程进行了系统的调查。得到下面的流程图**

****

**4. 数据流图**

**5.数据库设计与实现**

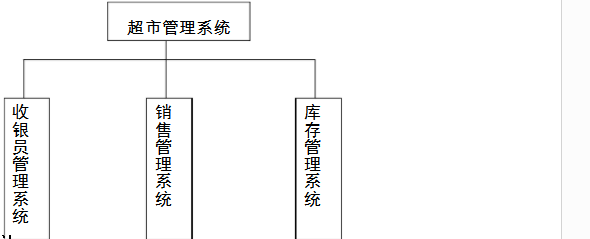
数据库是此类系统的核心，系统中所有的信息都存储在数据库中，在所有jsp页面中通过JDBC与数据库连接，操作数据库中的数据，从而实现所有需要的功能。数据库软件有很多。本系统中数据库采用MySQL作为数据库工具。数据库是网站的核心，数据库设计的合理与否对网站的建设有着至关重要的影响。一个良好的数据库开发过程一般是先系统分析，再逻辑设计，然后物理实现。系统分析中分析数据库用户需求，并画出E-R图直观地表示出数据库系统的需求。逻辑设计中根据系统分析设计的一个关系模型。物理设计为逻辑设计中的数据模型选取一个最合理的应用环境的物理结构，通常选取一个常用数据库软件，用数据库软件创建与逻辑设计中数据模型对应的数据库。

**1）数据库目的与要求**

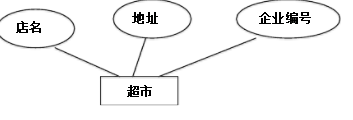
**包括对超市管理系统进行数据库设计，有需求分析，概念设计，逻辑设计和物理设计。**

**2）实验内容和步骤**

**①需求分析**

** ② 概念设计**

1. **实体及其属性**
2. **超市E-R图：**

****

1. **收银员 E-R图**
2. **收银小票E-R图**
3. **商品库存E-R图**

**③ 逻辑设计**

**超市表（企业编号，店号，地址），主键为企业编号，符合三范式**

**。**

**收银员表（收银员编号，姓名，性别，工资），主键为收银员编号，符合三范式。**

**用户登录表（用户编号，用户名，用户账号，性别，年龄，电话，地址,权限）主键为编号ID，符合三范式。**

**商品库存表（商品编号，商品名称，单价，库存量，供应商名）主键为商品编号，符合三范式。**

**生成账单表（账单编号，商品名称，供应商，账单金额，是否付款，操作时间）主键为账单编号，符合三范式。**

**供应商信息（供应商编号，供应商名称，联系人，联系电话，传真，操作时间） ，主键为供应商编号。**

**④ 数据库物理实现**

经过数据库系统分析后，数据库的结构已经非常清晰，剩下的就是用一种数据库软件实现这样的结构。本系统中数据库采用MySQL作为数据库。首先我们在MySQL中建一个数据库，命名为“supermarket”。 选中supermarket数据库，向数据库中添加3个表：gongyingshang，yonghu，zhangdan。下面是每个数据表中的属性。

1. gongyingshang表。gongyingshang表用来存储供应商信息。

gongyingshang表

|  |  |
| --- | --- |
| 字段名称 | 数据类型 |
| gys\_id | 数字 |
| gys\_name | 文本 |
| gys\_miaoshu | 文本 |
| gys\_lianxiren | 文本 |
| gys\_phone | 文本 |
| gys\_chuanzhen | 数字 |
| gys\_address | 数字 |

1. yonghu表。yonghu表用来存储领登陆用户明细。

|  |  |
| --- | --- |
| 字段名称 | 数据类型 |
| yh\_id | 数字 |
| yh\_username | 文本 |
| yh\_password | 文本 |
| yh\_sex | 文本 |
| yh\_age | 数字 |
| yh\_phone | 数字 |
| yh\_address | 文本 |
| yh\_quanxian | 文本 |

1. zhangdan表。zhangdan表用来存储账单的信息。

zhangdan表

|  |  |
| --- | --- |
| 字段名称 | 数据类型 |
| zd\_id | 数字 |
| zd\_spmingcheng | 文本 |
| zd\_spshuliang | 数字 |
| zd\_jymoney | 数字 |
| zd\_sfzhifu | 文本 |
| zd\_gysname | 文本 |
| zd\_spmiaoshu | 文本 |
| zd\_data | 文本 |
| zd\_danwei | 文本 |

1. 数据库代码

1. **SQLyog 企业版 - MySQL GUI v8.14**
2. **MySQL - 5.5.40 : Database - smbms**
3. **\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***
4. **\*/**
5. **/\*!40101 SET NAMES utf8 \*/;**
6. **/\*!40101 SET SQL\_MODE=''\*/;**
7. **/\*!40014 SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0 \*/;**
8. **/\*!40014 SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0 \*/;**
9. **/\*!40101 SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='NO\_AUTO\_VALUE\_ON\_ZERO' \*/;**
10. **/\*!40111 SET @OLD\_SQL\_NOTES=@@SQL\_NOTES, SQL\_NOTES=0 \*/;**
11. **CREATE DATABASE /\*!32312 IF NOT EXISTS\*/`smbms` /\*!40100 DEFAULT CHARACTER SET utf8 \*/;**
12. **USE `smbms`;**
13. **/\*Table structure for table `smbms\_bill` \*/**
14. **DROP TABLE IF EXISTS `smbms\_bill`;**
15. **CREATE TABLE `smbms\_bill` (**
16. **`id` bigint(20) NOT NULL AUTO\_INCREMENT COMMENT '主键ID',**
17. **`billCode` varchar(20) COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '账单编码',**
18. **`productName` varchar(50) COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '商品名称',**
19. **`productDesc` varchar(50) COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '商品描述',**
20. **`productUnit` varchar(10) COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '单位',**
21. **`productCount` decimal(20,2) DEFAULT NULL COMMENT '商品数量',**
22. **`totalPrice` decimal(20,2) DEFAULT NULL COMMENT '账单总金额',**
23. **`isPayment` int(10) DEFAULT NULL COMMENT '是否付款（1 未支付 2 已支付）',**
24. **`providerId` bigint(20) DEFAULT NULL COMMENT '供应商ID',**
25. **`creationDate` datetime DEFAULT NULL COMMENT '创建时间',**
26. **`createdBy` bigint(20) DEFAULT NULL COMMENT '创建者',**
27. **`modifyDate` datetime DEFAULT NULL COMMENT '更新时间',**
28. **`modifyBy` bigint(20) DEFAULT NULL COMMENT '更新者',**
29. **PRIMARY KEY (`id`)**
30. **) ENGINE=InnoDB AUTO\_INCREMENT=23 DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci;**
31. **/\*Data for the table `smbms\_bill` \*/**
32. **insert into `smbms\_bill`(`id`,`billCode`,`productName`,`productDesc`,`productUnit`,`productCount`,`totalPrice`,`isPayment`,`providerId`,`creationDate`,`createdBy`,`modifyDate`,`modifyBy`) values (2,'111','112','2222','112','11.20','111.20',2,2,'2016-10-28 20:28:30',6,'2016-10-28 23:04:04',6),(3,'1','1','11','1','111.00','1.00',1,1,'2016-10-28 20:33:43',6,NULL,NULL),(4,' 12312','31231','3123123','2321','3123.00','12312.00',2,1,'2016-10-28 20:38:31',6,NULL,NULL),(9,'23','321',NULL,NULL,NULL,NULL,1,1,NULL,NULL,NULL,NULL),(10,'231','231',NULL,NULL,NULL,NULL,2,2,NULL,NULL,NULL,NULL),(11,'231','213',NULL,NULL,NULL,NULL,1,1,NULL,NULL,NULL,NULL),(12,'231','32',NULL,NULL,NULL,NULL,1,2,NULL,NULL,NULL,NULL),(13,'231','123',NULL,NULL,NULL,NULL,2,1,NULL,NULL,NULL,NULL),(14,'321','231',NULL,NULL,NULL,NULL,1,2,NULL,NULL,NULL,NULL),(15,'1323','132',NULL,NULL,NULL,NULL,2,1,NULL,NULL,NULL,NULL),(16,'123','213',NULL,NULL,NULL,NULL,2,2,NULL,NULL,NULL,NULL),(17,'312','312',NULL,NULL,NULL,NULL,2,1,NULL,NULL,NULL,NULL),(18,'12','23',NULL,NULL,NULL,NULL,1,1,NULL,NULL,NULL,NULL),(19,'12','123',NULL,NULL,NULL,NULL,2,2,NULL,NULL,NULL,NULL),(20,'15','132',NULL,NULL,NULL,NULL,2,2,NULL,NULL,NULL,NULL),(21,'21','132',NULL,NULL,NULL,NULL,1,2,NULL,NULL,NULL,NULL),(22,'34',NULL,NULL,NULL,NULL,NULL,2,1,NULL,NULL,NULL,NULL);**
33. **/\*Table structure for table `smbms\_provider` \*/**
34. **DROP TABLE IF EXISTS `smbms\_provider`;**
35. **CREATE TABLE `smbms\_provider` (**
36. **`id` bigint(20) NOT NULL AUTO\_INCREMENT COMMENT '主键ID',**
37. **`proCode` varchar(20) COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '供应商编码',**
38. **`proName` varchar(20) COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '供应商名称',**
39. **`proDesc` varchar(50) COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '供应商描述信息',**
40. **`proContact` varchar(20) COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '供应商联系人',**
41. **`proPhone` varchar(20) COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '联系电话',**
42. **`proAddress` varchar(50) COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '地址',**
43. **`proFax` varchar(20) COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '传真',**
44. **`createdBy` bigint(20) DEFAULT NULL COMMENT '创建者（userId）',**
45. **`creationDate` datetime DEFAULT NULL COMMENT '创建时间',**
46. **`modifyBy` bigint(20) DEFAULT NULL COMMENT '更新者',**
47. **`modifyDate` datetime DEFAULT NULL COMMENT '更新时间',**
48. **PRIMARY KEY (`id`)**
49. **) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci;**
50. **/\*Data for the table `smbms\_provider` \*/**
51. **insert into `smbms\_provider`(`id`,`proCode`,`proName`,`proDesc`,`proContact`,`proPhone`,`proAddress`,`proFax`,`createdBy`,`creationDate`,`modifyBy`,`modifyDate`) values (1,'111','北京','222','HAOha','13212345611','23a','2222',1,'2014-02-12 00:00:00',6,'2016-10-28 22:54:36'),(2,'22','是的','23','萨达','1212312','123123','213',3,'2014-02-12 00:00:00',NULL,NULL);**
52. **/\*Table structure for table `smbms\_user` \*/**
53. **DROP TABLE IF EXISTS `smbms\_user`;**
54. **CREATE TABLE `smbms\_user` (**
55. **`id` bigint(20) NOT NULL AUTO\_INCREMENT COMMENT '主键ID',**
56. **`userCode` varchar(20) COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '用户编码',**
57. **`userName` varchar(20) COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '用户名称',**
58. **`userPassword` varchar(20) COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '用户密码',**
59. **`gender` int(10) DEFAULT NULL COMMENT '性别（1 男 2 女）',**
60. **`birthday` date DEFAULT NULL COMMENT '出生日期',**
61. **`phone` varchar(20) COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '手机',**
62. **`address` varchar(20) COLLATE utf8\_unicode\_ci DEFAULT NULL COMMENT '地址',**
63. **`userType` int(10) DEFAULT NULL COMMENT '用户类型（1 系统管理员 2 经理 3 普通员工 ）',**
64. **`createdBy` bigint(20) DEFAULT NULL COMMENT '创建者（userId）',**
65. **`creationDate` datetime DEFAULT NULL COMMENT '创建时间',**
66. **`modifyBy` bigint(20) DEFAULT NULL COMMENT '更新者（userId）',**
67. **`modifyDate` datetime DEFAULT NULL COMMENT '更新时间',**
68. **PRIMARY KEY (`id`)**
69. **) ENGINE=InnoDB AUTO\_INCREMENT=13 DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci;**
70. **/\*Data for the table `smbms\_user` \*/**
71. **insert into `smbms\_user`(`id`,`userCode`,`userName`,`userPassword`,`gender`,`birthday`,`phone`,`address`,`userType`,`createdBy`,`creationDate`,`modifyBy`,`modifyDate`) values (2,'2','测试2','2',1,'1974-10-07','13523456789','那是的是第三代卡',1,1,'2016-10-17 19:33:02',NULL,NULL),(6,'1','人生大事','1',1,'2016-10-04','13523456789','',2,2,'2016-10-17 20:39:15',NULL,NULL),(9,'4','123','123456',2,'2013-10-15','13123456789','3216549887',2,6,'2016-10-18 23:06:25',6,'2016-10-18 23:08:38'),(10,'22','121212','323213',2,'2013-10-15','13123456789',NULL,1,2,'2016-10-18 23:06:25',NULL,NULL),(11,'2345','2231',NULL,2,'2013-10-15','13123456789',NULL,2,6,'2016-10-18 23:06:25',NULL,NULL),(12,'213','3213',NULL,1,'2013-10-15',NULL,NULL,2,6,'2016-10-18 23:06:25',NULL,NULL);**
72. **/\*!40101 SET SQL\_MODE=@OLD\_SQL\_MODE \*/;**
73. **/\*!40014 SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS \*/;**
74. **/\*!40014 SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS \*/;**
75. **/\*!40111 SET SQL\_NOTES=@OLD\_SQL\_NOTES \*/;**

**五．应用程序设计**

## 1. Servlet的详细应用

在Servlet工作流程中，根据web.xml中的配置将拦截所有通过servlet的请求，并根据URI决定分发给doGet还是doPost处理请求，然后调用相应的业务逻辑方法，调用相应的页面去执行响应。利用JDBC（Java Data Base Connectivity）技术处理与数据库连接，利用其Servlet API所定义的接口和类来封装与数据库交互的相关操作，使对数据库的操作变得相当容易，减少开发时间，而且能对系统进行灵活的设计和实现，使系统的各模块容易维护和重用。

## 2. 特定模块的详细介绍

每个包都有其各自的作用，是杂乱的java文件分层，具有层次感，也可以很好地进行维护。

### (1) 登陆模块

输入不同的角色名就会进入不同的模块，并完成相应的功能；登陆模块如图4-1所示



如图 4-1

输入正确的用户名和密码后进行登陆。



代码：**public** class HandelLogin extends HttpServlet{

UserOper oper=new UserOper();

**public** void init(ServletConfig config)throws ServletException{

super.init(config);

}

**public** void destroy() {

super.destroy();

}

**public** void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

this.doPost(request, response);

}

**public** void doPost(HttpServletRequest request,HttpServletResponse response)

throws ServletException,IOException{

request.setCharacterEncoding("utf-8");

response.setCharacterEncoding("utf-8");

response.setContentType("text/html;charset=utf-8");

HttpSession session=request.getSession(true);

String op = request.getParameter("op");

String opg= request.getParameter("opg");

String opyh = request.getParameter("opyh");

String zdgroup=request.getParameter("zdgroup");

System.out.println(zdgroup);

String gysgroup=request.getParameter("gysgroup");

Login judgeLogin = (Login)(session.getAttribute("login"));

**if** ("doLogin".equals(op)) {

doLogin(request, response);

**return**;

}

**if** (judgeLogin!=null&&!"".equals(judgeLogin)){

**if**("经理".equals(judgeLogin.getYh\_quanxian())){

**if** ("ZD".equals(op)){

doLnquiresZD(request,response);

}**else if**("ZDAdd".equals(op)){

doZDAdd(request,response);

}**else if**("ZDXinXi".equals(op)){

doZDXinXi(request,response);

}**else if**("修 改".equals(op)){

doZDXiuGai(request,response);

}**else if**("删 除".equals(op)){

doZDDel(request,response);

}**else if**("组合查询".equals(zdgroup)){

doZDChaXun(request,response);

}**else if**("GYS".equals(op)){

doLnquiresGYS(request,response);

}**else if**("GYSAdd".equals(op)){

doGYSAdd(request,response);

}**else if**("GYSXinXi".equals(op)){

doGYSXinXi(request,response);

}**else if**("修 改".equals(opg)){

doGYSXiuGai(request,response);

}**else if**("删 除".equals(opg)){

doGYSDel(request,response);

}**else if**("组合查询".equals(gysgroup)){

doGYSChaXun(request,response);

}**else if**("YH".equals(op)){

doLnquiresYH(request,response);

}**else if**("YHAdd".equals(op)){

doYHAdd(request,response);

}**else if**("YHXinXi".equals(op)){

doYHXinXi(request,response);

}**else if**("修 改".equals(opyh)){

doYHXiuGai(request,response);

}**else if**("删 除".equals(opyh)){

doYHDel(request,response);

}**else if** ("查询".equals(opyh)) {

doYhChaXun(request,response);

}**else if**("修改密码".equals(opyh)){

doYHMiMa(request,response);

}**else if**("修改密码".equals(op)){

doAdminMiMaXiuGai(request,response);

}

}

**if**("普通用户".equals(judgeLogin.getYh\_quanxian())){

**if** ("ZD".equals(op)){

doLnquiresZD(request,response);

}**else if**("ZDAdd".equals(op)){

doZDAdd(request,response);

}**else if**("ZDXinXi".equals(op)){

doZDXinXi(request,response);

}**else if**("修 改".equals(op)){

doZDXiuGai(request,response);

}**else if**("删 除".equals(op)){

doZDDel(request,response);

}**else if**("组合查询".equals(zdgroup)){

doZDChaXun(request,response);

}**else if**("GYS".equals(op)){

doLnquiresGYS(request,response);

}**else if**("GYSAdd".equals(op)){

doGYSAdd(request,response);

}**else if**("GYSXinXi".equals(op)){

doGYSXinXi(request,response);

}**else if**("修 改".equals(opg)){

doGYSXiuGai(request,response);

}**else if**("删 除".equals(opg)){

doGYSDel(request,response);

}**else if**("组合查询".equals(gysgroup)){

doGYSChaXun(request,response);

}**else if**("YH".equals(op)||"修改密码".equals(op)||"修改密码".equals(opyh)||

"查询".equals(opyh)||"删 除".equals(opyh)||"修 改".equals(opyh)||

"YHXinXi".equals(op)||"YHAdd".equals(op)){

Information info=new Information();

String msg="对不起您没有那么高的权限";

info.setMsg(msg);

request.setAttribute("info", info);

request.getRequestDispatcher("mainWindow.jsp").forward(request, response);

}

}

}else{

response.sendRedirect("login.jsp");

}

}

管理员（用户）登陆以后就会进入总页面如图：4-2所示：在左侧可以员进来后

（普通用户）各自的权限。

管理员登陆：



普通用户登陆：



点击用户账号链接可以查询该用户全部信息，子功能有修改、删除、重置密码、返回上个页面。



### (2)供应商管理模块

管理员权限下的功能操作及代码

如图：4-6所示



如图 4-6

代码：

**private** **void** doGYSChaXun(HttpServletRequest request,

HttpServletResponse response) **throws** ServletException,IOException{

String gys\_name=request.getParameter("commodityname").trim();

String gys\_miaoshu=request.getParameter("commoditymiaoshu").trim();

List<GYSBean> list=oper.GYSChaXun(gys\_name,gys\_miaoshu);

**if**(list!=**null**){

request.setAttribute("list", list);

request.getRequestDispatcher("admin\_supplier\_list.jsp").forward(request, response);

}**else** {

response.sendRedirect("admin\_supplier\_list.jsp");

}

}

**private** **void** doLnquiresGYS(HttpServletRequest request,

HttpServletResponse response)**throws** ServletException,IOException {

List<GYSBean> list=oper.GYSLnquires();

**if**(list!=**null**){

request.setAttribute("list", list);

request.getRequestDispatcher("admin\_supplier\_list.jsp").forward(request, response);

}**else** {

response.sendRedirect("admin\_supplier\_list.jsp");

}

}



如图 4-7

代码：

**private** **void** doGYSAdd(HttpServletRequest request,

HttpServletResponse response) **throws** ServletException,IOException{

String gys\_id = request.getParameter("gysid").trim();

String gys\_name =request.getParameter("gysname").trim();

String gys\_miaoshu=request.getParameter("gysmiaoshu").trim();

String gys\_lianxiren=request.getParameter("gyslianxi").trim();

String gys\_phone=request.getParameter("gysphone").trim();

String gys\_chuanzhen=request.getParameter("gyschuanzhen").trim();

String gys\_address=request.getParameter("gysaddress").trim();

List params = **new** ArrayList();

params.add(gys\_id);

params.add(gys\_name);

params.add(gys\_miaoshu);

params.add(gys\_lianxiren);

params.add(gys\_phone);

params.add(gys\_chuanzhen);

params.add(gys\_address);

Information info=oper.GYSAdd(params);

request.setAttribute("info", info);

request.getRequestDispatcher("admin\_supplier\_add.jsp").forward(request, response);

}



如图 4-8

代码：

**private** **void** doGYSXinXi(HttpServletRequest request,

HttpServletResponse response)**throws** ServletException,IOException {

String gys\_id = request.getParameter("id");

GYSBean gysbean=**new** GYSBean();

gysbean = oper.GYSXinXi(gys\_id);

request.setAttribute("gysbean", gysbean);

request.getRequestDispatcher("admin\_supplier\_update.jsp").forward(request, response);

}



如图 4-9

代码：

**private** **void** doGYSXiuGai(HttpServletRequest request,

HttpServletResponse response) **throws** ServletException,IOException{

String gys\_id= request.getParameter("gysid").trim();

String gys\_miaoshu = request.getParameter("gysmiaoshu").trim();

String gys\_lianxiren=request.getParameter("gyslianxi").trim();

String gys\_phone=request.getParameter("gysphone").trim();

String gys\_chuanzhen=request.getParameter("gyschuanzhen").trim();

String gys\_address=request.getParameter("gysaddress").trim();

String gys\_name=request.getParameter("gysname").trim();

List params=**new** ArrayList();

params.add(gys\_id);

params.add(gys\_miaoshu);

params.add(gys\_lianxiren);

params.add(gys\_phone);

params.add(gys\_chuanzhen);

params.add(gys\_address);

params.add(gys\_name);

Information info=oper.GYSXiuGai(params);

request.setAttribute("info", info);

request.getRequestDispatcher("admin\_supplier\_update.jsp").forward(request, response);

}

**private** **void** doGYSDel(HttpServletRequest request,

HttpServletResponse response) **throws** ServletException,IOException{

String gys\_id= request.getParameter("gysid").trim();

Information info=oper.GYSDel(gys\_id);

request.setAttribute("info", info);

request.getRequestDispatcher("admin\_supplier\_update.jsp").forward(request, response);

}

**六．附：页面前端程序代码**

1. **登陆入口页面**

|  |
| --- |
|  |
| <html> |
|  | <head> |
|  | <title>超市账单管理系统-欢迎页</title> |
|  | <meta http-equiv="Content-Type" content="text/html; charset=utf-8" /> |
|  | <meta http-equiv="pragma" content="no-cache"> |
|  | <meta http-equiv="cache-control" content="no-cache"> |
|  | <meta http-equiv="expires" content="0"> |
|  |  |
|  | <!-- |
|  | <link rel="stylesheet" type="text/css" href="styles.css"> |
|  | --> |
|  | </head> |
|  | <body> |
|  | 欢迎使用超市账单管理系统 |
|  | </body> |
|  | </html> |
|  |  |

1. **登陆页面**

|  |  |
| --- | --- |
|  | <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"> |
|  | <html> |
|  | <head> |
|  | <meta http-equiv="Content-Type" content="text/html; charset=utf-8" /> |
|  | <title>系统登录 - 超市账单管理系统</title> |
|  | <link type="text/css" rel="stylesheet" href="[css/style.css](file:///D:\\%E7%B4%A0%E6%9D%90\\%E8%B6%85%E5%B8%82%E8%B4%A6%E5%8D%95%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F\\css\\style.css" \t "_blank)" /> |
|  | <script type="text/javascript"> |
|  | function validate(){ |
|  | window.location.href="frame.html" |
|  | } |
|  | </script> |
|  | </head> |
|  | <body class="blue-style"> |
|  | <div id="login"> |
|  | <div class="icon"></div> |
|  | <div class="login-box"> |
|  | <form action="" name="actionForm" id="actionForm" method="post" > |
|  | <dl> |
|  | <dt>用户名：</dt> |
|  | <dd><input type="text" class="input-text" /> <span id="loginNameSpan"></span></dd> |
|  | <dt>密　码：</dt> |
|  | <dd><input type="password" class="input-text" /><span id="passwordSpan"></span></dd> |
|  | </dl> |
|  | <div class="buttons"> |
|  | <input type="button" value="登录系统" class="input-button" onclick="validate();" /> |
|  | <input type="reset" value="重　　填" class="input-button" /> |
|  | </div> |
|  | </form> |
|  | </div> |
|  | </div> |
|  | </body> |
|  | </html> |

1. **系统主界面**

|  |  |
| --- | --- |
|  | <html> |
|  | <head> |
|  | <style> |
|  | \* {padding:0px; margin:0px;height:100%; width:100%;} |
|  | .content {padding-top:100px;box-sizing:border-box} |
|  | .top {height:100px; width:100%;position:absolute; top:0px;} |
|  | .left{ float:left;width:20%;} |
|  | .right{ float:left;width:80%;} |
|  | #top{height:100px;} |
|  | </style> |
|  | </head> |
|  | <body> |
|  | <div class="content"> |
|  | <div class="top"> |
|  | <iframe src="[top.html](file:///D:\\%E7%B4%A0%E6%9D%90\\%E8%B6%85%E5%B8%82%E8%B4%A6%E5%8D%95%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F\\top.html" \t "_blank)" scrolling="No" id="top" noresize="noresize"> |
|  | </iframe> |
|  | </div> |
|  | <div class="main"> |
|  | <div class="left"> |
|  | <iframe src="[left.html](file:///D:\\%E7%B4%A0%E6%9D%90\\%E8%B6%85%E5%B8%82%E8%B4%A6%E5%8D%95%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F\\left.html" \t "_blank)" scrolling="No" id="left" noresize="noresize"> |
|  | </iframe> |
|  | </div> |
|  | <div class="right"> |
|  | <iframe src="[main.html](file:///D:\\%E7%B4%A0%E6%9D%90\\%E8%B6%85%E5%B8%82%E8%B4%A6%E5%8D%95%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F\\main.html" \t "_blank)" id="mainFrame" name="mainFrame" noresize="noresize" > |
|  | </iframe> |
|  | </div> |
|  | </div> |
|  | </div> |
|  |  |
|  | </body> |
|  | </html> |

1. **修改密码**

|  |  |
| --- | --- |
|  | <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"> |
|  | <html> |
|  | <meta http-equiv="Content-Type" content="text/html; charset=utf-8" /> |
|  | <title>Insert title here</title> |
|  | <link type="text/css" rel="stylesheet" href="[css/style.css](file:///D:\\%E7%B4%A0%E6%9D%90\\%E8%B6%85%E5%B8%82%E8%B4%A6%E5%8D%95%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F\\css\\style.css" \t "_blank)"> |
|  | <script type="text/javascript" src="[js/jquery-1.8.3.min.js](file:///D:\\%E7%B4%A0%E6%9D%90\\%E8%B6%85%E5%B8%82%E8%B4%A6%E5%8D%95%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F\\js\\jquery-1.8.3.min.js" \t "_blank)"></script> |
|  | </head> |
|  | <body> |
|  | <div class="main"> |
|  | <div class="optitle clearfix"> |
|  | <div class="title">密码修改&gt;&gt;</div> |
|  | </div> |
|  | <div class="content"> |
|  | <table class="box"> |
|  | <input type="hidden" id="userId" value="${userInfo.userId}"/> |
|  | <tr> |
|  | <td class="field">旧密码：</td> |
|  | <td><input type="password" name="oldPassword" class="text" id="oldPassword" /> <font color="red">\*</font><font color="red" id="password\_span"></font></td> |
|  | </tr> |
|  | <tr> |
|  | <td class="field">新密码：</td> |
|  | <td><input type="password" name="newPassword" class="text" id="newPassword" /> <font color="red">\*</font></td> |
|  | </tr> |
|  | </table> |
|  | </div> |
|  | <div class="buttons"> |
|  | <input type="button" id="update" value="修改" class="input-button" /> |
|  | </div> |
|  | </div> |
|  | <script type="text/javascript" src="[js/user/updatePwd.js](file:///D:\\%E7%B4%A0%E6%9D%90\\%E8%B6%85%E5%B8%82%E8%B4%A6%E5%8D%95%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F\\js\\user\\updatePwd.js" \t "_blank)"></script> |
|  | </body> |
|  | </html> |
|  |  |

**七．结束语**

通过java语言程序设计基本实现了超市管理系统，本系统完成了用户登录注册页面的功能以及供应商管理的一些信息，但实用性效果还有待改进，希望在以后的学习中能及时知道自己存在的不足，对一些不会的只是及时查漏补缺，才可以。