目 录

[一、 物流管理系统 1](#_Toc3460)

[二、 设计目的 3](#_Toc1701)

[1、 初衷 3](#_Toc4294)

[2、 结合实际 3](#_Toc12036)

[3、 使用工具 3](#_Toc1756)

[4、 使用数据库以及数据表展示 4](#_Toc28237)

[三、 总体设计 6](#_Toc5885)

[1、登录模块 6](#_Toc15384)

[2、 管理系统功能概述 6](#_Toc31410)

[3、 功能框架图 7](#_Toc21202)

[四、 重要模块详细设计 9](#_Toc30108)

[1、登录界面设计 9](#_Toc5875)

[2、物流管理系统总界面 1](#_Toc13374)0

[3、基本管理界面 1](#_Toc15640)0

[4、出货入货管理界面 12](#_Toc13374)

[3、查询视图界面 1](#_Toc15640)4

[五、 总结及收获体会 1](#_Toc31563)5

[六、 代码列表 1](#_Toc31563)7

# 题目：物流管理系统

## 一、选题

### 基于Spring mvc,Spring,Hibernat的物流管理系统

## 设计目的

### （1）初衷

解决国内的很多企业，由于信息的记录、收集、汇总、处理分析还处于比较低的水平，记录中信息分散、格式随意、应用单一，收集中不及时、不全面、不完善、保留不科学，汇总时口径不一、主题多元、结果不一，处理分析方法落后、效率低下、只有短期，难以支持趋势的问题。

### 结合实际

论文首先研究了系统开发的可行性，然后进行了需求分析，并将需求分析以文档的形式反馈给用户，并和用户进行交流进一步确认需求，然后根据需求分析采用了结构化分析方法进行了系统分析设计，并进行了数据库的建模工作，根据模型构建了数据库。后台数据库采用MYSQL5.7完成。基于开放的开发架构Java，实现了系统登录模块、操作员管理模块、供应商管理模块、入库模块、出库模块等。

### 使用工具

IDEA 2017.2.1

SQLyog

Mysql 5.7

Jquery 1.5.2

Easyui 1.5

### 使用数据库以及数据表展示

#### 货物信息表

货物信息表主要用于保存货物信息，包括货物编号id、货物名称name、货物价格price、货物类型type、货物数量quantity、货物供应商编号supid等几个字段。在该表中，设置id一列为主键。此表由货物信息的系统管理人员来对其进行修改添加和删除等操作并对其进行维护。

货物信息表ProductData的设计如表3.1所示。

表3.1 货物信息表ProductData

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 名称 | 数据类型 | 长度 | 允许空 | 是否为主键 | 备注 |
| id | int | 4 | 否 | 是 | 商品编号 |
| name | varchar | 50 | 是 | 否 | 商品名称 |
| price | float | 8 | 是 | 否 | 价格 |
| type | varchar | 50 | 是 | 否 | 类型 |
| quantity | float | 8 | 是 | 否 | 数量 |
| supId | int | 4 | 是 | 否 | 供应商编号 |

#### 供应商信息表

供应商信息表主要用于保存供应商的各种信息，包括供应商编号supplierid，供应商名称suppliername、供应商地址supplieraddress、供应商邮编suppliercode、供应商电话suppliertelephone、供应商传真supplierfax、供应商联系人supplierrelationer、供应商电子邮箱supplieremail 等几个字段，在该表中，设置供应商编号supplierid一列为主键。此表由系统管理员来实现供应商的修改信息，添加和删除供应商的操作，此表由系统操作员来进行直接控制和维护。

供应商信息表SupplierData的设计如表3.2所示。

表3.2 供应商信息表 SupplierData

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 名称 | 数据类型 | 长度 | 允许空 | 是否为主键 | 备注 |
| Supplierid | Int | 4 | 否 | 是 | 供应商编号 |
| Suppliername | varcher | 50 | 是 | 否 | 名称 |
| Supplieraddress | Varcher | 50 | 是 | 否 | 地址 |
| Postcode | Varcher | 50 | 是 | 否 | 邮编 |
| Suppliertelephone | Varcher | 50 | 是 | 否 | 电话 |
| Supplierfax | Varcher | 50 | 是 | 否 | 传真 |
| Supplierrelationer | Varcher | 50 | 是 | 否 | 联系人 |
| supplieremail | varcher | 50 | 是 | 否 | Email |

#### 操作员信息表

操作员信息表主要用于保存系统管理人员的信息，包括操作员编号userid，操作员姓名username、操作员密码userpassword、操作员类型userrole等字段，在该表中，设置操作员编号Userid一列为主键。此表只限于操作员进行操作和维护。

操作员信息表UserData的设计如表3.3所示。

表3.3 操作员信息表UserData

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 名称 | 数据类型 | 长度 | 允许空 | 是否为主键 | 备注 |
| Userid | Int | 4 | 否 | 是 | 操作员编号 |
| realName | varchar | 50 | 否 | 否 | 真实姓名 |
| Username | Varchar | 50 | 是 | 否 | 用户名 |
| Userpassword | varchar | 50 | 是 | 否 | 密码 |
| userrole | int | 4 | 是 | 否 | 用户类型 |

## 总体设计

### 总体需求

通过对物流企业的调查，要求物流管理系统需要实现以下目标：

·能够对货物进行系统管理，进行出库入库操作，能够浏览货物信息，按不同条件快速方便的查找出所需要的货物信息；

·能够对货物进行清理，及时删除多余的货物信息，同时能够添加新的货物；

·对数据库信息进行整体的数据导入导出；

·能够对供应商进行管理，包括添加该供应商、删除供应商和修改供应商信息；

·对操作员的信息管理，包括添加操作员、删除操作员和修改操作员信息；

#### 1．登陆模块

在物流管理系统中，登陆模块是非常重要的，它是整个物流管理系统的入口，只有通过了此登陆界面的认可，才能够进入物流管理系统，正确的执行物流管理的功能。登陆部分实现对企业员工登陆的信息验证，只有在数据库中已经存储的操作员才能够进行信息验证登陆，从而为用户安全性提供保障。同时登陆模块也是对操作员的一个标识，一旦登陆了系统，系统自动调用该操作员的信息，从而自动划定了该操作员只能够执行自己的权限内所能够执行的操作，例如，能否执行删除操作员的操作。登陆模块的流程图如图2.2所示。

#### 2．管理系统总功能概述

从物流管理系统主界面可以进入物流管理系统的全部分支功能子页面。如图2.3的系统框图，本系统大体分为五个模块，分别是：基本管理、进货出货管理、查询视图、操作员系统管理。

在基本管理中，包括供应商管理和商品管理两部分。供应商管理部分具有添加供应商、修改供应商、删除供应商的功能。通过该模块，可以对供应商的ID、邮编、姓名、地址、电话、传真、联系人、E-Mail等信息进行管理，并将管理后的数据存入系统，该模块是企业与客户联系的通道，管理着所有的客户信息，是整个企业中至关重要的一部分；商品管理部分具有添加商品、修改商品、删除商品的功能。通过该模块，可以对商品的ID、名称、价格、种类、数量、供应商等信息进行管理，并将管理后的数据存入系统。该模块是企业物流管理的主要部分，只要保证该模块准确无误的工作，才能够保障企业的正常运作，是对一个企业信誉度的考验。

进货出货管理主要包括商品入库和商品出库两个功能。入库功能主要是实现登记入库记录的功能。这些信息应包括入库商品信息、入库数量信息等。入库登记功能产生货物的入库记录，为以后进行入库货物的查询和生成报表以及出库操作产生操作数据，供管理人员掌握仓库内存储的货物情况。出库功能是仓库管理系统的基本功能之二。出库功能主要是实现登记出库货物信息的功能。这些出库记录包括出库商品信息、出库数量信息等。出库功能依据入库登记功能产生的入库记录产生货物的出库记录，将出库记录存入系统，为以后的出库记录查询以及掌握仓库货物存储情况作基础。

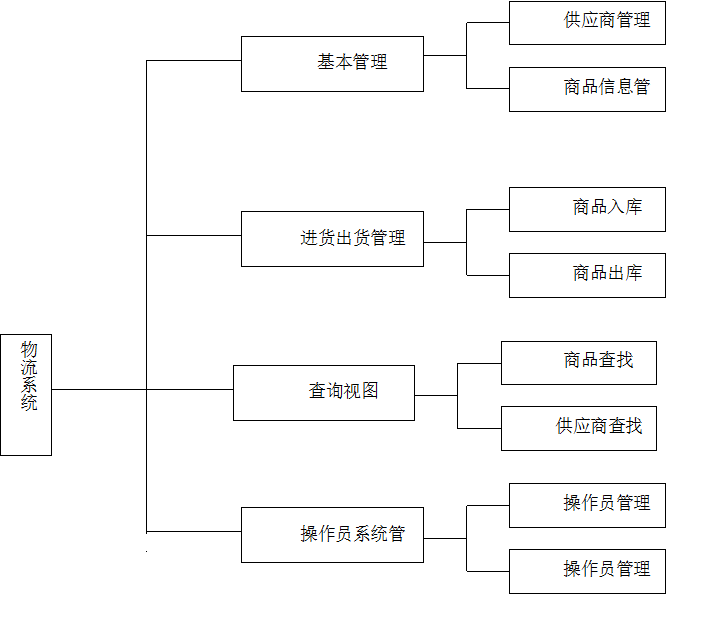
在查询视图中，包括供应商视图查找和商品视图查找两部分，供应商视图查找部分具有汇总供应商的功能，可以根据ID查出供应商信息，便于管理者方便快捷的了解供应商的信息，也可以显示所有供应商，使操作员能够掌握数据库中所有供应商的总信息，以便对物流系统进行及时的更新和添加。商品视图查找部分具有汇总商品的功能，可以根据商品的id号、价格、供应商等不同的条件查找出所需商品的信息并显示出来，使对商品的管理一目了然。该模块同时可以进行数据的总体的导入导出，让系统的操作更加快捷和方便，为物流服务提高效率。

操作员系统管理部分主要功能是对操作员的信息进行管理，具有添加操作员、删除操作员、修改操作员信息的功能。该模块可以及时删除无效的操作员信息，确保及时清理与系统无关人员，防止信息的泄漏，增加了系统安全性。

#### 3、功能框架图

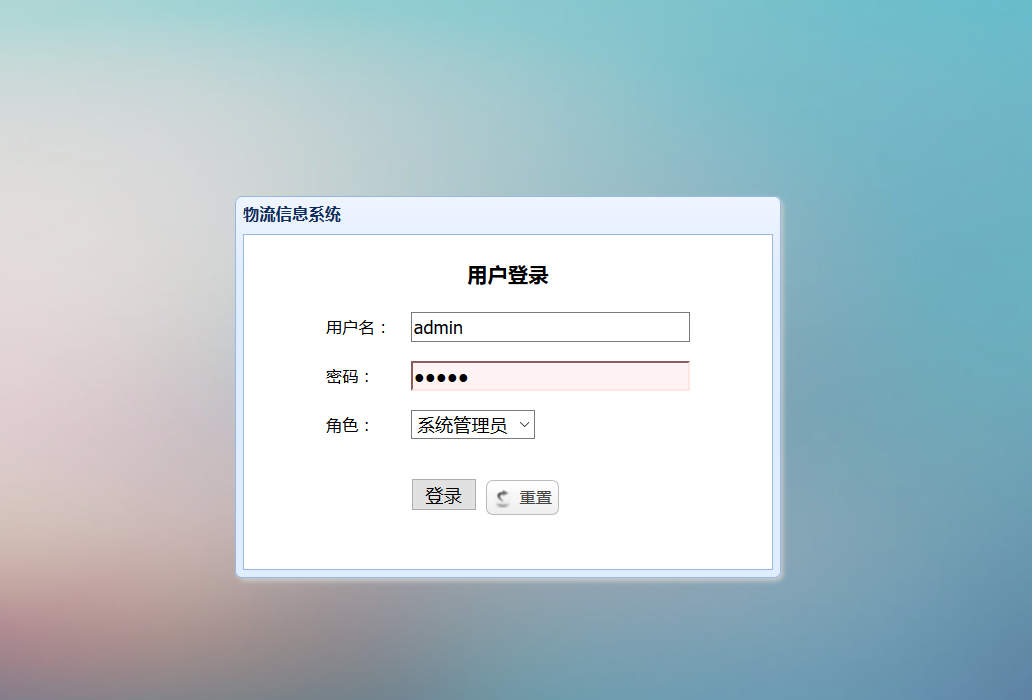
本系统的核心模块：物流管理系统总界面的总体框图如图2.3所示。

本系统属于树形结构，这种结构的优点就是使系统结构清晰，层次明朗，分支明确。运用树形结构，可以在不减少系统功能的基础上使系统的总界面更加简明，使每个界面的归属性更加确切，从而增加了系统的可操作性，大大的简化了操作员的工作。



## 重要模块详细设计

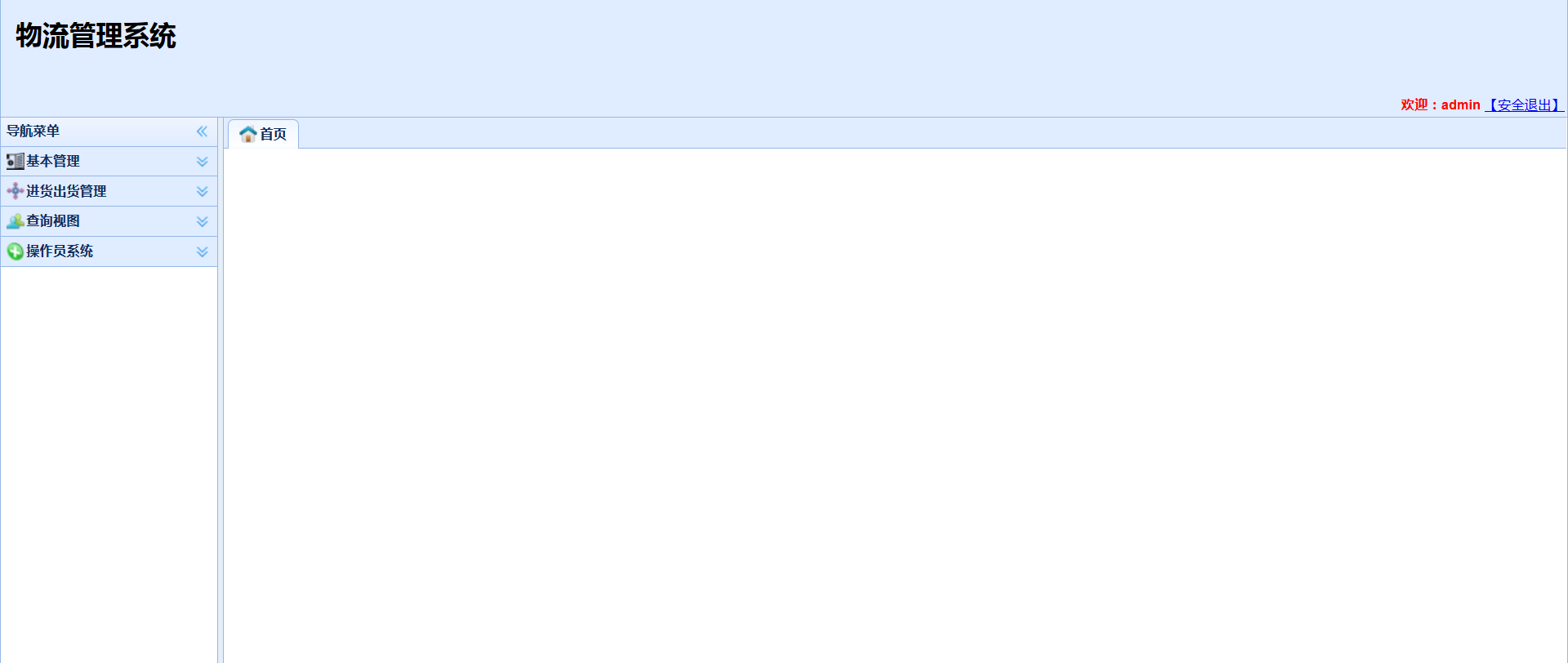
### 4.1登陆界面设计



通过输入用户名和密码及选择正确的角色进行登录，若用户名或密码或角色未选择正确则返回登录页面，若正确则进入物理管理总界面，并根据角色开放一定权限。



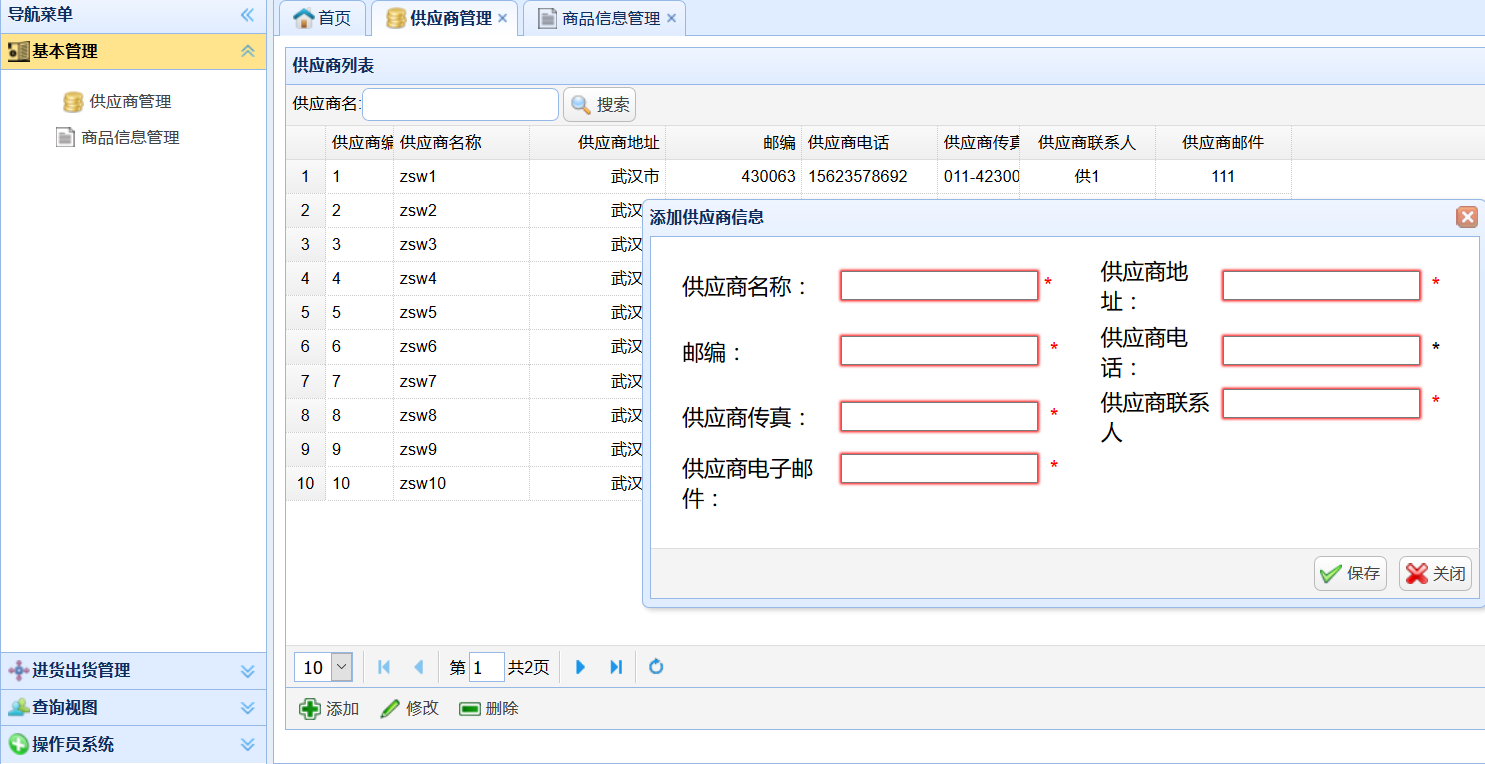
### 4.2物流管理系统总界面



物流管理系统总界面是对总体功能的汇总 进行各类操作

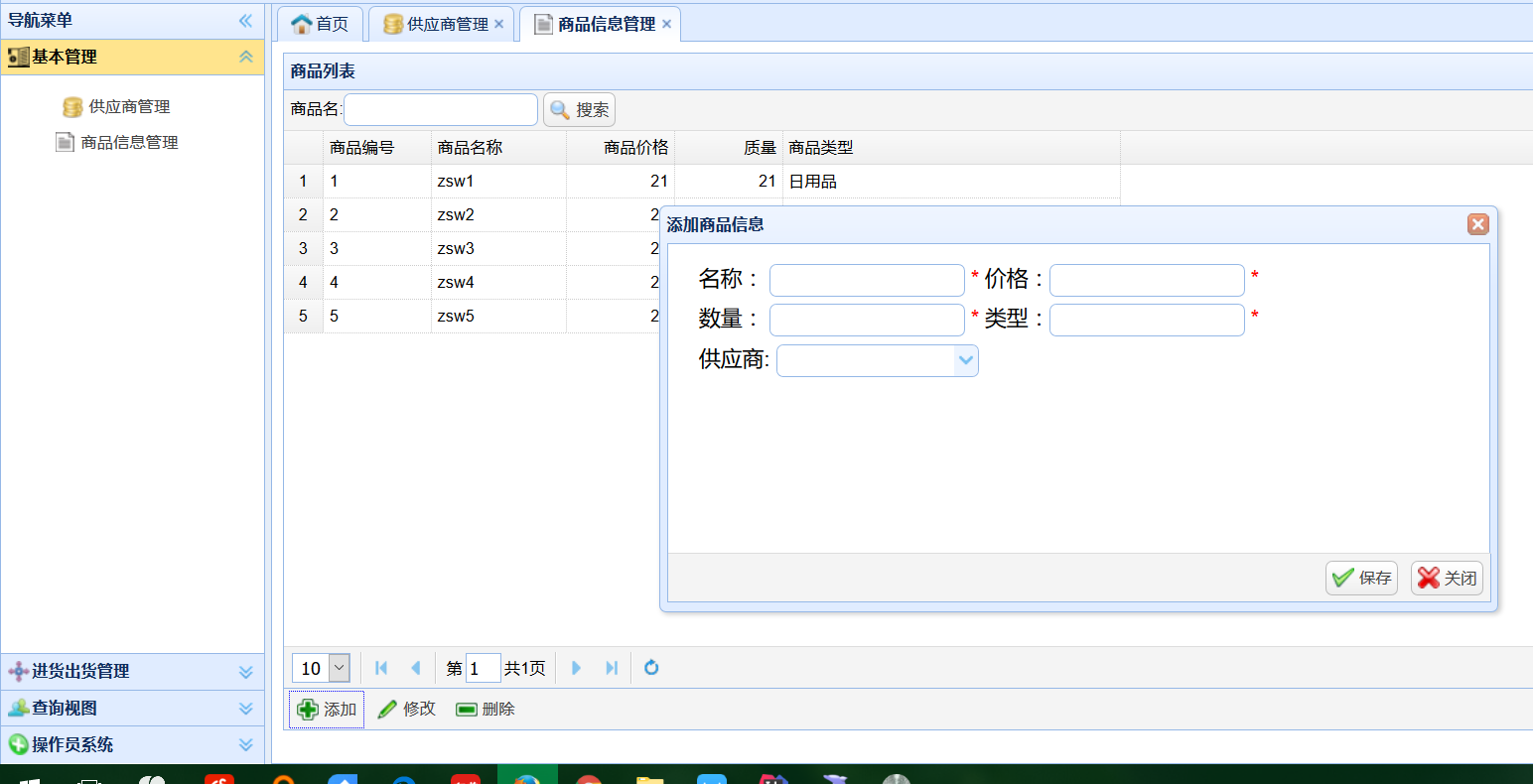
### 4.3基本管理界面

#### 4.3.1 供应商管理选项卡



可以对供应商进行添加 删除 查找 修改

#### 4.3.2 商品信息管理界面



可以对商品进行添加 删除 查找 修改

### 4.4进货出货管理界面

#### 4.4.1 商品出库



通过选择动态载入的商品，点击核实获得该商品实时的仓库数量，在进库量面板填入数目 点击金库 使数目减少达到出库的效果

清空按钮则清空出库量输入框

#### 4.4.2 商品入库

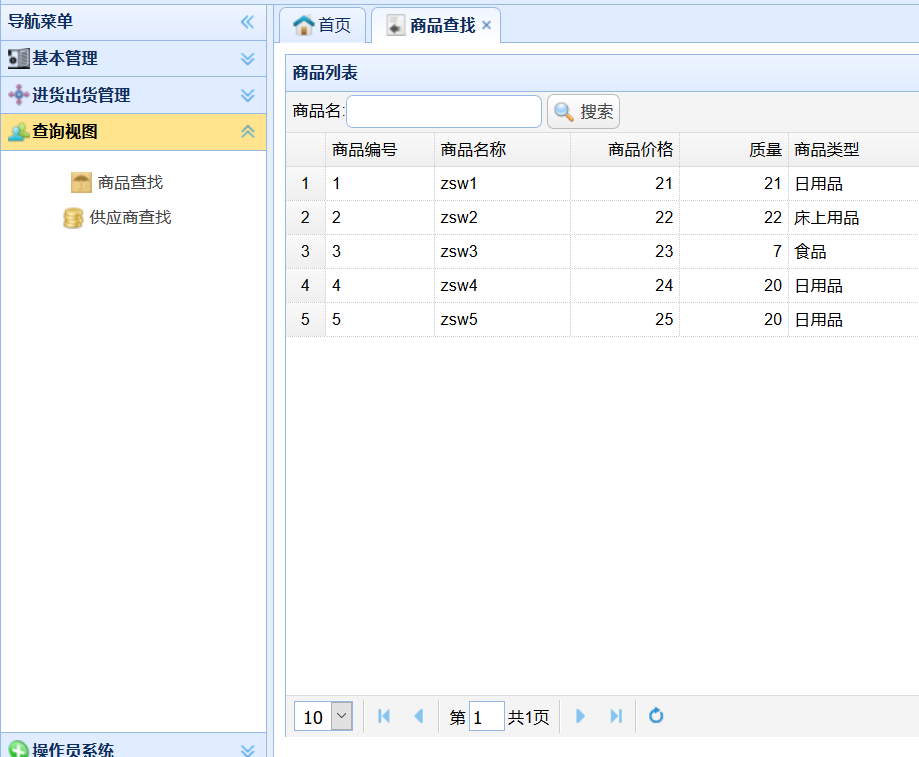


通过选择动态载入的商品，点击核实获得该商品实时的仓库数量，在进库量面板填入数目 点击金库 使数目增加 达到入库的效果

清空按钮则清空进库量输入框

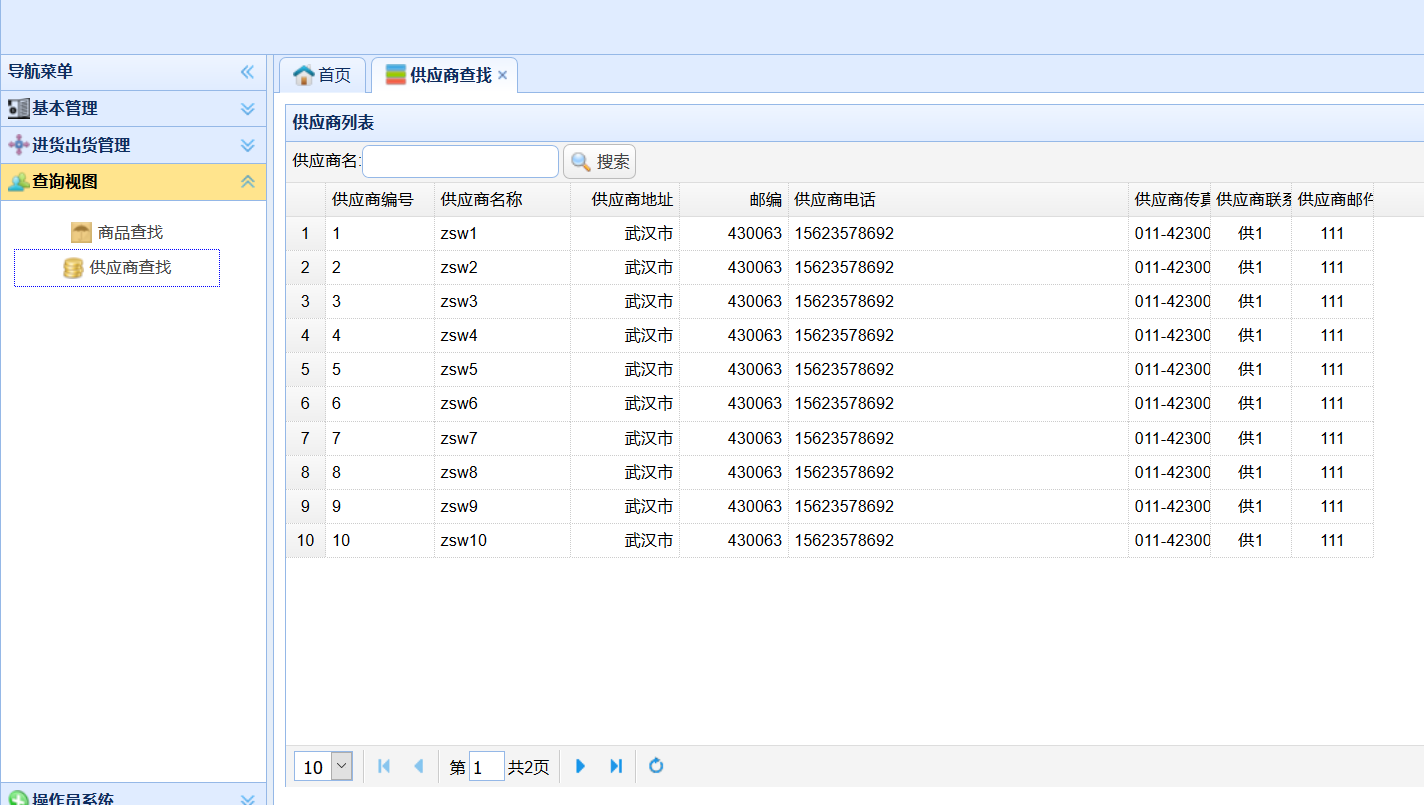
### 4.5查询视图界面

#### 4.5.1 商品视图



通过商品名称搜索相关商品信息，达到商品查找功能

#### 4.5.2 供应商视图



通过供应商搜索相关商品信息，达到商品查找功能

## 五、总结及收获体会

经过了一个学期的努力，在老师和同学的大力支持和帮助下，我在规定时间内完成了此次课程设计的任务“基于Java语言的物流信息系统”。

通过此次课程设计，使我们摆脱了单纯理论知识的学习状态，理论和实际的结合，锻炼了我解决实际设计中遇到问题的能力，也更加深入了我们对所学的基础知识的掌握，深深地体会到了课程设计对理论知识促进与提高和对实践的具体应用。

选择设计题目很关键，经过我与老师的讨论与协商，在老师的精心研究下，最终确定了基于Java的物流信息管理系统。之后在收集资料过程中，我去学校图书馆查阅资料，不懂得地方去上网查询下载，为了增强设计的实际应用性，我还去了几家物流公司进行实地的考察，对物流信息的管理进行了需求分析和系统了解，虽然期间查询到的资料众多而且复杂，但我通过有条理有顺序的梳理，最终克服知识的琐碎繁杂性了。

从最开始的不适应到将程序顺利的完成，是需要我付出努力的过程，同时要有百折不挠的精神。这期间让我懂得了许多在课堂上学不懂，学不透的知识，同时让我感受到理论知识的重要性。开始时，我处于对很多相关知识不是很熟悉甚至不知道的状态，便开始研究和查看大量的相关资料和书籍，让自己头脑中模糊的概念逐渐清晰，使自己的作品一步步完善起来，每一次改进都是我学习的收获。从中我也充分认识到了面向对象Java程序设计给我的学习带来了很大的帮助，也让我学到了很多团结合作的经验，同时锻炼了我的整体思维的能力。有的时候把自己的新想法与同组的其他人分享，大家往往会提出置疑，但也就是在这种争论中才使我们的想法不断的提高、成熟。

通过对整体的设计，对局部的取舍，以及对细节的斟酌处理，都使我的能力得到了锻炼，经验得到了丰富，并且意志品质力、抗压能力以及耐力也都得到了不同程度的提升，我想这就是我们课程设计要达到的目的。

经过长时间的艰辛努力最终换来了丰硕的成果，虽然有些不完善的地方，但毕竟是自己花费大量的时间设计出来的，也感到有很大的欣慰。可以肯定地说现在的我对程序设计这一块也有了一定的了解，但我深知自己所了解的不过仅仅是皮毛而已，离真正的运用自如还相距甚远。

学无止境，学习生活无穷尽，只有掌握更多的、更新的知识才能使自己始终立于不败之地。如果只为现在所取得的成绩而沉浸于喜悦、骄傲之中的话，那么，我们就不会前进，只能停留在原来的地方踏步，甚至还会倒退。所以从毕业设计所暴露出的问题入手，不断努力完善自己，哪里欠缺，哪里补足。

## 附:代码视图列表

