

淮阴师范学院

毕 业 论 文

学生姓名 _____ 学 号 _____

学 院 _____ 计算机科学与技术学院

专 业 _____ 计算机科学与技术学院（软件工程）

题 目 _____ 基于 JSP 的仓库管理系统

_____ 的设计与实现

摘要：仓库管理系统是一种企业不可或缺的部分，高效的仓库管理系统可认为企业的发展 and 经营提供坚强的后盾。伴随科技的不停发展，用计算机对仓库进行管理，具有人工管理所无法比拟的长处，例如：检索迅速、查找以便、保密性好、成本低、寿命长等。这些长处能极大地提高仓库管理的效率，使企业的仓库管理井井有条，为企业的健康发展发明良好的条件。本文在分析既有管理系统开发技术的基础上，实现了以 Eclipse 为开发平台的基于 JSP 的，以 MySQL 作为后台数据库的仓库管理系统。该系统重要具有基本信息管理、库存管理、信息查询、顾客管理等功能模块。

关键词：仓库管理，JSP，MySQL 数据库

Abstract: Warehouse management system is an indispensable part of the enterprise. A warehouse management system with high efficiency can provide a strong backing for the development and management of enterprises. With the development of technology, to manage the warehouse with computer has the incomparable advantages of manual management. For example, quick search, find convenient, good security, low cost, long service life etc. These advantages can greatly improve the efficiency of warehouse management, the enterprise storage management be arranged in good order, and create good conditions for the healthy development of enterprises. In this paper, based on analysis of the existing management system development technology, in order to realize the Eclipse as the development platform based on JSP, using MySQL as the background database warehouse management system. The system mainly has the basic information management, inventory management, information query, user management module

Keywords: Warehouse management, JSP, MySQL database

目 录

1 绪论.....	4
1.1 开发背景	4
1.2 开发意义	4
1.3 本文组织构造	4
2 系统处理方案.....	5
2.1 JSP 语言概述.....	5
2.2 SSH 框架.....	5
2.3 MySql 数据库.....	5
2.4 系统平台环境	5
3 需求分析.....	6
3.1 面向的顾客人群	6
3.2 功能需求分析	6
3.3 业务流程图	7
4 数据库设计.....	9
4.1 E-R 模型.....	9
4.2 数据模型	12

5 详细设计.....	15
5.1 数据库连接类	15
5.2 系统重要模块	16
结 论	34
参 考 文 献	35
致 谢	36

1 绪论

1.1 开发背景

一直以来企业使用老式人工的方式来管理文献库存,这种管理方式存在许多缺陷,如:保密性差、效率低等,并且时间一长,将产生大量的数据和文献,这对于企业查找、更新和维护都带来了诸多的困难。由于缺乏科学的管理和管理工具,企业在业务上和管理上的安排都存在着许多不便。因而针对这些问题而开发的系统具有很好的前景。

1.2 开发意义

伴随时代的不停发展,社会正在不停向信息化时代前进。由于可以有效地搜集和处理多种信息,提高办事效率,电子自动化办公越来越被重视。而伴随互联网技术的不停发展,又为电子自动化办公的发展提供了良好的前景。如今,诸多企业已经拥有了计算机设备,具有物质条件,但诸多工作仍由管理者手工操作完毕,这就导致不必要的挥霍。计算机流行的原因重要有如下几种方面:

1. 计算机可以替代手工操作进行许多繁杂的工作;

2. 计算机可以节省大量资源；
3. 计算机可以极大地提高人们的工作效率；
4. 计算机可以使敏感文档愈加安全；
5. 电子自动化办公是先进生产力发展的重要标志。

综上所述，用计算机开发的软件系统，替代人工操作进行管理，可以节省大量的人力、物力，是企业发展的必然趋势^[1]！

1.3 本文组织构造

本系统将以书本和某些课外书籍为参照，从系统开发背景→需求分析→概要设计→详细设计→详细开发一步步对系统进行分析和设计。各个章节安排如下：

第一章为绪论，简述项目的开发背景和开发意义；

第二章详细简介了项目开发的应用技术，例如 JSP，框架和数据库等；

第三章是需求分析，简介了系统的设计目的和系统的重要功能模块。

第四章是数据库设计，在 E-R 模型的基础上，再设计出对应的逻辑模型。

第五章根据系统功能模块详细简介了各模块的设计过程并给出部分实现代码。

2 系统处理方案

2.1 JSP 语言概述

JSP (Java Server Pages)是由 Sun Microsystems 企业倡导、许多企业参与一起建立的一种动态网页技术原则。它的重要目的是将表达逻辑从 Servlet 中分离出来。它是在老式的网页 HTML 文献 (*.htm, *.html)中插入 Java 程序段 (Scriptlet) 和 JSP 标识 (tag)，从而形成 JSP 文献 (*.jsp)。

所有程序操作都在服务器端执行，网络上传送给客户端的仅是得到的成果，这样大大减少了对客户浏览器的规定，虽然客户浏览器端不支持 Java，也可以访问 JSP 网页。用 JSP 开发的 Web 应用是跨平台的，既能在 Linux 下运行，也能在其他操作系统上运行^[2]。

2.2 SSH 框架

SSH 在 J2EE 项目中表达了 3 种框架，即 Spring + Struts +Hibernate。本系统采用 MVC 开发模式来编写代码，MVC 全名是 Model View Controller，是模型(model)－视图(view)－控制器(controller)的缩写^[3]。Struts 对 Model，View 和 Controller 都提供了对应的组件。Spring 是一种轻量级的控制反转 (IoC) 和面向切面 (AOP) 的容器框架，它由 Rod Johnson 创立。它是为了处理企业应用开发的复杂性而创立的^[4]。Spring 使用基本的 JavaBean 来完毕此前只也许由 EJB 完毕的事情。Hibernate 是一种开放源代码的对象关系映射框架，它对 JDBC 进行了非常轻量级的对象封装，可以应用在任何使用 JDBC 的场所，可以在 Servlet/JSP 的 Web 应用中使用，也可以在应用 EJB 的 J2EE 架构中取代 CMP，完毕数据持久化的重任^[5]。

2.3 MySql 数据库

MySQL 是一种关系型数据库管理系统，由瑞典 MySQL AB 企业开发，目前属于 Oracle 企业。MySQL 是一种关联数据库管理系统，关联数据库将数据保留在不一样的表中，而不是将所有数据放在一种大仓库内，这样就增长了速度并提高了灵活性。MySQL 的 SQL 语言是用于访问数据库的最常用原则化语言。由于其体积小、速度快、总体拥有成本低，尤其是开放源码这一特点，一般中小型网站的开发都选择 MySQL 作为网站数据库^[6]。

2.4 系统平台环境

(1) 硬件环境：

服务器：

CPU： 2G 或以上；

内存： 1G RAM 或以上；

硬盘： 80G 或以上；

客户端：

CPU： 1G 或以上；

内存： 64MB 以上内存或更高；

硬盘： 20G 或以上。

(2) 软件开发环境：

操作系统： WindowsXP；

所用工具： Eclipse, Navicat for MySQL；

数据库： MySQL。

3 需求分析

3.1 面向的顾客人群

1. 管理员：作为系统的管理员和维护者，对系统基本信息、仓库管理等模块进行数据的管理，并对系统的更新作出及时的响应，使得系统能更好地运行。
2. 一般顾客：作为系统的使用者，注册登陆系统后，可以查询系统基本信息、仓库信息、修改个人信息等。

3.2 功能需求分析

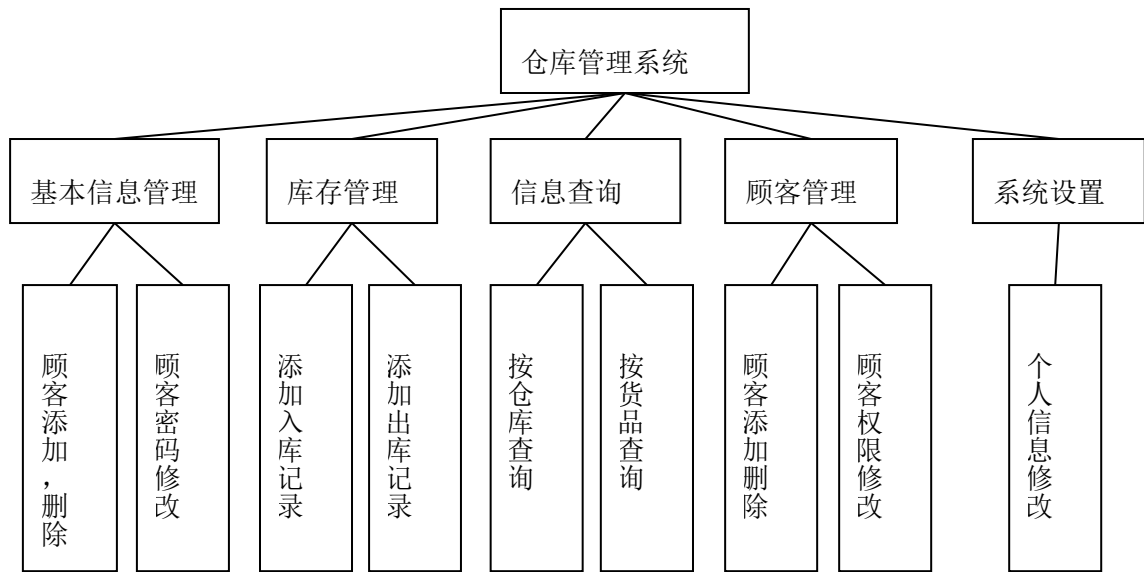


图 3-1 系统功能模块图

本系统是一款仓库管理系统,意在 对仓库的自动化管理以适应信息化社会的生产方式。根据仓库管理系统的需求,划分了仓库的功能模块,重要分为顾客管理模块、入库出库模块、仓库查询模块、盘点记录模块^[7]。详细功能如图 3-1 所示。

3.3 业务流程图

该系统的顾客分为一般顾客和管理员。管理员拥有所有权限,而一般顾客只能进行查询的操作。

3.3.1 管理员模块

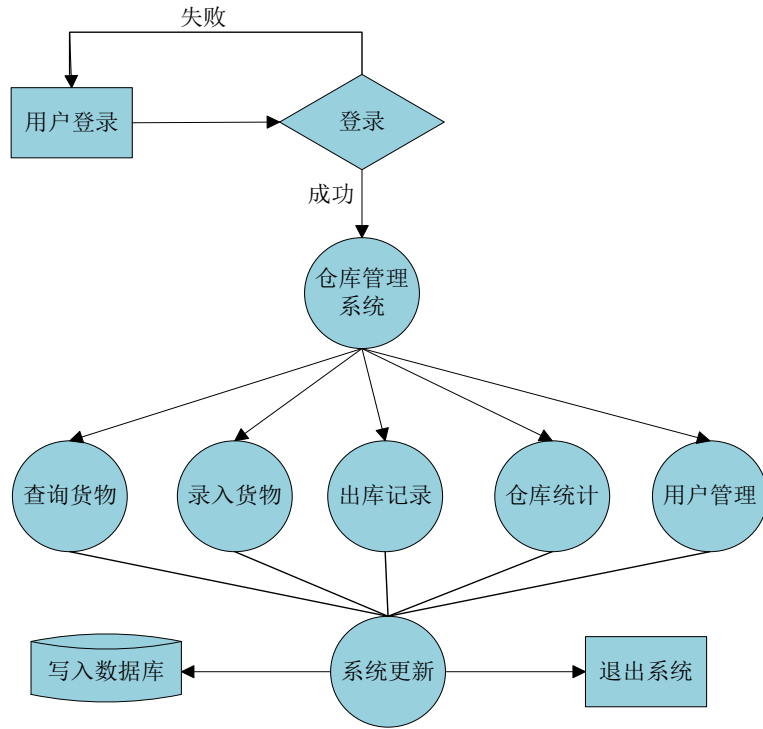


图 3-2 管理员使用流程图

管理员使用流程如图 3-2 所示。

1. 登录

登陆界面包括账号和密码两个文本框和身份选择单项选择框，顾客登陆成功就会转到系统主页面，否则将会提醒账号密码错误。

2. 查询信息

顾客登陆查询信息界面可以对仓库信息（仓库、货品、供应商、部门、入库单、出库单）进行查询。

3. 设置信息（管理员）

管理员顾客登陆设置信息界面可以对仓库信息（仓库、货品、供应商、部门、入库单、出库单）进行设置。

4. 添加记录（管理员）

管理员顾客登陆添加界面可以添加入库和出库信息。

5. 修改密码

顾客可以修改自己的密码，需要先验证旧密码，验证成功后通过输入两次新密码对密码进行修改。

6. 顾客管理（管理员）

管理员顾客可以对已添加的顾客权限进行修改，也可以删除已存在的顾客。

3.3.2 一般顾客模块

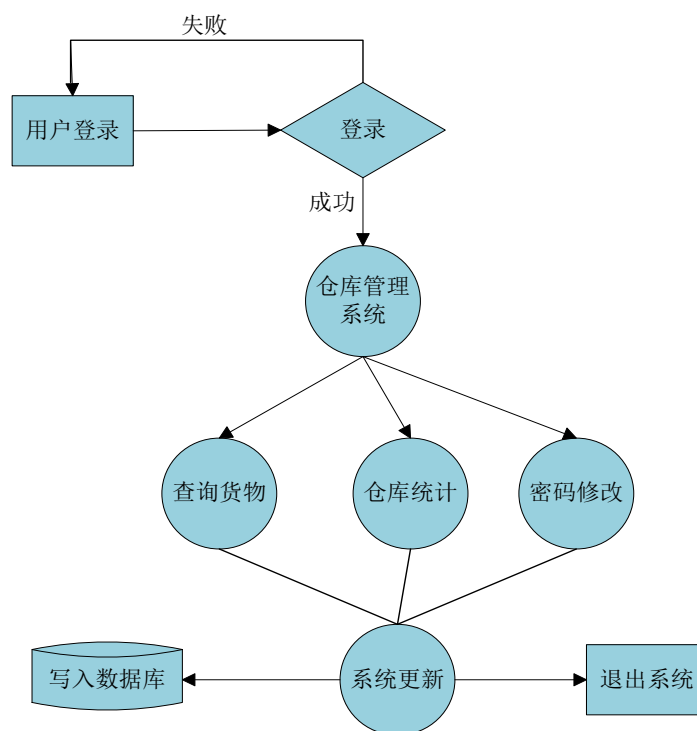


图 3-3 一般顾客使用流程图

一般顾客使用流程如图 3-3 所示。

一般顾客只能进行查询操作和修改个人信息。

4 数据库设计

数据库是一种存储数据并对数据进行操作的工具。数据库的作用在于组织和体现信息，即数据库就是有关信息的集合。本章从 E-R 模型的分析设计为切入点，重点简介了数据库的设计。根据仓库管理系统的功能，将其 E-R 图转化为数据库表，为系统的实现奠定了基础。

4.1 E-R 模型

E-R 模型即实体-联络模型，它是由实体集、属性和联络集构成。实体集用矩形框表达，矩形框内写上实体名。实体的属性用椭圆框表达，框内写上属性名，并用无向边与其对应的实体集相连。实体间的联络用菱形框表达，菱形框中写上以合适的含义命名的名字，用无向连线将参与联络的实体矩形框分别与菱形框相连，并在连线上标明联络的类型，即 1—1、1—N 或 M—N。

本系统的 E-R 模型重要包括部门实体、货品实体、顾客实体、供应商实体、仓库实体、入库单实体和出库单实体共七个实体集和一种各实体联络集。E-R 模型如下文所述。

1. 部门实体用来存储部门信息，E-R 图如图 4-1 所示：

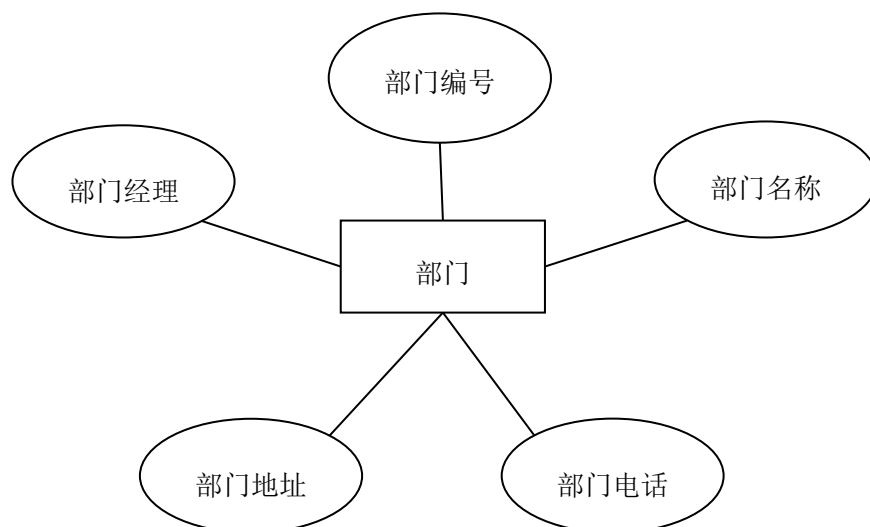


图 4-1 部门实体 E-R 图

1. 货品实体用来存储货品信息，E-R 图如图 4-2 所示：

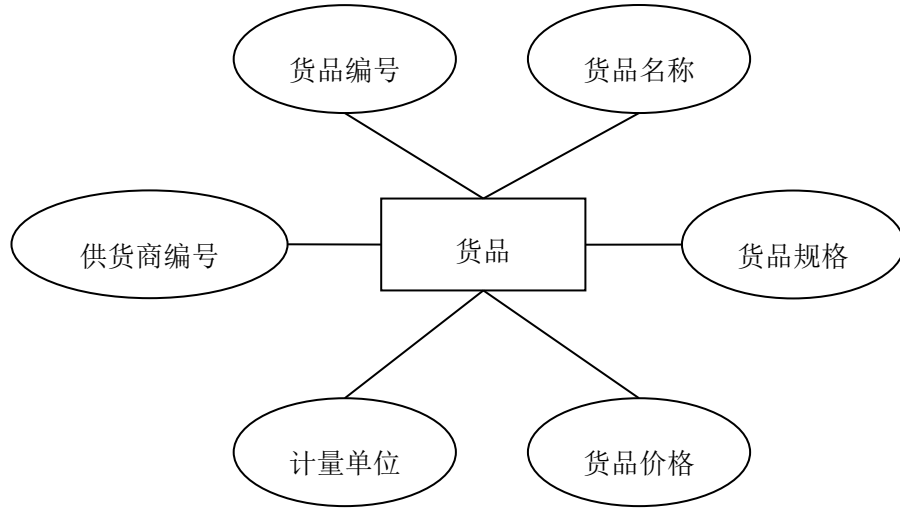


图 4-2 货品实体 E-R 图

2. 顾客实体用来存储顾客信息，E-R 图如图 4-3 所示：

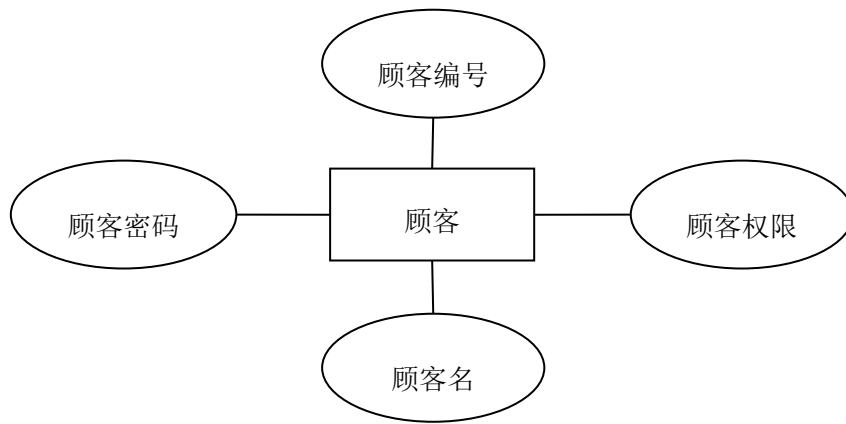


图 4-3 顾客实体 E-R 图

3. 供应商实体用来存储供应商信息，E-R 图如图 4-4 所示：

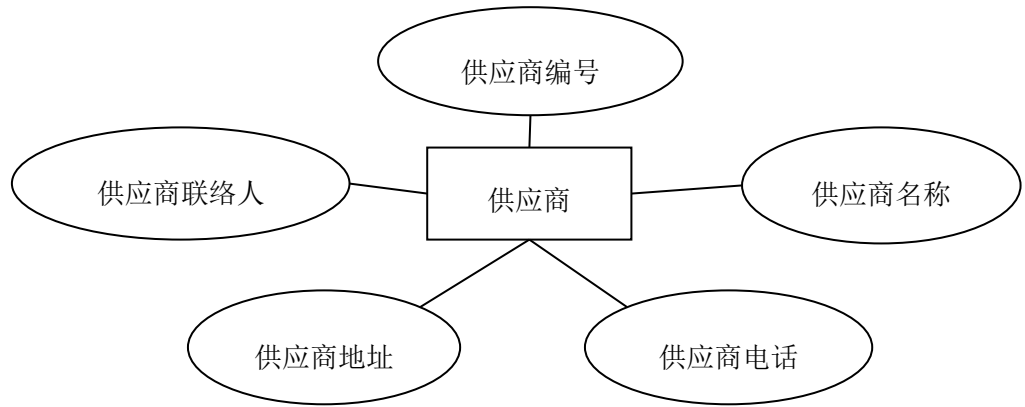


图 4-4 供应商实体 E-R 图

1. 仓库实体用来存储仓库信息，E-R 图如图 4-5 所示：

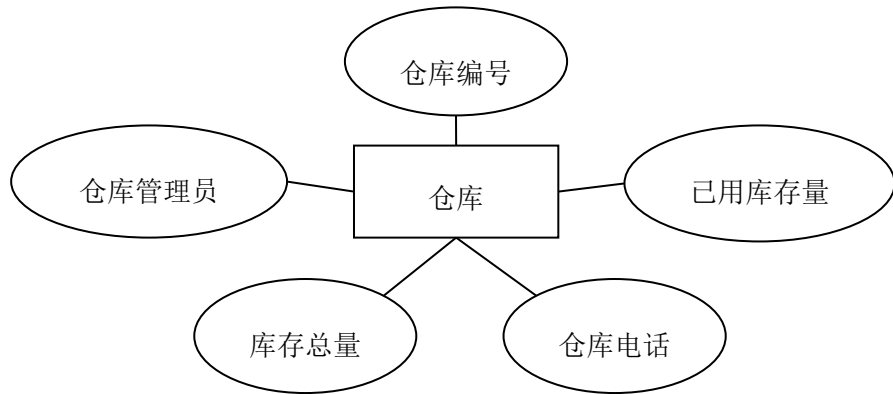


图 4-5 仓库实体 E-R 图

2. 入库单实体用来存储入库单信息，E-R 图如图 4-6 所示：

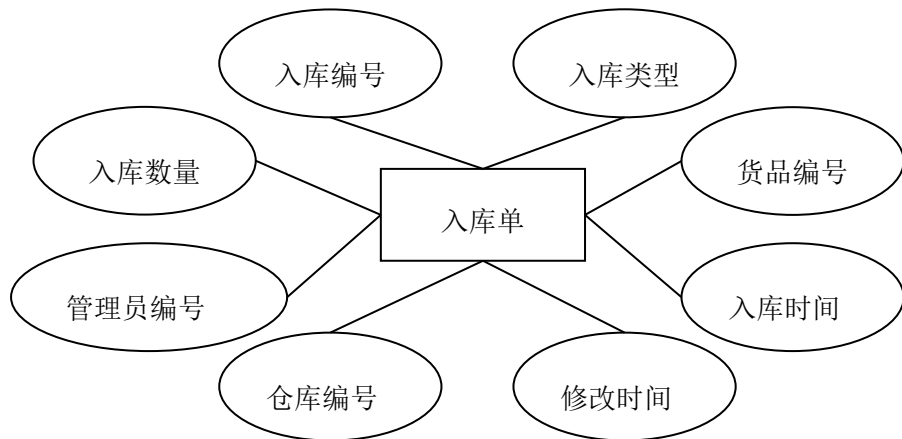


图 4-6 入库单实体 E-R 图

2. 出库单实体用来存储出库单信息，E-R 图如图 4-7 所示：

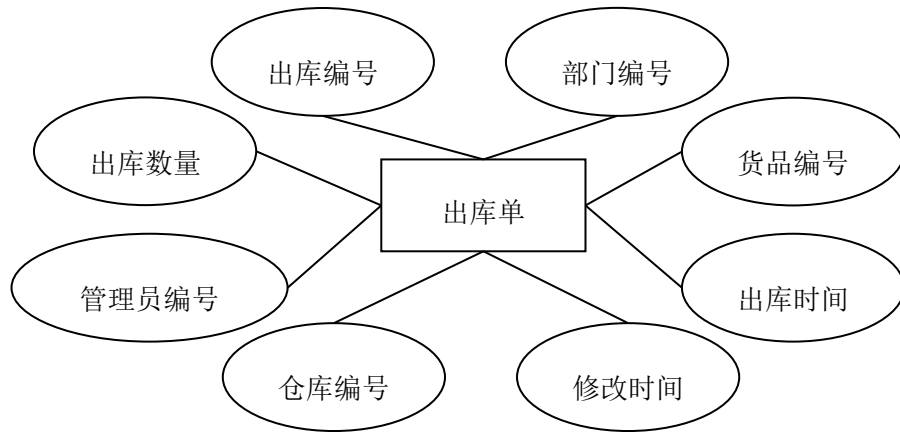


图 4-7 出库单实体 E-R 图

8. 各实体联络 E-R 图如图 4-8 所示。顾客通过添加入库单和出库单记录入库信息和出库信息，供应商提供货品，部门使用货品，一条记录对应一种供应商或者部门，

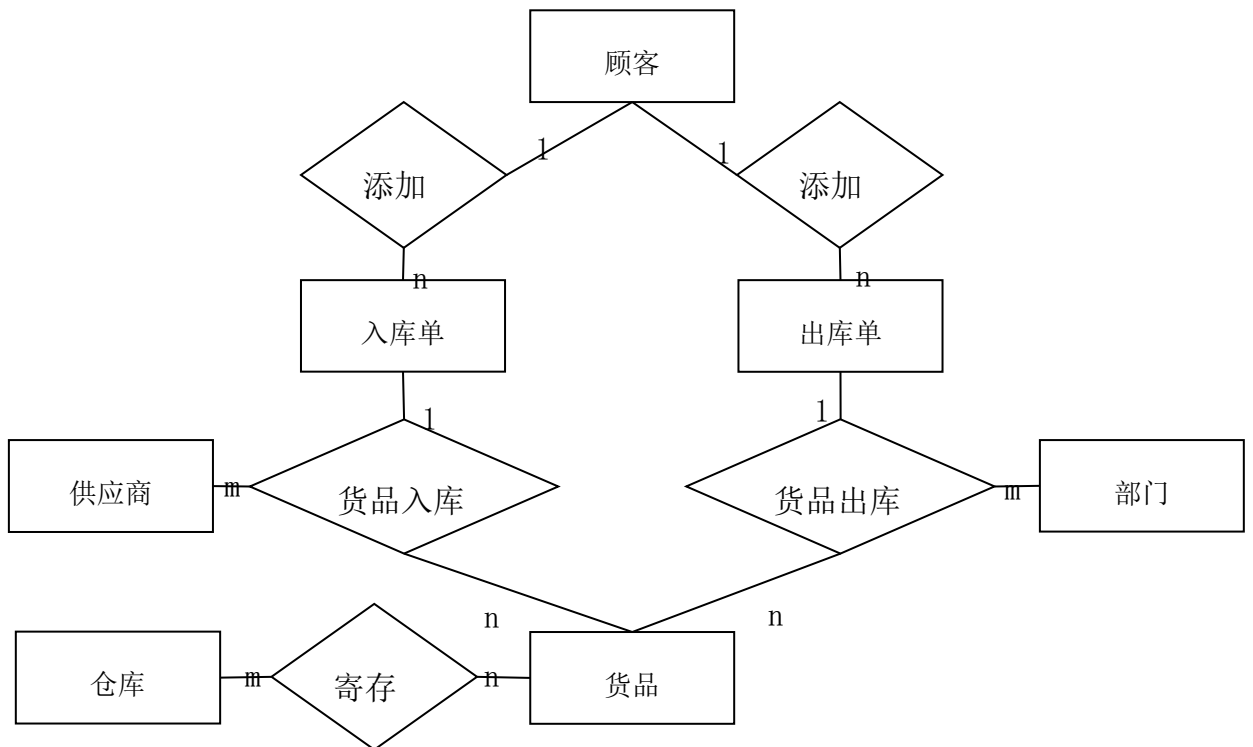


图 4-8 各实体联络 E-R 图

4.2 数据模型

根据 4.1 中分析得出的 E-R 图，本系统采用 MySQL 数据库管理系统，在 MySQL 数据库管理系统中新建名称为 mystorage 的数据库，本系统所使用的的所有数据信息均将存储于该数据库中。如下列举数据表。

1. 部门表 (department)

表 4-1 部门表

列名	数据类型	大小	阐明
depaId	int	4	部门编号
depaName	varchar	24	部门名称
depaPerson	varchar	15	部门经理
depaPhone	char	11	部门电话
depaAddr	varchar	90	部门地址

2. 货品表 (goods)

表 4-2 货品表

列名	数据类型	大小	阐明
goodsId	int	4	货品编号
suppId	int	4	供应商编号
goodsName	varchar	30	货品名称
goodsSize	varchar	10	货品规格
goodsPrice	int	4	货品价格
unit	char	6	计量单位

3. 入库表 (instore)

表 4-3 入库表

列名	数据类型	大小	阐明
instId	int	4	入库编号
instClass	char	1	入库类型

goodsId	int	4	货品编号
instNum	int	4	入库数量
instTime	datetime	8	入库时间
modifyTime	datetime	8	修改时间
stoId	int	4	仓库编号
userId	int	4	管理员编号

4. 出库表 (outstore)

表 4-4 出库表

列名	数据类型	大小	阐明
outstId	int	4	出库编号
depaId	int	4	部门编号
goodsId	int	4	货品编号
outstNum	int	4	出库数量
outstTime	datetime	8	出库时间
modifyTime	datetime	8	修改时间
stoId	int	4	仓库编号
userId	int	4	管理员编号

5. 仓库表 (store)

表 4-5 仓库表

列名	数据类型	大小	阐明
stoId	int	4	仓库编号
stoPerson	varchar	15	仓库管理员
stoPhone	char	11	仓库电话
usedVolu	int	4	已用库存量
allVolu	int	4	库存总量

6. 供应商表 (supplier)

表 4-6 供应商表

列名	数据类型	大小	阐明
----	------	----	----

suppId	int	4	供应商编号
suppName	varchar	45	供应商名称
suppPerson	varchar	15	供应商联络人
suppPhone	char	11	供应商电话
suppAddr	varchar	90	供应商地址

7. 顾客表 (user)

表 4-7 顾客表

列名	数据类型	大小	阐明
userId	int	4	顾客编号
userName	varchar	15	顾客名
pass	varchar	16	顾客密码
userLevel	char	1	顾客权限

8. 库存表 (stock)

表 4-8 库存表

列名	数据类型	大小	阐明
stoId	int	4	仓库编号
goodsId	int	4	货品编号
stock	int	4	库存量

以上就是 mystorage 数据库的数据表。

5 详细设计

5.1 数据库连接类

在对数据库连接的过程中，为了使代码更精简，将对数据库的连接措施封装在 DriverMana 公共类中，其中包括数据库连接措施 getConn、ResultSet 的关闭措施 closers、PreparedStatement 的关闭措施 closestmt、Connection 的关闭措施 closecon。如下即为代码：

```

public class DriverMana {

    public static Connection getConn() {

        Connection con=null;

        try {

            Class.forName("com.mysql.jdbc.Driver");

            con=DriverManager.getConnection("jdbc:mysql://localhost:3306/mystorage
", "root", "root");

        } catch (ClassNotFoundException e) {

            e.printStackTrace();

        } catch (SQLException e) {

            e.printStackTrace();

        }

        return con;

    }

    public static void closers(ResultSet rs){

        if(rs!=null){

            try {

                rs.close();

            } catch (SQLException e) {

                e.printStackTrace();

            }

        }

    }
}

```

```

    }

    public static void closestmt(PreparedStatement pstmt) {

        if(pstmt!=null) {

            try {

                pstmt.close();

            } catch (SQLException e) {

                e.printStackTrace();

            }

        }

    }

    public static void closecon(Connection con) {

        if(con!=null) {

            try {

                con.close();

            } catch (SQLException e) {

                e.printStackTrace();

            }

        }

    }

}

```

5.2 系统重要模块

5.2.1 登录模块

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/635222110314011231>

5.2.2