

学号

10212817305

武汉华夏理工学院

# 课 程 设 计

课程名称 软件工程

题 目 超市库存管理系统的设计与开发

专 业 软件工程

班 级 软件 1173

姓 名 郑秋

成 绩 \_\_\_\_\_

指 导 老 师 刘琳琳

2019 年 12 月 30 日至 2020 年 1 月 3 日

# 武汉华夏理工学院信息工程学院

## 课程设计任务书

课程名称：\_\_\_\_软件工程\_\_\_\_

指导教师：\_\_\_\_刘琳琳\_\_\_\_

班级名称：\_\_\_\_软件 1173\_\_\_\_

开课院、系：\_\_\_\_计算机与网络工程系\_\_\_\_

### 一、课程设计目的与任务

软件工程课程设计是《软件工程》课程的后续实践课程，本课程设计的目的是通过一周的实践训练，在前期实验课基础上继续完成系统全生命周期的各个阶段任务，通过软件工程实验课到软件工程课程设计，使同学们经历一个软件从问题定义、分析、设计、开发和测试的全过程和受到一次软件系统开发的综合训练，以便能熟练掌握软件开发的完整生命周期过程及较全面地理解、掌握和综合运用所学的软件工程的知识。结合具体的学生信息管理系统等选题开发项目，理解并掌握需求分析、系统设计、系统实施的主要环节和步骤以及软件文档的制作能力，进一步提高学生分析问题、解决问题，进行一个完整系统开发的能力。

### 二、课程设计的内容与基本要求

要求学生根据所学的软件工程和编程知识，利用所学的 C# 和 SQL 知识实现一个超市库存管理系统。功能详见下表。

表 1.1 超市库存管理系统主要功能表

编号	功能名称	功能说明
1	用户管理	管理员添加用户，普通用户无法访问此页面
2	类别管理	对商品类别进行增删改查，类别可以在商品管理页面中的下拉菜单中实时更新
3	商品管理	登记商品的基本信息（编号，名称，类型，价格，数量，备注），提供查询、修改删除功能
4	出库管理	登记商品出库情况，包括数量、经手人等，提供查询，商品库存不足警告功能
5	入库管理	登记商品入库情况，包括数量、经手人等，提供查询
6	用户操作	登录，退出系统等

### 三、学时分配进度安排

序号	设计内容	所用时间
1	选题及需求调研	1 天
2	软件结构设计	1 天
3	软件编码设计	1 天
4	软件测试及撰写报告	1 天
5	答辩	1 天
合 计		1 周

#### 四、课程设计考核及评分标准

##### 1. 设计报告要求

课程设计报告要求逻辑清晰、层次分明、书写整洁。格式包括标题、提要、正文（包括①项目背景和意义。②可行性分析和需求分析。③总体设计。④详细设计。⑤编码测试。⑥答辩。⑦成绩评定。）附录(图纸、程序清单等)。设计报告须每人一份，独立完成。

##### 2. 图纸要求

系统流程图、数据流图、E-R图、用例图、类图、时序图等。

##### 3. 评分标准

评分依据	评分成绩
1. 选题合理、目的明确	10 分
2. 态度认真、学习刻苦、遵守纪律	10 分
3. 设计方案正确，具有可行性	20 分
4. 独立性与创新性	5 分
5. 系统调试与结果（功能操作使用正常、软件程序完整）	20 分
6. 参考文献充分（不少于 5 篇）	5 分
7. 设计报告撰写规范	10 分
8. 答辩	20 分
总分	100 分

注：成绩等级：优（90分—100分）、良（80分—89分）、中（70分—79分）、及格（60分—69分）、60分以下为不及格。

##### 五、指导时间

周次	星期一	星期二	星期三	星期四	星期五
第 18 周	第 5-6 节	第 3-4 节		第 3-4 节	第 1-4 节
地点	实验室 231	实验室 231		实验室 231	实验室 231

执笔： 刘琳琳      日期：2019-12-1  
审阅： 钱小红      日期： 2019-12-2

# 目 录

1 目标 .....	5
2 软件开发环境与技术简介: .....	5
2.1 Visual Studio 开发环境简介 .....	5
2.2 C#语言简介 .....	5
2.3 数据库技术 .....	6
2.3.1 SQL Server 简介 .....	6
2.3.2 SQL SERVER 访问机制 .....	6
3. 需求分析 .....	7
4. 系统总体设计 .....	8
4.1 系统体系结构设计 .....	8
4.2 系统数据库设计 .....	10
4.2.1 实体属性定义 .....	10
4.2.2 数据库物理设计 .....	10
4.2.3 数据库结构 E-R 图 .....	11
5. 系统详细设计 .....	12
5.1 数据库的创建和访问 .....	12
5.1.1 数据库的实现 .....	12
5.1.2 数据库的连接 .....	12
5.2 页面的实现 .....	12
5.2.1 登录页面 .....	13
5.2.2 系统管理 .....	14
5.2.3 用户管理 .....	15
5.2.4 商品类别管理 .....	15
5.2.5 商品管理 .....	20
5.2.6 入库操作 .....	23
6. 系统测试 .....	26
6.2.1 登录页面测试 .....	26
6.2.2 用户信息管理测试 .....	27
6.2.3 商品类别信息管理测试 .....	28
6.2.4 商品信息管理测试 .....	30
6.2.5 商品入库测试 .....	32
7. 系统设计与开发结论 .....	33

# 1 目标

本课程设计将利用 C#窗体，ADO.NET 数据访问技术和 LINQ 查询设计和开发一个管理员使用的超市库存管理系统。系统将实现以下功能：

- 1) 管理员登录：管理员登录成功才能完成管理功能；
- 2) 商品信息管理：将实现商品信息的增、删、改、查功能；
- 3) 商品类别管理：将实现商品类别信息的增、删、改、查功能；
- 4) 用户管理：将实现用户信息的增、删、改、查功能；
- 5) 商品入库：将实现商品入库功能；
- 6) 商品出库：将实现商品出库功能；

## 2 软件开发环境与技术简介：

硬件平台：个人 PC

软件平台：win 10

开发环境：Visual Studio 2013 SQL Server 2012

开发技术：C# SQL 面向对象编程

### 2.1 Visual Studio 开发环境简介

VS 是美国微软公司的开发工具包系列产品。VS 是一个基本完整的开发工具集，它包括了整个软件生命周期所需要的大部分工具，如 UML 工具、代码管控工具、集成开发环境 (IDE) 等等。

### 2.2 C#语言简介

C#是微软公司在 2000 年 7 月发布的一种全新且简单、安全、面向对象的程序设计语言，是专门为 .NET 的应用而开发的语言。它吸收了 C++、Visual Basic、Delphi、Java 等语言的优点，体现了当今最新的程序设计技术的功能和精华。C#继承了 C 语言的语法风格，同时又继承了 C++的面向对象特性。不同的是，C#的对象模型已经面向 Internet 进行了重新设计，使用的是 .NET 框架的类库；C#不再提供对指针类型的支持，使得程序不能随便访问内存地址空间，从而更加健壮；C#不再支持多重继承，避免了以往类层次结构中由于多重继承带来的可怕后果。.NET 框架为 C#提供了一个强大的、易用的、逻辑结构一致的程序设计环境。同时，公共语言运行时 (Common Language Runtime) 为 C#程序语言提供了一个托管的运行环境，使程序比以往更加稳定、安全。其特点有：语言简洁、保留了 C++的强大功能、快速应用开发功能、语言的自由性、强大的 Web 服务器控件、支持跨平台、与 XML 相融合。

## 2.3 数据库技术

### 2.3.1 SQL Server 简介

SQL Server 是 Microsoft 公司推出的关系型数据库管理系统。具有使用方便可伸缩性好与相关软件集成程度高等优点，可跨越从运行 Microsoft Windows 98 的膝上型电脑到运行 Microsoft Windows 2012 的大型多处理器的服务器等多种平台使用。Microsoft SQL Server 是一个全面的数据库平台，使用集成的商业智能 (BI) 工具提供了企业级的数据管理。Microsoft SQL Server 数据库引擎为关系型数据和结构化数据提供了更安全可靠的数据存储功能，使您可以构建和管理用于业务的高可用和高性能的数据应用程序。

### 2.3.2 SQL SERVER 访问机制

本系统采用 ADO.NET 技术和 LINQ 查询技术访问 SQL Server 2012 数据库服务器。

#### 2.3.2.1 ADO.NET 数据访问机制

ADO.NET 是基于 .NET 的应用程序的数据访问模型。可以使用它来访问关系数据库系统（如 SQL Server 2005、Oracle）和其他许多具有 OLE DB 或 ODBC 提供程序的数据源。在某种程度上，ADO.NET 代表 ADO 技术的最新进展。不过，ADO.NET 引入了一些重大变化和革新，旨在解决 Web 应用程序的松耦合特性以及在本质上互不关联的特性。

该模型重新整合 OLE DB 和 ADO，并在此基础上构造了新的对象模型。该模型既提供了保持连接的数据访问形式，又提供了松耦合的、以 DataSet 对象为操作目标的数据访问形式。

#### 2.3.2.2 LINQ 查询技术

LINQ 是 Language Integrate Query 的缩写，意为语言集成查询，是微软在 .Net Framework 4.5 版中推出的主要特性之一。

它为开发人员提供了统一的数据查询模式，并与 .Net 开发语言（如 C# 和 VB. Net）集成，很大程度上简化了数据查询的编码和调试工作，提供了数据查询的性能。

LINQ 中查询表达式访问的是一个对象，而该对象可以表示为各种类型的数据源。比如 SQL Server 数据库，XML 文档，ADO.NET 数据集，以及内存中的数据集合等。

### 3. 需求分析

本课程设计将利用 C#窗体，ADO.NET 数据访问技术和 LINQ 查询设计和开发一个管理员使用的超市库存管理系统。对于该系统，首先得实现对管理员用户的登录，管理员登录后进入系统主界面，从而对商品信息、商品类别信息和出入库信息各种操作。超市库存管理系统主要是为了管理员能够更好地去管理商品以及商品类别，及时了解商品的动态。其中包括商品信息的查询，能够及时的了解商品的情况；还包括类别信息的查询，使货物的出入库及时做调整；还包括管理员的信息的查询。

#### 3.1 数据流图

##### 3.1.1 顶层数据流图

管理员登录后进入主系统，主系统将管理员需要的信息返回，数据流图如下：

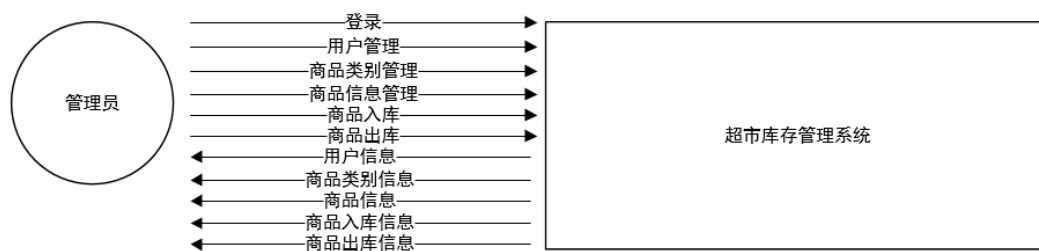


图 3.1

##### 3.1.2 1 层数据流图

管理员输入账号密码，通过查询用户表来验证，若验证通过，则进入主界面，主界面可供选择各种功能，数据流图如下：

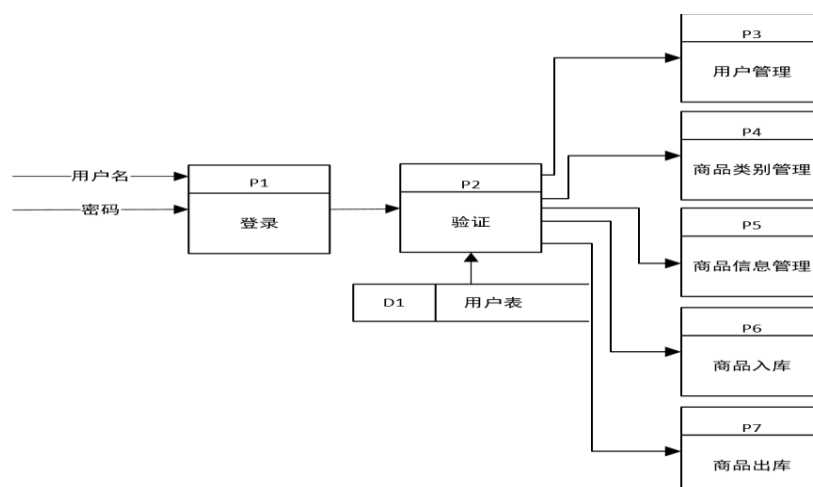


图 3.2

在用户管理模块中，先对用户表查询，判断是否为管理员登录，若不是，则提示权限不够，若为管理员，则可以执行各种操作，数据流图如下：

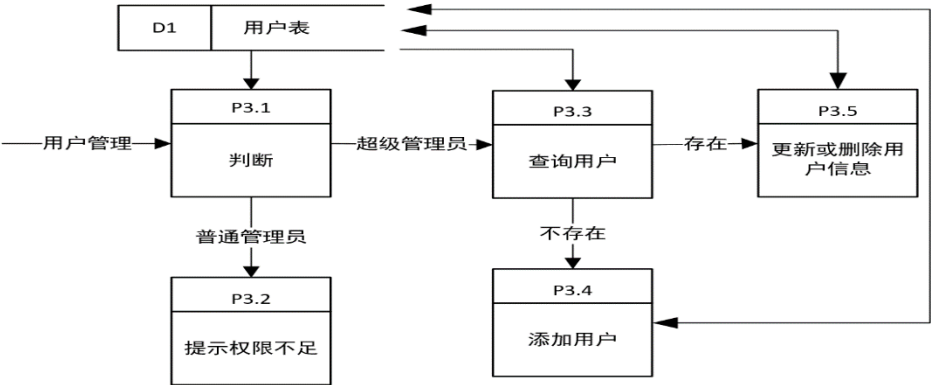


图 3.3

## 4. 系统总体设计

### 4.1 系统体系结构设计

#### 4.1.1 结构设计

从需求分析可以看出，本系统可以设计为五个功能模块，分别是用户信息管理模块、学生信息管理模块、课程信息管理模块、成绩信息管理模块、退出系统模块。系统整体功能如图 4.1 所示：

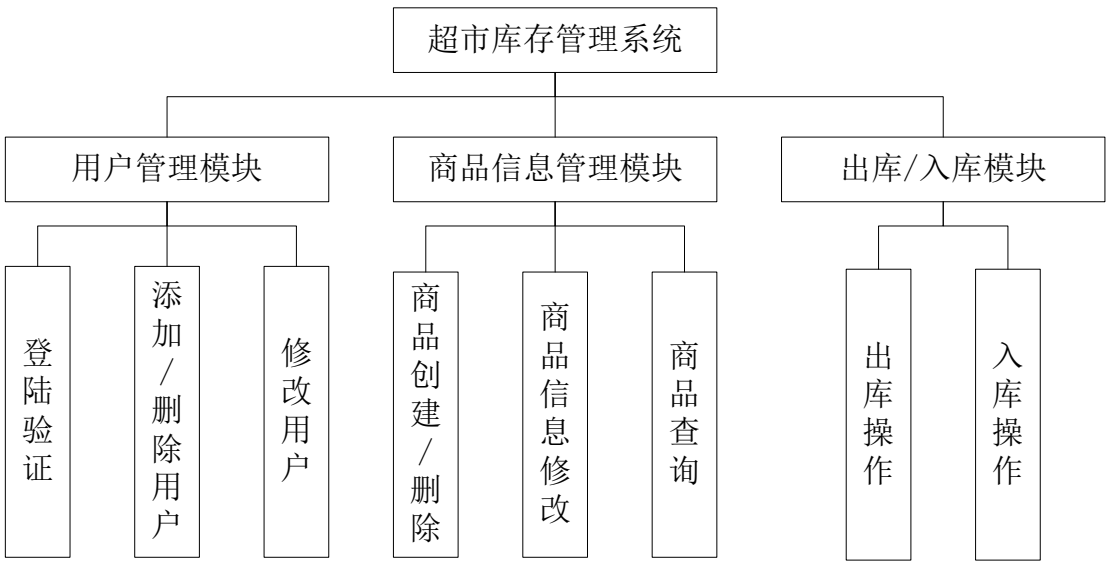
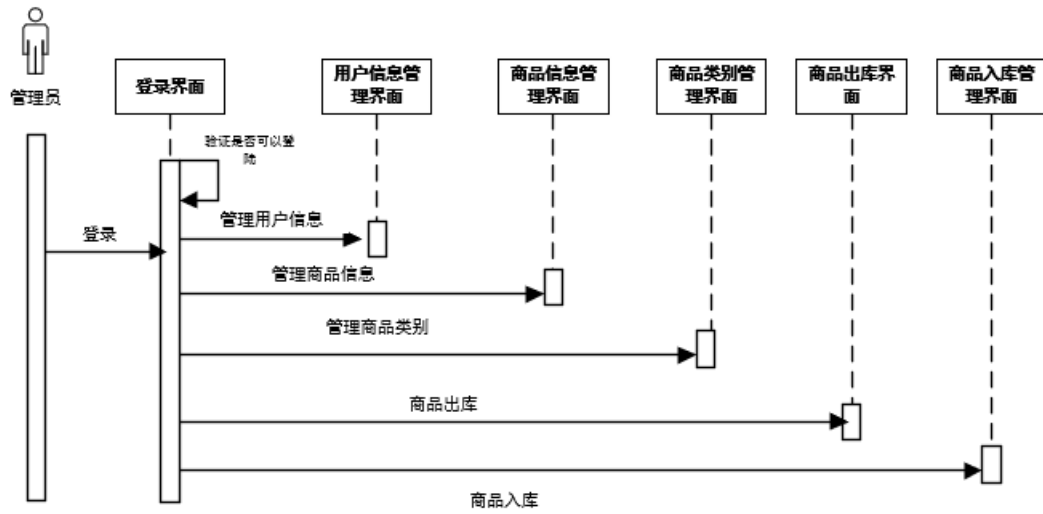


图 4.1 系统模块结构图

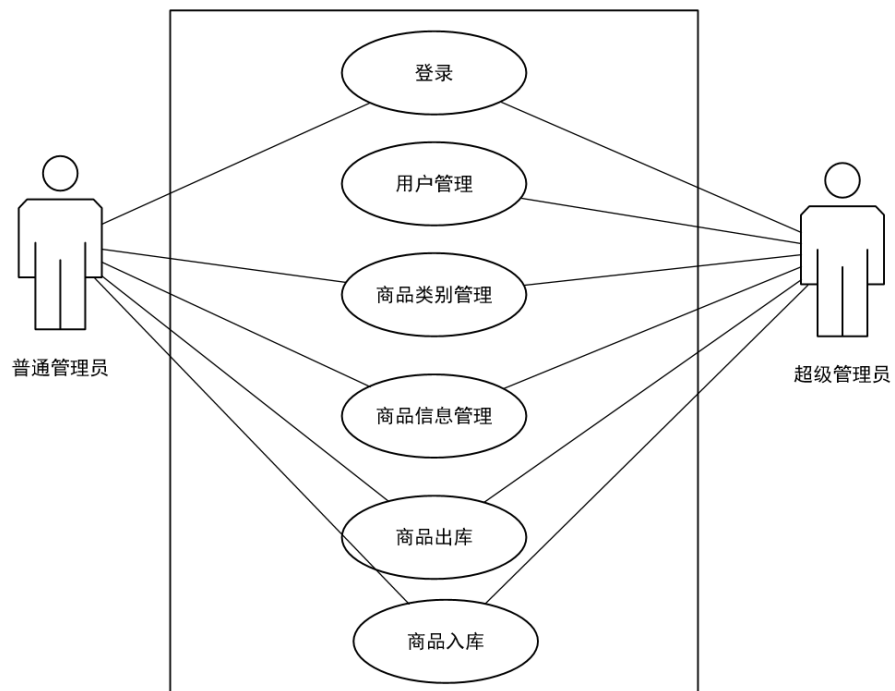


#### 4.1.2 系统时序图

管理员通过验证登陆界面登录到主界面，在主界面选择需要的各种功能。



#### 4.1.3 用例图



## 4.2 系统数据库设计

### 4.2.1 实体属性定义

- (1) 用户表：(用户名称, 密码, 电话, 性别);
- (2) 商品信息表：(商品编号, 商品名, 卖出价, 进货价, 类别, 库存数量, 图片, 备注);
- (3) 商品类别表：(类别号, 类别名, 备注);
- (4) 出入库信息表：(操作编号, 商品编号, 数量, 价格, 出/入库, 操作人, 备注)

### 4.2.2 数据库物理设计

根据在数据库概念设计中给出的数据库建立管理系统的关系模型,可以设计数据表结构。其中包括以下表: 用户表 (Table\_Table\_User), 商品信息表 (Table\_Commodity), 类别信息表 (Table\_CommodityClass), 出入库信息表 (Table\_Console)。下面具体介绍数据库中各表的结构:

- (1) 表名: Table\_User, 即: 用户信息表, 储蓄用户(管理员)的相关信息, 其结构如图 4.2 所示:

图 4.2 用户表 (Table\_Table\_User)

	名称	数据类型	允许 Null	默认值	
	userName	varchar(50)	<input type="checkbox"/>		
	userPwd	varchar(50)	<input checked="" type="checkbox"/>		
	userSex	varchar(2)	<input checked="" type="checkbox"/>		
	userPhone	varchar(20)	<input checked="" type="checkbox"/>		
	note	varchar(50)	<input checked="" type="checkbox"/>		
			<input type="checkbox"/>		

- (2) 图名: Table\_Commodity, 即: 商品信息表, 储蓄商品的相关信息, 其结构如表 4.3 所示。

图 4.3 商品信息表 (Table\_Commodity)

	名称	数据类型	允许 Null	默认值	
	CId	int	<input type="checkbox"/>		
	CName	varchar(50)	<input type="checkbox"/>		
	CBuyPrice	varchar(50)	<input checked="" type="checkbox"/>		
	CSalePrice	varchar(50)	<input checked="" type="checkbox"/>		
	CType	varchar(50)	<input checked="" type="checkbox"/>		
	CQty	varchar(50)	<input checked="" type="checkbox"/>		
	CPicture	varchar(50)	<input checked="" type="checkbox"/>		
	CNote	varchar(50)	<input checked="" type="checkbox"/>		
			<input type="checkbox"/>		

(3) 图名: Table\_Ctype, 即: 课程信息表, 存储商品类别的相关信息, 其结构 4.4 所示。

图 4.4 类别信息表 (Table\_Commodityclass)

	名称	数据类型	允许 Null	默认值	
	SId	int	<input type="checkbox"/>		
	SName	varchar(50)	<input checked="" type="checkbox"/>		
	SNote	varchar(50)	<input checked="" type="checkbox"/>		
			<input type="checkbox"/>		

(4) 图名: Table\_console, 即: 出入库信息表, 存储商品出库/入库的相关信息, 其结构如表 4.5 所示。

表 4.5 出入库信息表 (Table\_console)

	名称	数据类型	允许 Null	默认值	
	RId	int	<input type="checkbox"/>		
	CId	int	<input type="checkbox"/>		
	Number	varchar(50)	<input type="checkbox"/>		
	Price	varchar(50)	<input type="checkbox"/>		
	Choose	varchar(50)	<input type="checkbox"/>		
	UserName	varchar(50)	<input type="checkbox"/>		
			<input type="checkbox"/>		

4. 2. 3 数据库结构 E-R 图

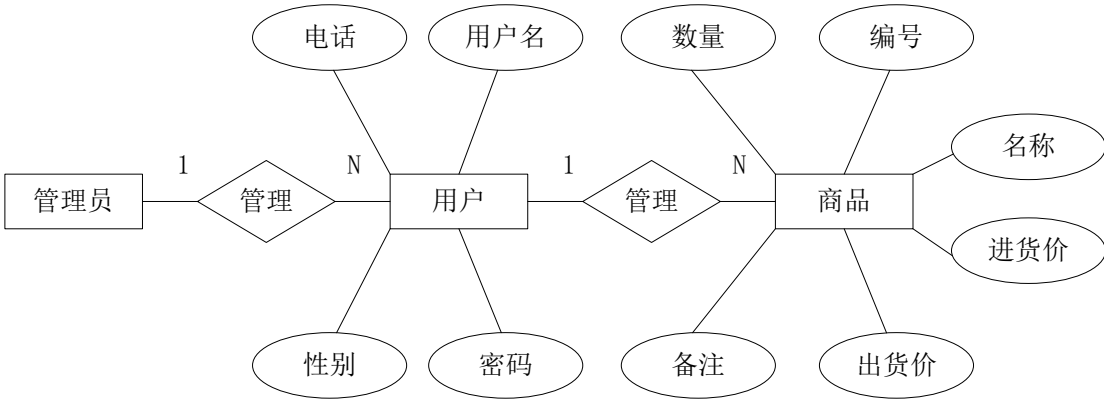


图 4.2 数据库结构 E-R 图

## 5. 系统详细设计

### 5.1 数据库的创建和访问

本系统用到的数据库是 Sql server 2012。

#### 5.1.1 数据库的实现

打开 Sql Server 2012, 先用 Windows 权限去登录。新建数据库 Table\_User 和用户表 (Table\_Table\_User), 商品信息 (Table\_Commodity), 类别表 (Table\_Commodityclass), 出入库信息表 (Table\_Console)。

- (1) 新建用户表 (Table\_User), 其表如图 4.2 所示;
- (2) 新建商品类别信息表 (Table\_Commodityclass), 其表如图 4.3 所示;
- (3) 新建商品信息表 (Table\_Commodity), 其表如图 4.4 所示;
- (4) 新建商品出入库信息表 (Table\_Console), 其表如图 4.5 所示;

#### 5.1.2 数据库的连接

数据库的访问, 在项目中, 我们利用的是 ADO.NET 技术获得与数据库的连接其中创建数据库连接字符串时, 可能会出现小问题, 连接不同的数据库服务, 使用的数据库连接字符串也不相同, 下面我使用的是第三种, 用来连接 Sql Server Express 版, 也就是随 VS 一起安装的那个数据库版本。

混合身份登录:

```
string connStr = "Data Source=localhost;Initial Catalog=SuperMarketDB;Integrated Security=True";

SqlConnection connection = new SqlConnection(connStr);
string sql = string.Format("select count(*) from SIMS_user where userName='{0}' and userPwd='{1}'", userName.Text, passWord.Text);

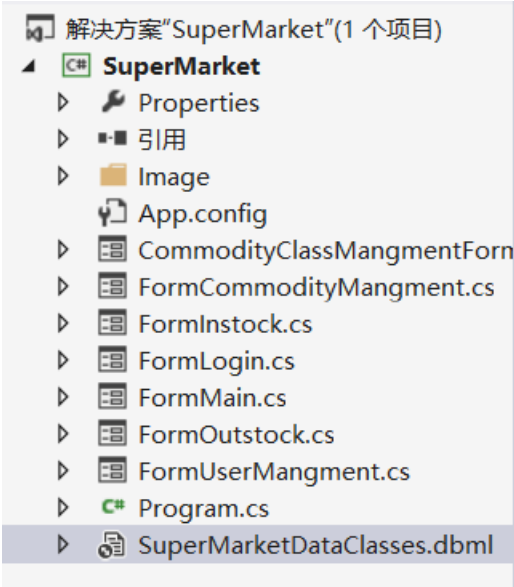
SqlCommand command = new SqlCommand(sql, connection);
```

### 5.2 页面的实现

使用 Connection 连接数据库, 使用 Command、DataReader 对象访问数据库, 使用 DataReader 与 DataSet 对象操作数据库, 实现 LINQ to SQL 的编程。

如图 5.2-1 所示为整个项目的目录结构, 其中 CategoryManger.cs 为商品管理类, CategoryTypeManger 为商品类别管理类, FormIndex.cs 为主界面类, 使用菜单栏的形式将

各个模块联系在一起，FormLogin.cs 为实现登录验证的模块，InOperation 为入库操作模块类，OutOperation 为出库操作模块类，Program.cs 为主方法入口，Table\_UserManger 为用户管理模块类，SuperMarketMangerDataSet.xsd 为创建了数据库连接及 DataSet 之后，系统自动创建的文件。



5.2-1 项目解决方案

5.2.1 登录页面

用户访问超市库存管理系统时，首先进入的登录界面，登录成功会进入到主界面，在设计视图下，触发登录窗体中“登录”按钮的 Click 事件，重写按钮的响应事件，如图 5.2.1-1 为登录页面，图 5.2.1-2 为登录验证的登录流程图。



图 5.2.1-1 登录界面

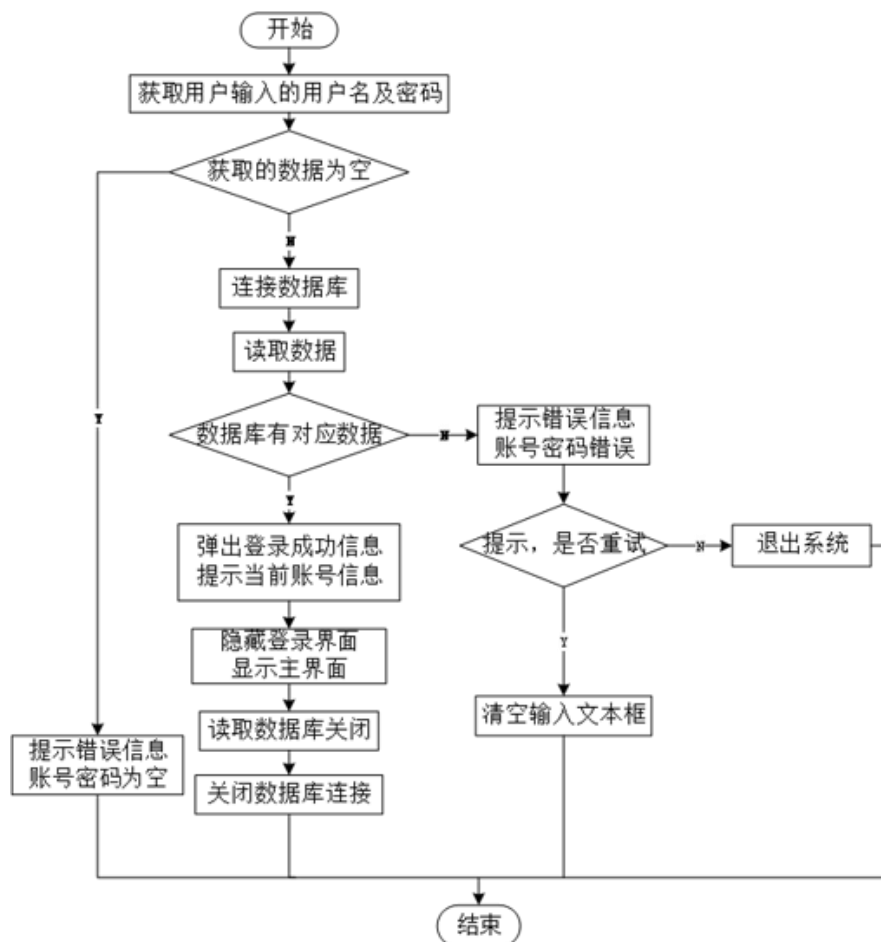


图 5.2.1-2 登录验证流程图

## 5.2.2 系统管理

本页面提供用户操作入口,用户可以选择相应模块对其进行增、删、改、查等操作。其界面如下图 5.2.2 所示:本页面提供用户操作入口,用户可以选择相应模块对其进行增、删、改、查等操作。其界面如下图 5.2.2-1 所示:

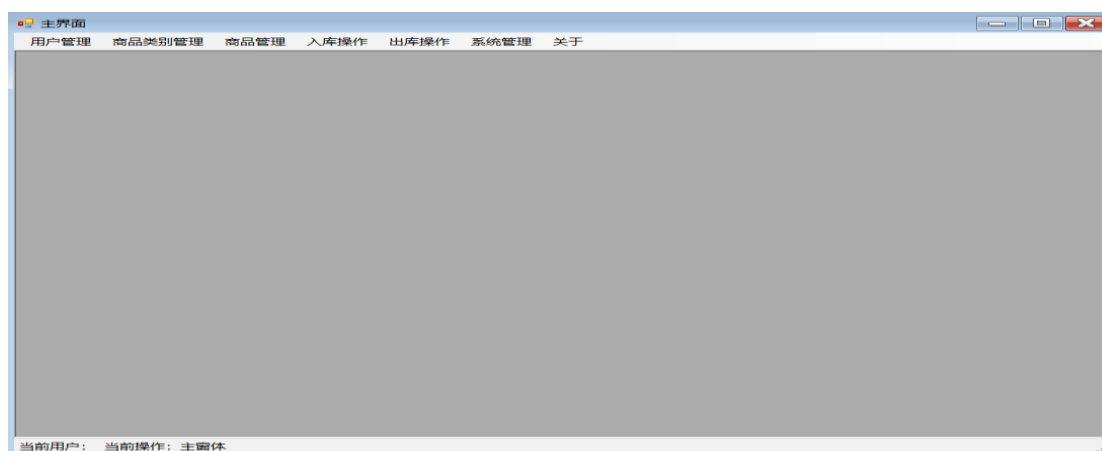


图 5.2.2-1 主界面设计

5.2.3 用户管理

通过切换主页上的选项卡，可进入到用户信息管理页面，本页面提供用户操作入口, 用户可以选择操作模块对其进行增、删、改、查等操作，其界面如下图 5.2.3-1 所示：

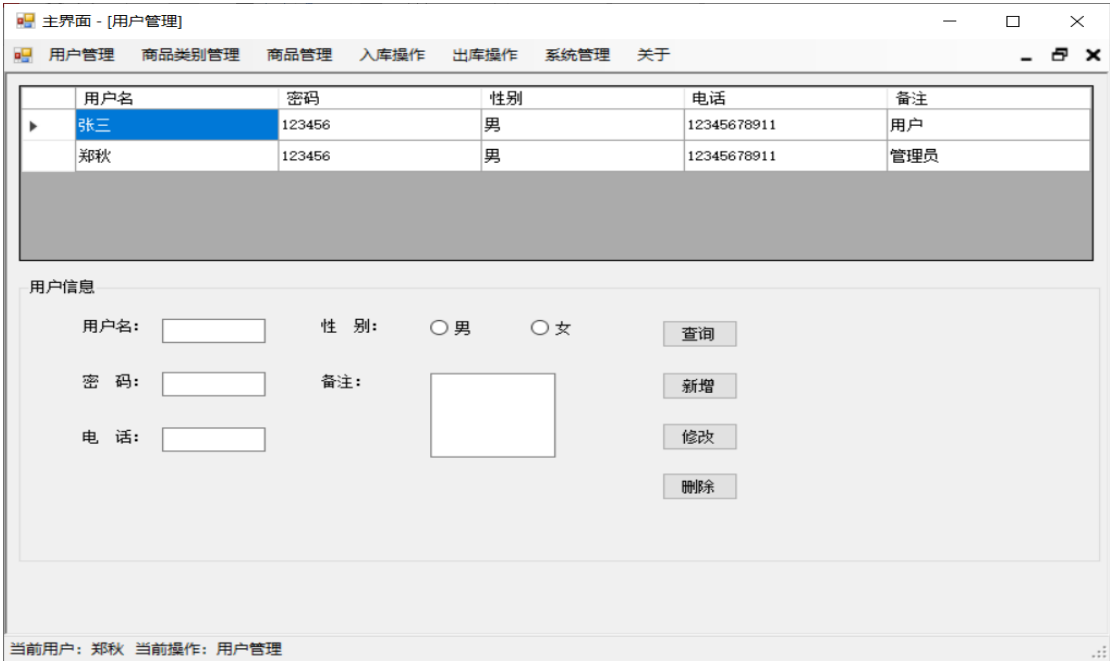


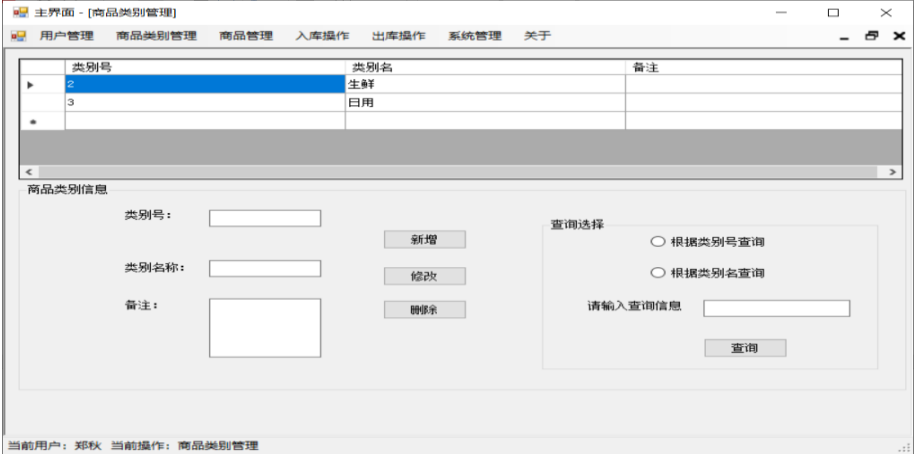
图 5.2.3-1 用户管理设计视图

上方是作为显示用户的基本信息，使用DataGridView控件将数据做成表格，用户在进行增删改查时，不同的按钮点击事件通过调用dataGridView.DataSource方法将读取的数据库信息显示在表格上,其方法代码如下：

```
SuperMarketDataClassesDataContext db = new SuperMarketDataClassesDataContext();
dataGridView1.DataSource = db.SIMS_user;
```

5.2.4 商品类别管理

通过切换主页上的选项卡，可进入到商品类别管理页面，本页面提供用户操作入口, 用户可以选择商品类别管理模块对商品类别进行增、改、查等操作，其界面如下图 5.2.4-1 所示：



5.2.4-1 商品类别管理设计视图

商品类别管理实现“增加”，“修改”功能的代码如下：

//添加

```
private void button1_Click(object sender, EventArgs e)
{
    SuperMarketDataClassesDataContext db = new
SuperMarketDataClassesDataContext();
    SISM_commodityclass scc = new SISM_commodityclass();
    if (textBox4.Text != "")
    {
        scc.SId = int.Parse(textBox4.Text);
        scc.SName = textBox2.Text;
        scc.SNote = textBox3.Text;
        db.SISM_commodityclass.InsertOnSubmit(scc);
        try
        {
            db.SubmitChanges();
            dataGridView1.DataSource = db.SISM_commodityclass;
            MessageBox.Show("添加成功!");
            clear();
        }
        catch (Exception ex)
        {

        }
    }
    else MessageBox.Show("请输入添加信息!");
}
```

//清除文本框内容

```
public void clear()
{

    textBox2.Clear();
    textBox3.Clear();
    textBox4.Clear();

}
```

//删除

```
private void button4_Click(object sender, EventArgs e)
{
    SuperMarketDataClassesDataContext db = new
```



```

SuperMarketDataClassesDataContext();
//id删除
if (textBox4.Text != "") {

    var q = (from m in db.SISM_commodityclass
              where m.SId == int.Parse(textBox4.Text)
              select m);
    foreach (var id in q)
    {
        db.SISM_commodityclass.DeleteOnSubmit(id);
    }
    var num = (from m in db.SISM_commodityclass
                where m.SId == int.Parse(textBox4.Text)
                select m).Count();
    if (num > 0)
    {
        try
        {
            db.SubmitChanges();
            MessageBox.Show("删除成功!");
            dataGridView1.DataSource = db.SISM_commodityclass;
            clear();

        }
        catch (Exception ex)
        {

        }
    }
    else MessageBox.Show("没有此类别号的商品!");
}
else if (textBox2.Text != "")
{ //根据名称删除
    var p = (from m in db.SISM_commodityclass
              where m.SName == textBox2.Text
              select m);
    foreach (var name in p)
    {
        db.SISM_commodityclass.DeleteOnSubmit(name);
    }
    var num = (from m in db.SISM_commodityclass
                where m.SName == textBox2.Text
                select m).Count();
}

```

```

        if (num > 0)
        {
            try
            {
                db.SubmitChanges();
                MessageBox.Show("删除成功!");
                dataGridView1.DataSource = db.SISM_commodityclass;
                clear();
            }
            catch (Exception ex)
            {
            }
        }
        else MessageBox.Show("没有此类别的商品!");
    }
    else MessageBox.Show("请输入删除条件");
}

//查询
private void button3_Click(object sender, EventArgs e)
{
    SuperMarketDataClassesDataContext db = new
    SuperMarketDataClassesDataContext();
    if (radioButton1.Checked)
    {
        var q = (from m in db.SISM_commodityclass
                  where m.SId == int.Parse(textBox1.Text)
                  select m);
        foreach (var c in q)
        {
            textBox4.Text = c.SId.ToString();
            textBox2.Text = c.SName;
            textBox3.Text = c.SNote;
        }
    }
    else if (radioButton2.Checked)
    {
        var q = (from m in db.SISM_commodityclass
                  where m.SName == textBox1.Text

```

```

                select m);
foreach (var name in q)
{

    textBox4.Text = name.SId.ToString();
    textBox2.Text = name.SName;
    textBox3.Text = name.SNote;

}
}
else MessageBox.Show("请输入查询条件!");
}

//更新
private void button2_Click(object sender, EventArgs e)
{
    SuperMarketDataClassesDataContext db = new
SuperMarketDataClassesDataContext();
    if (textBox4.Text != "")
    {
        var q = (from m in db.SISM_commodityclass
                where m.SId == int.Parse(textBox4.Text)
                select m);
        foreach (var c in q)
        {

            c.SId = int.Parse(textBox4.Text);
            c.SName = textBox2.Text;
            c.SNote = textBox3.Text;

        }
        var num = (from m in db.SISM_commodityclass
                where m.SId == int.Parse(textBox4.Text)
                select m).Count();
        if (num > 0)
        {
            try
            {
                db.SubmitChanges();
                MessageBox.Show("更新成功!");
                dataGridView1.DataSource = db.SISM_commodityclass;

            }
            catch (Exception ex)
            {

```

```

    }
}
}
else MessageBox.Show("请输入需更新的类别编号!");
}
}

```

### 5.2.5 商品管理

商品管理页面显示的信息是商品的具体信息，包括商品的编号，名称，卖价，进价，商品类别，商品数量以及商品的图片。通过 `comboBox.Item.Add()` 方法将商品类别管理中的商品类别名称与商品管理中的商品类别名称联系起来。其代码如下：

```

SuperMarketDataClassesDataContext db = new SuperMarketDataClassesDataContext();
dataGridView1.DataSource = db.SIMS_commodity;
var q = (from m in db.SISM_commodityclass
        select m
        );
foreach (var c in q) {

    comboBox1.Items.Add(c.SName);
}

```

图 5.2.5-1 商品管理设计视图

相较于商品类别管理，在实现上，商品管理多了一些数据类型上的转换以及实现图片类型上的显示，对于商品管理的增改查，这里不再赘述，在实现细节上与商品类别管理基本类似，对于图片的选择，首先这里使用 `openFileDialog` 控件打开文件，将图片显示到 `PictureBox` 控件前会显示对话框提示用户打开的图片路径，点击事件代码如下：

```
private void linkLabel1_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)
{
    OpenFileDialog ofd = new OpenFileDialog();
    if (ofd.ShowDialog() == DialogResult.OK)
    {
        sPicture = ofd.FileName;
        pictureBox1.Image = Image.FromFile(sPicture);
    }
}
```

其中删除按钮的实现细节代码如下：

```
private void button4_Click(object sender, EventArgs e)
{
    SuperMarketDataClassesDataContext db = new
    SuperMarketDataClassesDataContext();
    //根据id删除
    if (textBox7.Text != "") {

        var q = (from m in db.SIMS_commodity
                 where m.CId == int.Parse(textBox7.Text)
                 select m);
        foreach (var id in q)
        {
            db.SIMS_commodity.DeleteOnSubmit(id);
        }
        var num = (from m in db.SIMS_commodity
                   where m.CId == int.Parse(textBox7.Text)
                   select m).Count();
        if (num > 0)
        {
            try
            {
                db.SubmitChanges();
                MessageBox.Show("删除成功!");
                dataGridView1.DataSource = db.SIMS_commodity;
                clear();
            }
            catch { }
        }
    }
}
```

```

    }
    catch (Exception ex)
    {

    }
}
else MessageBox.Show("没有此编号的商品!");

}
else if (textBox1.Text != "")
{

//根据名称删除
var p = (from m in db.SIMS_commodity
         where m.CName == textBox1.Text
         select m);
foreach (var name in p)
{
    db.SIMS_commodity.DeleteOnSubmit(name);
}
var num = (from m in db.SIMS_commodity
           where m.CName == textBox1.Text
           select m).Count();
if (num > 0)
{
    try
    {
        db.SubmitChanges();
        MessageBox.Show("删除成功!");
        dataGridView1.DataSource = db.SIMS_commodity;
        clear();

    }
    catch (Exception ex)
    {

    }
}
else MessageBox.Show("没有此名字的商品!");
}

else

```

```

    {
        MessageBox.Show("请输入删除条件!");
    }
}

```

## 5.2.6 入库操作

出库操作与入库操作类似，这里不再过多赘述。如下图 5.2.6-1 为其设计视图。



图 5.2.6-1 入库操作设计视图

确认入库的实现代码如下：

```

private void button1_Click(object sender, EventArgs e)
{
    SuperMarketDataClassesDataContext db = new
SuperMarketDataClassesDataContext();
    SISM_console sc = new SISM_console();
    //查询是否存在该编号商品
    var num =(from m in db.SIMS_commodity
              where m.CId==int.Parse(textBox1.Text)
              select m).Count();

    //存在
    if (num > 0) {
        //查询
        var q = (from m in db.SIMS_commodity
                 where m.CId == int.Parse(textBox1.Text)
                 select m);
    }
}

```

```

foreach (var c in q)
{
    sc.Number = textBox2.Text;
    int number; //存放当前库存数目
    number = int.Parse(c.CQty) + int.Parse(sc.Number);
    c.CQty = number.ToString();
    db.SubmitChanges();
    label15.Text = "商品编号: " + c.CId + "\n" + "剩余库存: " + number
+ "\n" + "操作人: " + UserName;

    //添加开始
    sc.CId = int.Parse(textBox1.Text);
    sc.Number = textBox2.Text;
    sc.Price = textBox3.Text;
    sc.Choose = "入库";
    sc.UserName = UserName;
    db.SISM_console.InsertOnSubmit(sc);
    try
    {
        db.SubmitChanges();
        dataGridView1.DataSource = db.SISM_console;
        MessageBox.Show("入库成功!");
    }
    catch (Exception ex)
    {
    }
    //添加结束

    clean();
}
}
else MessageBox.Show("没有此编号商品!");
}

```



5.2.7 出库操作

出库操作与入库操作类似，这里不再过多赘述。如下图 5.2.7-1 为其设计视图。

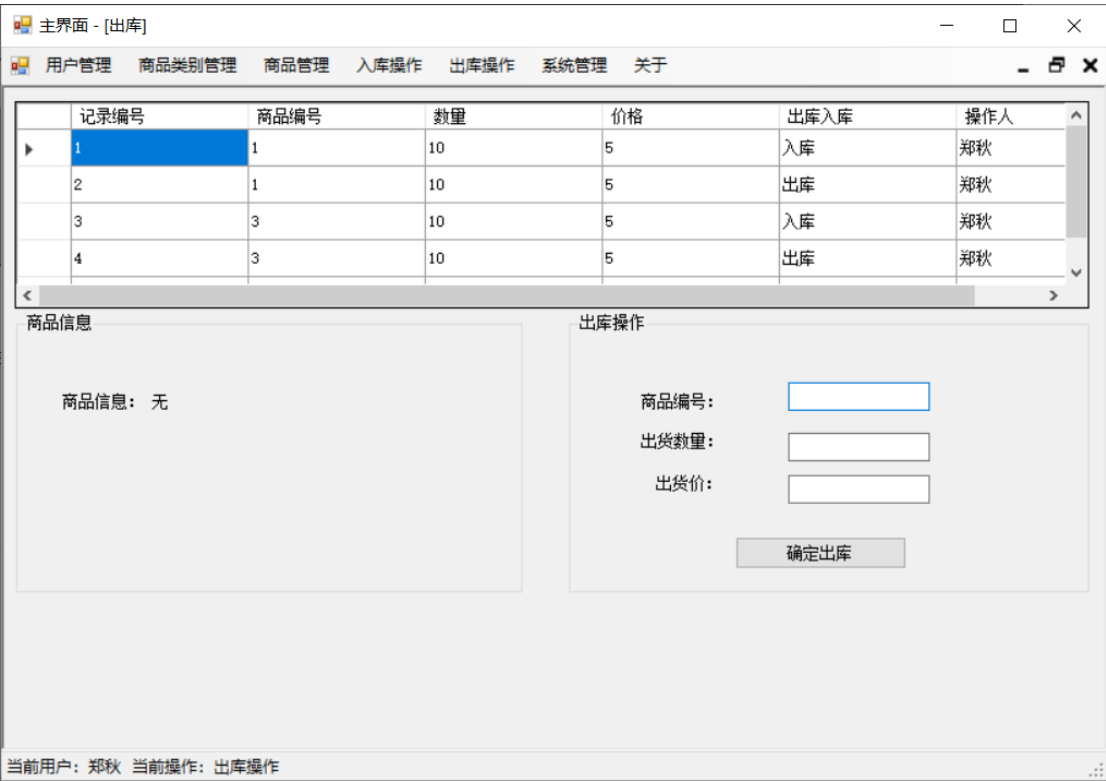
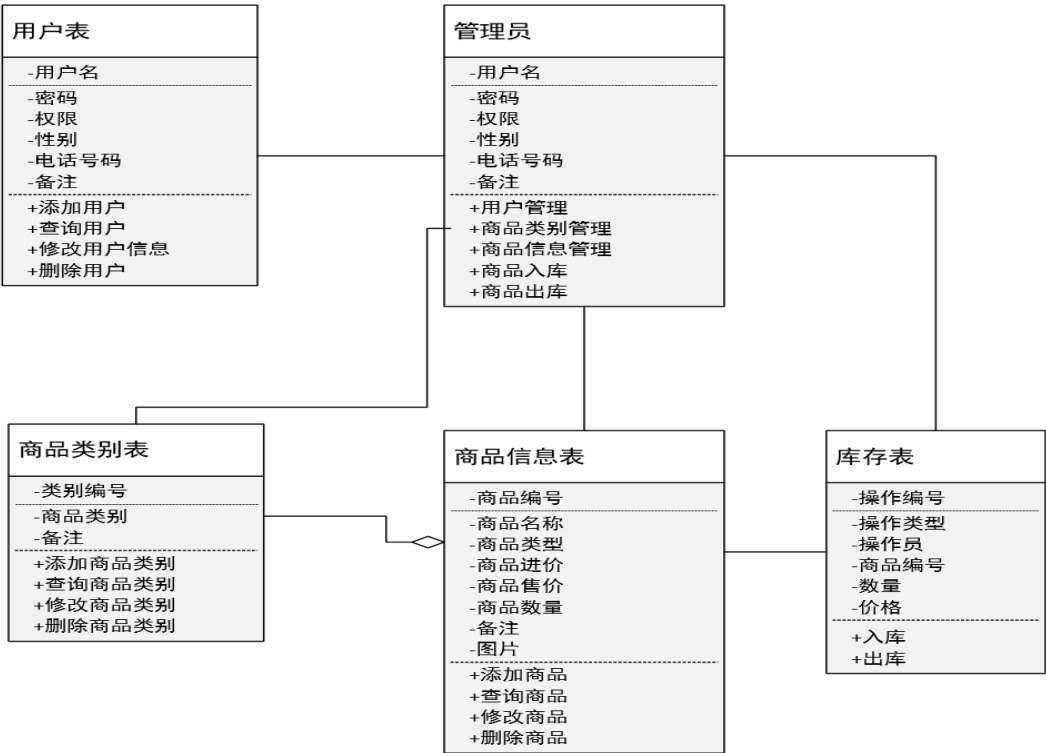


图 5.2.7-1 出库操作

5.3 系统类图



# 6. 系统测试

## 6.1 测试简介

本次功能测试采用黑盒测试方法来验证系统功能是否与系统的设计目标一致，系统的功能是否完整，程序界面是否友好和系统稳定性是否达到要求等问题。

## 6.2 功能测试

### 6.2.1 登录页面测试

(1) 首先运行程序，进入到登录页面，输入数据库“Table\_User 表”中的用户名和密码，进行登录，图 6.2.1-1 为数据库中的“Table\_User 表”，图 6.2.1-2 为用户进行登录：

	userName	userPwd	userSex	userPhone	note
▶	张三	123456	男	12345678...	用户
	郑秋	123456	男	12345678...	管理员

图 6.2.1-1 数据库中的用户表

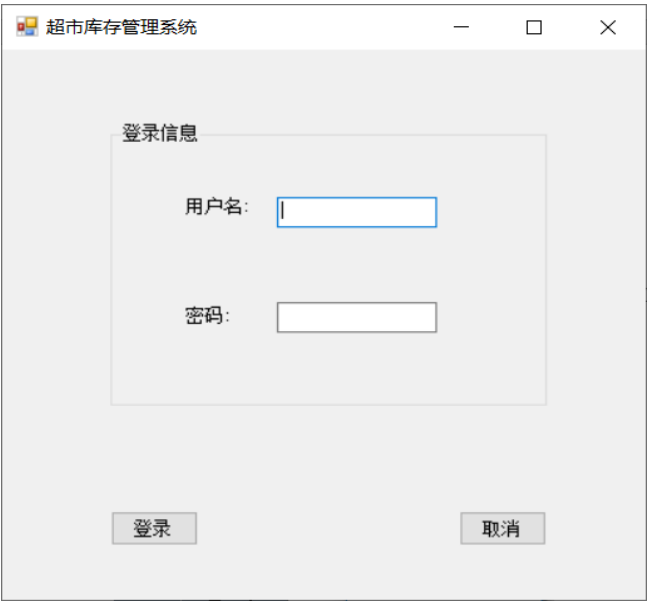


图 6.2.1-2 登录界面

(2) 点击“登录”按钮显示登录成功信息，以及提示账号信息，如图 6.2.1-3 及 6.2.1-4。



图 6.2.1-3 登录失败



图 6.2.1-4 登录成功进入主页面

## 6.2.2 用户信息管理测试

添加用户测试，添加用户信息为“用户名：李四，密码：1234，电话号码：123456789，性别：男”，如图 6.2.2-1 为添加前所有用户信息，如图 6.2.2-2 为点击确定后刷新页面后所有的用户信息，图 6.2.2-3 为数据库中的 Table\_User 表刷新之后的内容。

- 添加用户：点击“添加用户”进入到用户管理界面，添加一个用户
- 修改用户：点击“修改用户”进入到用户管理界面，修改用户
- 删除用户：点击“删除用户”进入到用户管理界面，删除用户
- 查询用户：点击“用户管理”进入到用户管理界面，查询用户

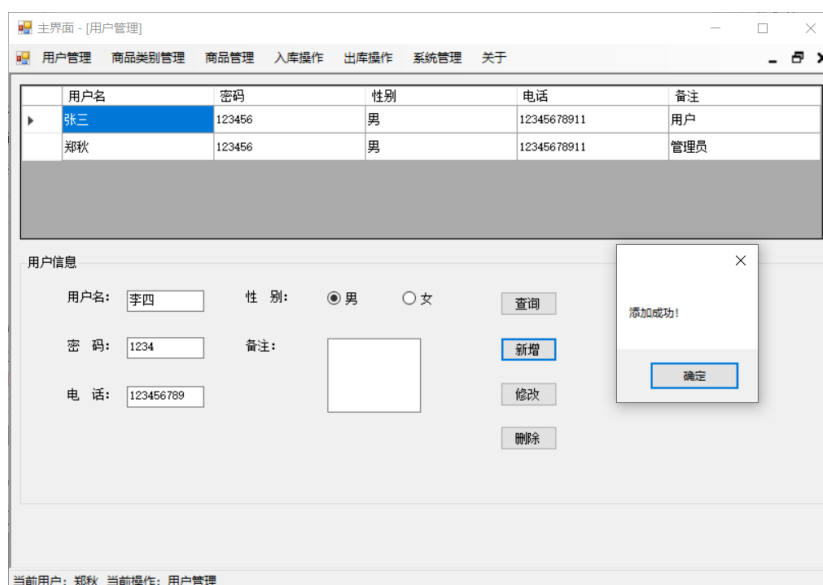


图 6.2.2-1 添加前所有用户信息

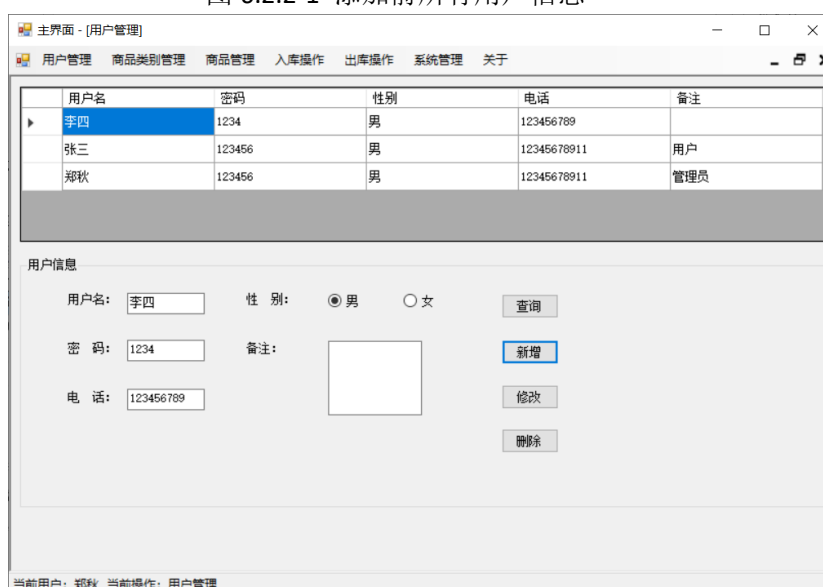


图 6. 2. 2-2 添加后所有用户信息

### 6. 2. 3 商品类别信息管理测试

(1) 添加功能测试，选择“商品类别管理”，填写商品类别信息，添加商品类别信息，类别号：4，类别名称：衣服，点击“新增”按钮，存储成功，如图 6. 2. 3-1

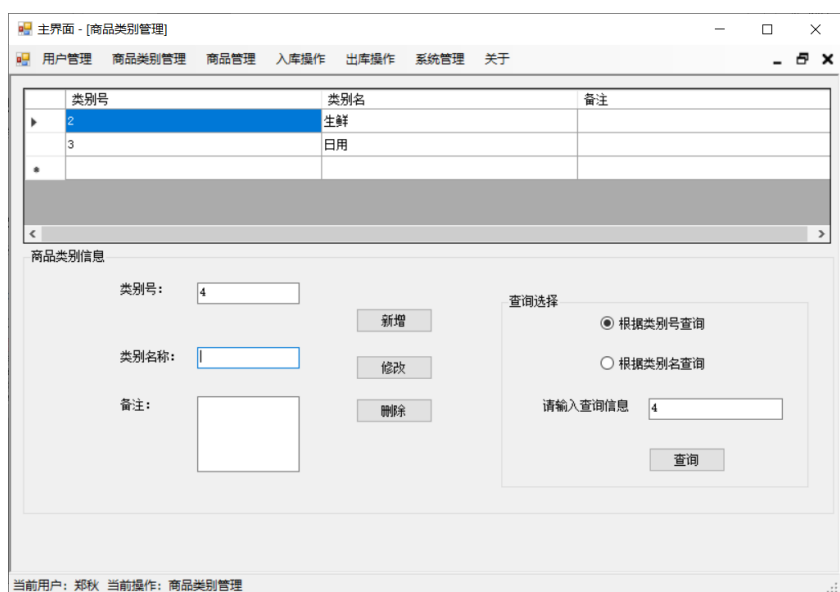


图 6.2.3-1 添加商品类别存储成功显示前

(2) 点击“确定”，将自动刷新列表，显示添加的信息，如图 6.2.3-2

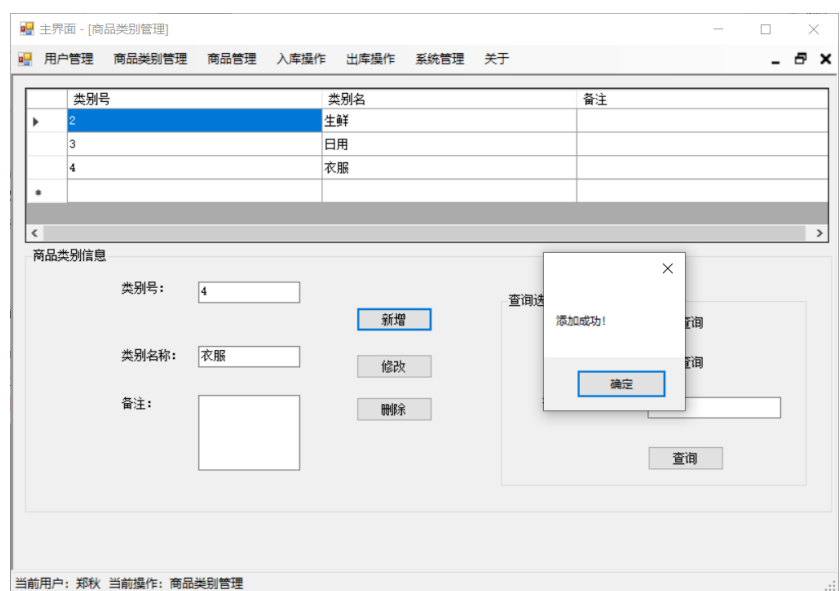


图 6.2.3-2 添加类别后

验证查询功能，以类别号查询为例，输入查询信息 4，得到结果如图 6.2.3-3。



图 6.2.3-3 查询类别

## 6.2.4 商品信息管理测试

这里我们可以验证上述商品类别的测试结果，在商品类别里面出现了“衣服”，可见“添加”是没有问题的，数据库上也有相应的数据，如图 6.2.4-1 为商品管理页面中的类别信息。

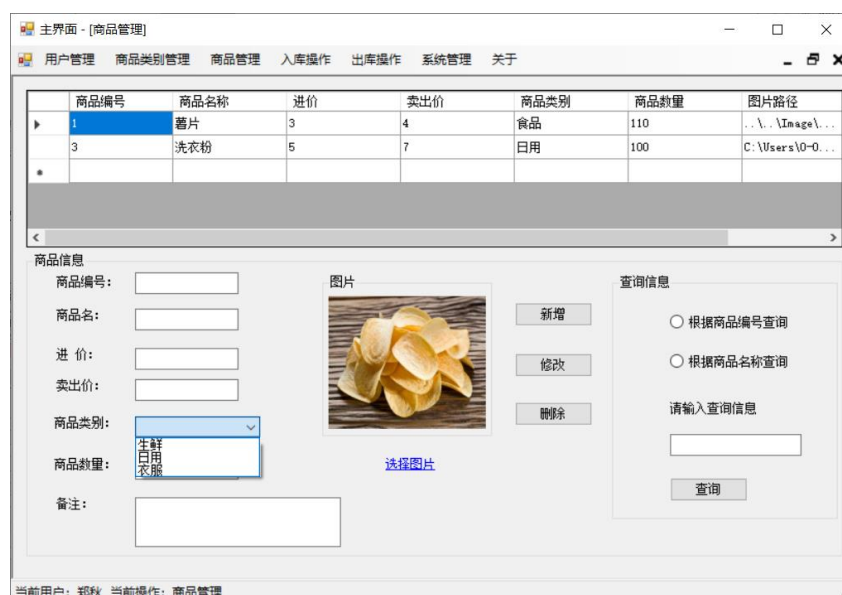


图 6.2.4-1 添加商品类别存储成功显示前



图 6.2.4-2 添加商品类别存储成功显示后

以按照商品名称查询为例，输入查询信息“薯片”，得到结果如图 6.2.4-3 所示。

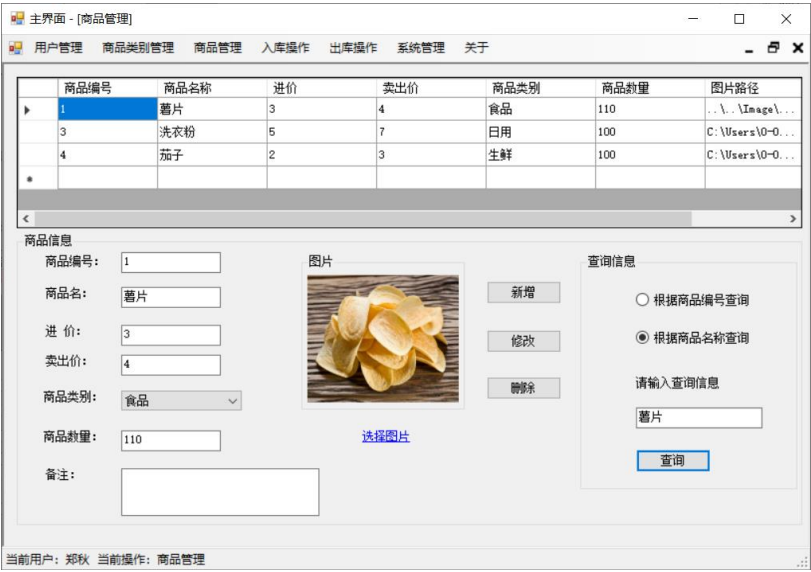


图 6.2.4-3 查询商品

验证修改商品信息，将薯片的类别修改为“生鲜”，卖出价改为“4”，进价改为6。

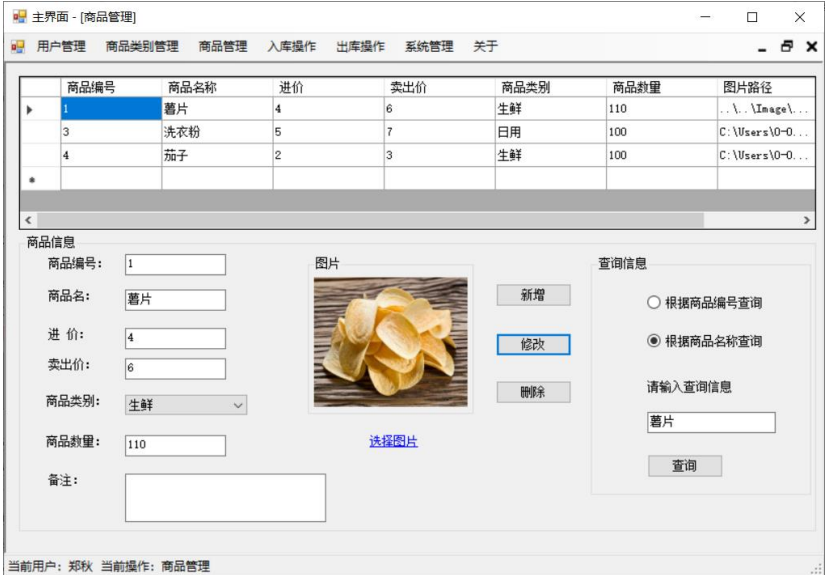


图 6.2.4-4 修改商品

验证删除商品信息，选择1号，点击删除按钮，弹出对话框，点击确定，如图 6.2.4-4 为删除信息提示，图 6.2.4-5 为删除后通过查询的结果。



图 6.2.4-5 删除前

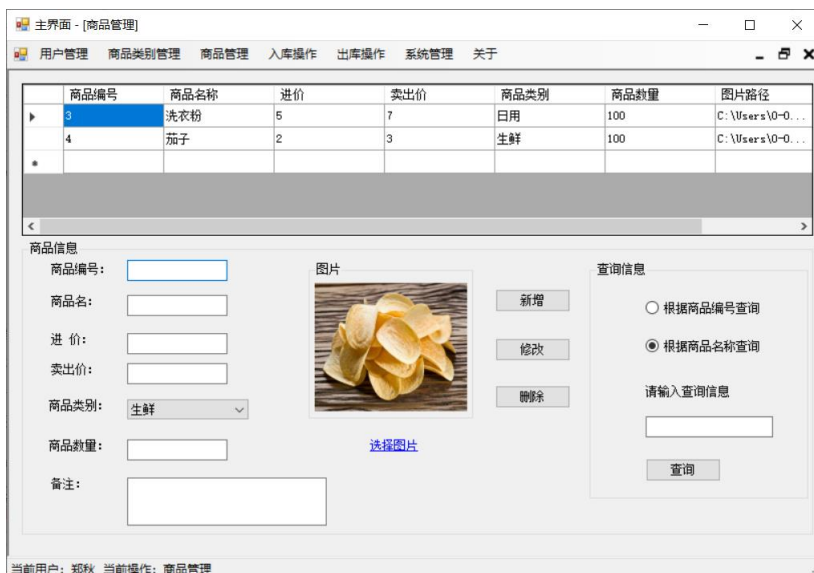


图 6.2.4-6 删除后

## 6.2.5 商品入库测试

选择“入库操作”，填写只填写编号，弹出提示信息。

填写一个数据库不存在的商品编号 1，弹出错误提示。

填写入库表，商品编号：3，进货数量 10，进货价 3，点击确认入库后自动刷新，数量自动+1 了，入库/出库变成“入库”，如图 6.2.5-1 所示。



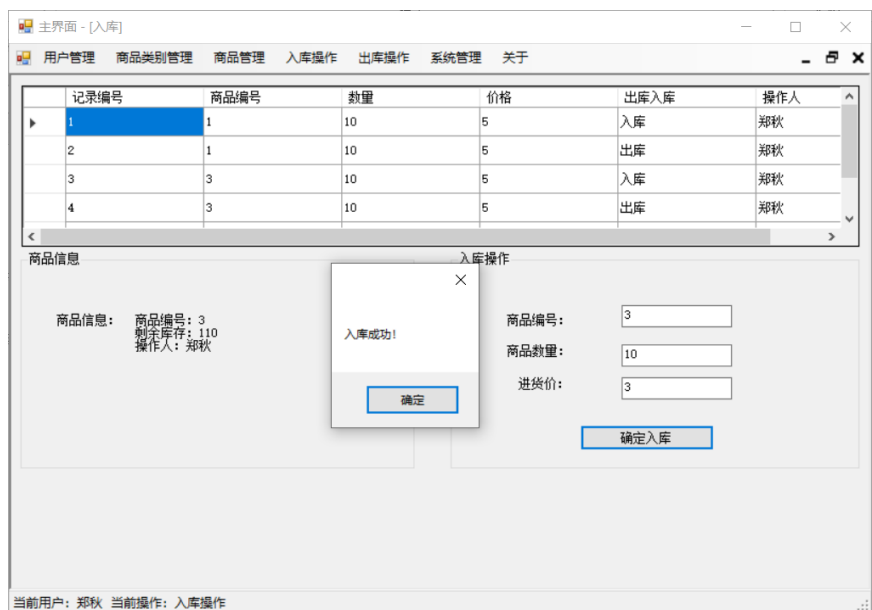


图 6.2.5-1 入库操作

## 7. 系统设计与开发结论

这次课程设计应该是大学目前为止做的最复杂的一个课设了,因为有非常多的页面要做,所以让我感受到提前构思好整个项目结构非常重要,只有把整个项目结构整理清楚,做起来才能有条不紊,一步一步去完善各个功能。

课程和设计是培养大学生的综合运用所学知识的能力解决实际问题,锻炼实践环节非常重要,随着科学技术的发展日新月异,软件工程已经成为计算机应用中非常重要的领域,因此掌握这门技术是非常重要的。

在开发的过程中也遇到了许多问题,也意识到写 Log 的好处,可以帮助我更好的判断配置的执行逻辑,也尝试看报错解决了一个问题。还很感谢在开发过程中一起探讨问题的同学,团队合作也是十分重要的能力。整个项目做完后,回过头来看真的有一种受益匪浅的感觉,也谢谢老师在平时讲授课程时的细心,让我了解到软件工程的各个细枝末节,平时学习到的知识在这个程序中基本上都用了一遍,希望以后能更多学习到有关软件工程的知识。

## 参考文献

- [1] 马骏.C#程序设计教程（第3版）.北京：人民邮电出版社，2014.
- [2] 邱珊.C#语言程序设计. 北京:科学出版出版社,2013.
- [3] 高金兰.数据库原理与 SQL Server 应用（第二版）.北京：科学出版社，2010.
- [4] 明日科技著.C#经典编程 220 例 北京：清华大学出版社 2012
- [5] 郑阿奇主编 .C#实用教程 北京：电子工业出版社 2008

课程设计成绩评定表		
课程设计题目	超市库存管理系统的设计与开发	
<p>课程设计学生答辩或质疑记录：</p> <p>1. 你是怎么连接数据库的？</p> <p>在登陆页面利用数据库账号密码连接，在其他页面利用 LINQ 方式。</p> <p>2. 如何将类别管理的类别名称反映到商品管理的 comboBox 控件中？</p> <p>利用 comboBox.item.add ( ) 方法。先选中类型所在列，再遍历列，将所有类型都加入 comboBox 中。</p> <p>3. 你觉得有什么不足之处？</p> <p>在出入库页面中没有做修改前信息的即时显示，如果要验证数量改变，需要回到商品管理页面中查看。</p>		
评 分 依 据		分 值
1. 设计方案正确，具有可行性、创新性		25 分
2. 系统开发效果较好		25 分
3. 态度认真、刻苦钻研、遵守纪律；		10 分
4. 设计报告规范、课程设计报告质量高、参考文献充分		20 分
5. 课程设计答辩概念清晰，内容正确		10 分
6. 课程设计期间的课堂考勤、答疑与统筹考虑。		10 分
总 分		100 分
<p>最终评定等级为：</p> <p style="text-align: right;">指导老师签字：_____</p> <p style="text-align: right;">2020 年 1 月 3 日</p>		