

茗品销售管理系统

组长姓名 孙子晋

组长学号 18852133

组 员 张秩，刘金宝

二零二零年 十一 月 八 日

# 目 录

[目 录 III](#_Toc42974459)

[第1章 绪论 1](#_Toc42974460)

[1.1 选题背景及意义 1](#_Toc42974461)

[1.2 国内外发展状况 2](#_Toc42974462)

[1.3 设计内容以及设计方案 2](#_Toc42974463)

[1.4 本文的组织结构 3](#_Toc42974464)

[第2章 相关技术介绍 4](#_Toc42974465)

[第3章 系统分析 7](#_Toc42974472)

[3.1 需求概述 7](#_Toc42974473)

[3.2 功能需求分析 7](#_Toc42974474)

[3.3 本章小结 16](#_Toc42974482)

[第4章 系统设计 17](#_Toc42974483)

[4.1 体系架构设计 17](#_Toc42974484)

[4.2 数据库设计 22](#_Toc42974497)

[4.3 本章小结 28](#_Toc42974500)

[第5章 系统实现 29](#_Toc42974501)

[第6章 系统测试 46](#_Toc42974510)

[第7章](#_Toc42974522) [总结和展望 52](#_Toc42974522)

[参考文献 53](#_Toc42974525)

[致 谢 54](#_Toc42974526)

# 第1章 绪论

随着社会的经济开展，网络的遍及，当代的大学计算机教育已经从偏重于理论教学的倾向转变成了更加重视实际的操作了。其中，计算机软件的应用已经影响到了多种产业的方方面面，比如一些实体销售产业，建立自己的销售系统已经成为了必不可少的一环。

## 选题背景及意义

茶饮料所属的饮品行业不停地发展，各式各样的产品层出不穷，而茶饮料因为 种类繁多，口味多变，发展得更是迅猛。随着茶饮料的出现，茶饮料迅速风靡全国，各种茶饮料店也如雨后春笋般出现在城市的大街小巷。茶饮料经过十几年的演变发展，消费者对于奶茶的消费更加趋于理性，也更注重茶饮料的品质。这使得茶饮料店的发展有所放缓，投资者对于茶饮料的品质也有了更多的考究。

## 国内外发展状况

据市场调查的最新研究，2019年国际上前十大茶饮料销售国家和地区的茶饮料销售量达到3003.50万t，销售额达到404.40亿美元。亚洲是全球即饮茶市场的中心，亚太地区即饮茶消费占全球72%的份额。其中，中国是世界上最大的茶饮料生产国和消费国，2019年茶饮料消费量达到1335.6万t，其次是日本;印度尼西亚和越南是目前茶饮料消费增长最快的国家。

## 设计内容以及设计方案

在开发管理系统时，根据用户的设定来实现，依据各种渠道对资源进行收集、整理和分类，向用户提供和推荐相关信息，能够充分利用各种资源优势，主动开展以满足用户个性化需求为目的的设计。

## 本文的组织结构

销售管理系统主要为茗品店的日常营运、决策管理提供信息服务，它覆盖了茗品店的库存、销售等各个单元，涉及所有业务职能，系统能够对用户管理、售价管理、库存数量管理等等方面提供存储、更新、查询、统计汇总等功能，茗品店中商品及供货商等信息由管理人员在系统的仓库管理模块录入和管理。管理人员在系统管理单元负责管理，库存商品资料，上、下架信息由库存管理人员在系统库存单元负责管理。

# 第2章 相关技术介绍

## 2.1 Visual Studio 2012 和 SQL Server 2008 R2工具

## 2.2 C#编程语言简介

C#是一种最新的、面向对象的编程语言。它使得程序员可以快速地编写各种基于Microsoft [.NET](https://baike.baidu.com/item/.NET" \t "https://baike.baidu.com/item/C%23%E8%AF%AD%E8%A8%80/_blank)平台的应用程序，Microsoft .NET提供了一系列的工具和服务来最大程度地开发利用计算与通讯领域。

正是由于C#面向对象的卓越设计，使它成为构建各类组件的理想之选——无论是高级的商业对象还是系统级的应用程序。使用简单的C#语言结构，这些组件可以方便的转化为[XML](https://baike.baidu.com/item/XML" \t "https://baike.baidu.com/item/C%23%E8%AF%AD%E8%A8%80/_blank) 网络服务，从而使它们可以由任何语言在任何操作系统上通过Internet进行调用。

最重要的是，C#使得C++程序员可以高效的开发程序，而绝不损失C/C++原有的强大的功能。因为这种继承关系，C#与C/C++具有极大的相似性，熟悉类似语言的开发者可以很快的转向C#。

## 2.3 SQL Server数据库简介

SQL Server是一个可扩展的、高性能的、为分布式客户机/服务器计算所设计的数据库管理系统，实现了与WindowsNT的有机结合，提供了基于事务的企业级信息管理系统方案。

## 2.4 本章小结

简单介绍了使用的工具以及它们基本功能用途，优缺点等等。

# 第3章 系统分析

本设计以方便消费者和管理者提高管理效率为目的，来设计其功能。系统首要考虑的部分应该是系统需求分析，而它的好坏会影响该商品销售平台后期框架和开发。这一部分能够让开发者对平台设计有一些更深的了解而系统的开发框架、功能、流程方案都需要通过需求分析来进行确定。

## 3.1 需求概述

要开发一个功能比较完整，具有使用价值的一个系统，首先要分析其使用该商品管理系统的时候有什么意义。我们首先把自己想成是老板，我们想用这个系统实现什么样的价值，如何最大效益的方便自己，提高商品的售卖效率。其次站在消费者的角度，想要用该系统实现怎样的快速便捷，节约消费者的等待时间。本章对消费者购买商品的需求进行整理分析。把老板使用是否方便作为重点之一，其次是使用这个系统是否有利于提高商品售卖和购买效率。所以，我们对各个模块进行具体分析。

## 3.2 功能需求分析

### 3.2.1 购买商品功能分析

本设计主要信息通过数据库进行采集，在本设计中所用到的数据库是SQL Server，当用户登录之后，可以看到数据库提供的商品属性，通过选择商品属性可以购买商品，最后付款。

当管理员登录成功后，可以注销用户和修改用户的权限。接着进入仓库管理，数据库提供仓库中的剩余商品信息，管理员可以通过管理页面进行上下架商品和修改数据属性。

当用户登录成功后，选择商品信息管理，就可以看到每个商品信息，如表3-1所示。

表3-1 购买信息用例说明表

| 用例名称 | 查看商品信息 |
| --- | --- |
| 执行者 | 用户 |
| 前置条件 | 用户登录成功 |
| 后置条件 | 用户对购买功能进行操作，保存结果。 |
| 基本流程 | （1）在系统首页，点击用户登陆。  （2）看到商品属性信息。  （3）选择商品，点击购买。  （4）系统调取该用户保存的所有操作结果，并显示在相应功能页面。 |

### 3.2.2 管理仓库功能分析

本设计主要信息通过数据库进行采集，在本设计中所用到的数据库是SQL Server，当管理员登录成功后，可以注销用户和修改用户的权限。接着进入仓库管理，数据库提供仓库中的剩余商品信息，管理员可以通过管理页面进行上下架商品和修改数据属性。如表3-2所示。

表3-2 管理仓库信息用例说明表

| 用例名称 | 查看商品信息 |
| --- | --- |
| 执行者 | 管理员 |
| 前置条件 | 管理员登录成功 |
| 后置条件 | 管理员对管理仓库功能进行操作，保存结果。 |
| 基本流程 | （1）在系统首页，点击前往仓库。  （2）看到仓库中的基本商品信息。  （3）上下架商品或修改商品属性。  （4）系统调取该管理员保存的所有操作结果，并显示在相应功能页面。 |

## 3.3 本章小结

本章对系统的功能模块进行分析，展现了该系统功能模块图和各功能模块的用例图。为商品销售系统之后的设计和实现打下了基础。

# 第4章 系统设计

## 4.1 体系架构设计

本小节主要是描述对茗品销售管理系统使用的整体功能结构以及具体功能的具体设计内容。系统设计体系结构主要是系统开发的主要部分，可以理解为系统的真个模型，体系结构关系到系统的核心部分，也是同以往系统设计有很大的区别，这对于系统软件设计有着很大的作用，特别是系统的设计工作者。

### 4.1.1 整体架构设计

[对茗品销售管理系统](javascript:;)使用的整体功能结构以及具体功能的具体设计内容是本小节主要的描述。对使用的技术框架和功能结构设计是在本章节中的主要任务。而在开发过程中需要完成的任务是通过对整个设计过程的梳理以及关键技术的介绍描述。

系统是通过Visual Studio 2012设计实现的，具体内容表现为用户登录成功后进入界面进行操作。

### 4.1.2 功能架构设计

在完成对系统中功能进行结构设计的任务中首先需要搞定的是对本系统首先应该实现的功能以及功能之间的关联性去进行实现。

通讯用户模块和管理员模块是本平台的主要具体表现。

本系统的功能结构设计如图4-1所示。

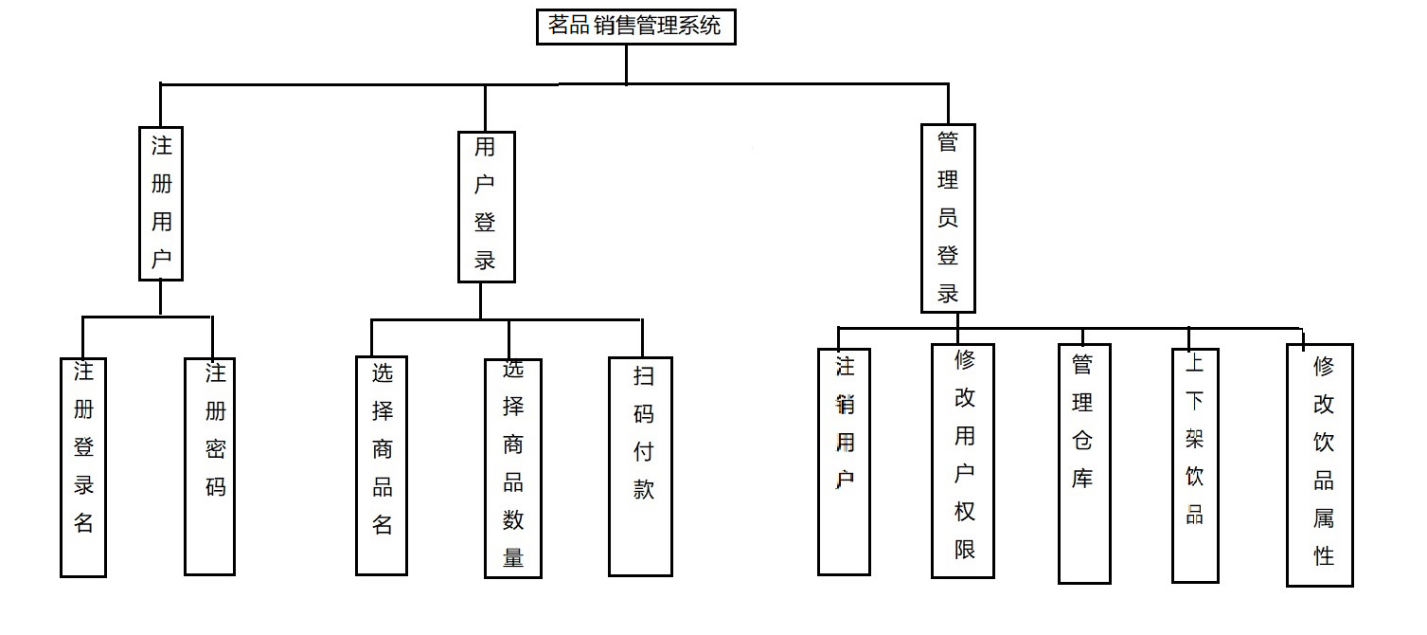


图4-1 功能模块图

## 4.2 数据库设计

数据库设计能够体现系统的各个属性的具体属性值，可以结合对于每个功能模块进行分析，进而制定相应的数据库表结构，对应每个功能模块，能够构建表的关系，通过数据库管理来协调功能模块的调试过程。

### 4.2.1 逻辑结构设计

对该系统所用到的表进行详细介绍展示，以E-R图和表格的方式。

所有角色用户表，主要属性包括了用户id、当前状态、用户密码等几个字段。具体结构如图4-2所示。

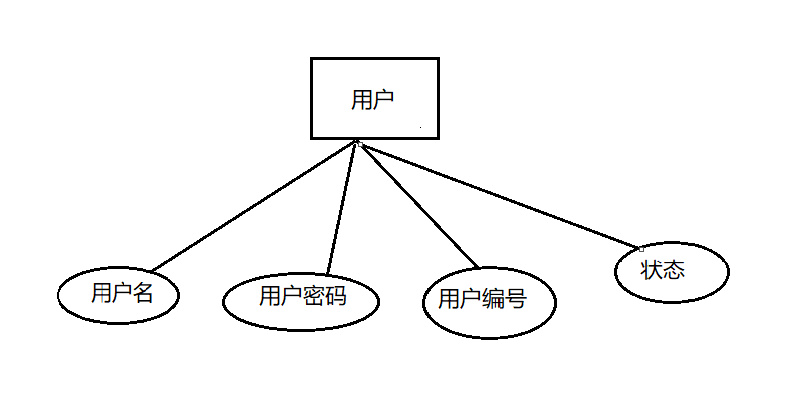


图4-2 用户实体属性图

用于存储该表的数据信息如表4-3所示。

表4-3 用户信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段 | 描述 | 类型和长度 | 主键 | 是否可空 |
| 1 | name | 用户名 | varchar(20) | 否 | 不允许 |
| 2 | id | 编号 | int | 是 | 不允许 |
| 3 | type | 当前状态 | varchar(10) | 否 | 不允许 |
| 4 | password | 密码 | varchar(10) | 否 | 不允许 |

所有商品表，主要属性包括了商品id、商品名、商品数量、商品价格等几个字段。具体结构如图4-4所示。

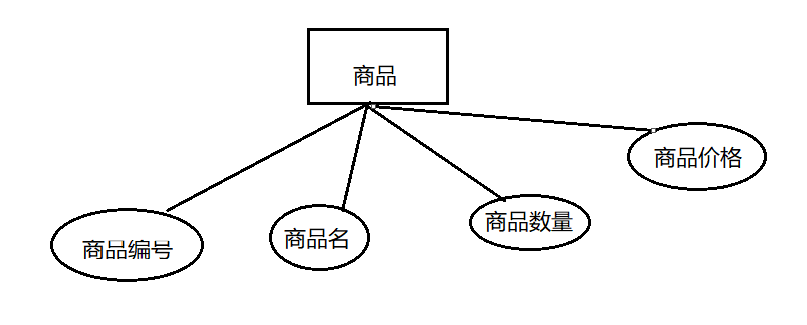


图4-4 商品实体属性图

用于存储该表的数据信息如表4-5所示。

表4-5 商品信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段 | 描述 | 类型和长度 | 主键 | 是否可空 |
| 1 | name | 商品名 | varchar(20) | 否 | 不允许 |
| 2 | id | 商品编号 | int | 是 | 不允许 |
| 3 | count | 商品数量 | int | 否 | 不允许 |
| 4 | price | 商品价格 | money | 否 | 不允许 |

## 4.3 本章小结

介绍了本系统的商品和用户的具体结构、存储信息，以及功能结构设计图。

# 第5章 系统实现

茗品销售管理系统的设计与实现的各模块功能实现结果截图为本章所描述的主要内容。

## 5.1 注册用户模块功能实现

用户点击登陆页面的注册按钮，然后通过输入用户名和密码注册新用户。

**用户注册功能代码：**

private void button1\_Click(object sender, EventArgs e)

{

DBA dba = new DBA();

string Na = txtlgname.Text;

string PWD = txtlgpasswd.Text;

string PWD1 = txtlgpasswd1.Text;

string sql = "insert into User\_table values('" + Na + "','" + PWD + "','user')";

//bug调整

if (Na == "" || PWD == "" || PWD1 == "")

{

MessageBox.Show("亲，数据不能为空哦！");

}

else if (!PWD.Equals(PWD1))

{

MessageBox.Show("两次密码不一致，请检查并重新输入！");

return;

}

else

{

dba.nonSelect(sql);

MessageBox.Show("注册成功！");

dba.close();

}

//跳转至登录页面

this.Visible = false;

Form1 fr = new Form1();

fr.Visible = true;

}

其效果如图5-1所示：

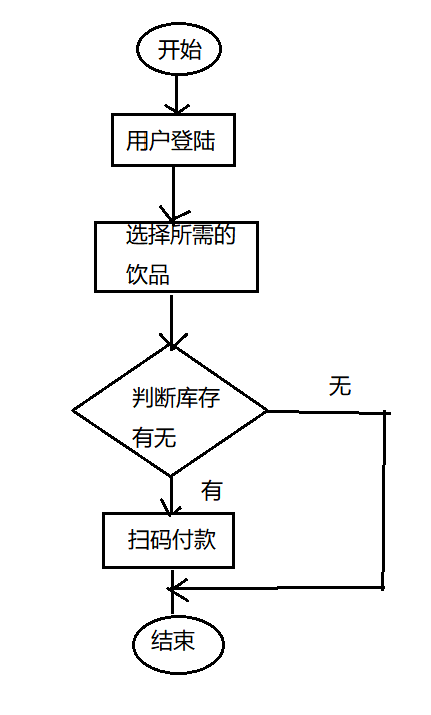


图5-1 用户注册效果图

## 5.2 用户购买模块功能实现

用户要想进入本系统之后，点击相应操作可以进入到不同功能的实现页面。

购买功能是每个登陆者都可以看到的一个部分，是为了方便每个用户可以快速的购买商品。进入该模块，直接展示所有商品的信息，可以在选择框选择想购买的物品，其流程图如图5-2所示。



*图5-2 通讯查找流程图*

通过用户名查询条件，用用户信息访问数据库，然后点击登录，进入到购买商品模块。

然后通过选择商品的商品名和数量，点击购买，进入到付款模块。

**登陆页面代码：**

private void button1\_Click(object sender, EventArgs e)

{

DBA dba = new DBA();

string loginname = txtlogin.Text;

string passwd = txtpasswd.Text;

string sql = "select \* from User\_table where name='" +loginname+ "' and password='" +passwd+ "'";

SqlDataReader sdr = dba.selectOneRow(sql);

if (sdr.Read())

{

MessageBox.Show("登录成功！欢迎-" + sdr["name"] + "-登录");

string n= sdr.GetString(sdr.GetOrdinal("type"));//获得数据库中的登录类型

if (n == "root")//进入管理员界面

{

this.Visible = false;

superuser su = new superuser();

su.Visible = true;

}

else//进入普通用户界面

{

this.Visible = false;

user us = new user();

us.Visible = true;

}

}

else

{

MessageBox.Show("登录失败！用户名或密码有误！");

}

dba.close();//关闭连接

}

**进入到用户模块代码：**

private void button2\_Click(object sender, EventArgs e)

{

//信息框（对话框）

DialogResult dr = MessageBox.Show("是否确定购买此商品?", "标题", MessageBoxButtons.YesNo, MessageBoxIcon.Warning);

if (dr == DialogResult.Yes)

{

string NA = comboBox1.Text;

string co = comboBox2.Text;

int CO = Convert.ToInt32(co);//要买的奶茶数

string sql = "select \* from Object\_table where name='" + NA + "'";

DBA dba = new DBA();

SqlDataReader sdr = dba.selectOneRow(sql);

if (sdr.Read())

{

int c = sdr.GetInt32(sdr.GetOrdinal("count")); //获得数据库中奶茶的数量

string sql2 = "update Object\_table set count=count-'" + CO + "' where name='" + NA + "'";

if (CO > c || c <= 0)

{

MessageBox.Show("库存不足，无法购买！");

}

else

{

DBA da = new DBA();

da.nonSelect(sql2);

da.close();

this.Visible = false;

buy bu = new buy();

bu.Visible = true;

}

}

dba.close();//关闭连接

}

}

**付款页面代码：**

private void button1\_Click(object sender, EventArgs e)

{

MessageBox.Show("购买成功！");

this.Visible = false;

user us = new user();

us.Visible = true;

}

其效果如图5-3所示：





图5-3 用户登录购买商品效果图

## 5.3 管理员管理模块功能实现

管理员要进入本系统之后，点击相应操作可以进入到不同功能的实现页面。

管理员进入该模块，首先可以通过填写基本的用户信息注销用户，并且修改用户的属性。然后可以进入管理仓库模块，可以通过填写原有的商品属性信息下架商品，也可以通过填写新商品属性信息上架新的商品，还可以修改原有的商品属性。其流程图如图5-4所示。

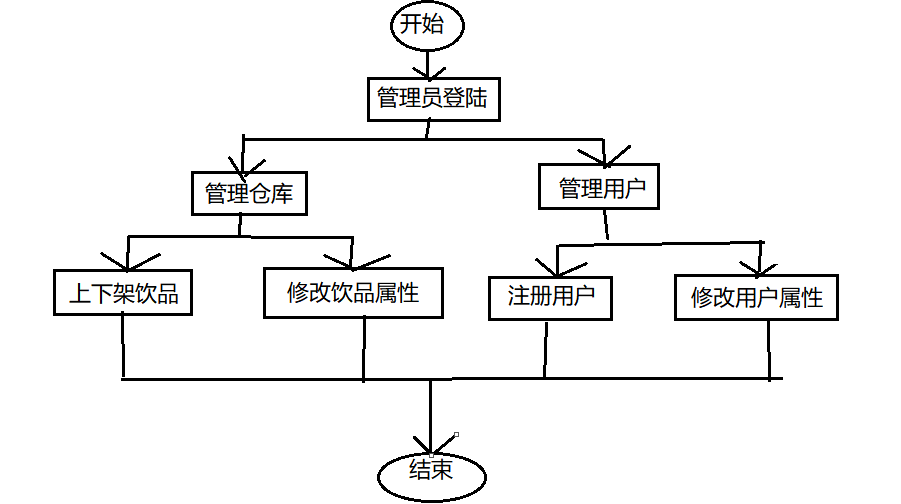


图5-4 管理员管理流程图

**商品上下架功能代码：**

private void button2\_Click(object sender, EventArgs e)

{

string name = textBox1.Text;

string count1 = textBox2.Text;

int count = Convert.ToInt32(count1);

string price1 = textBox3.Text;

double price = Convert.ToDouble(price1);

if (radioButton1.Checked)//上架功能实现

{

string sql = "insert into Object\_table values('" + name + "','" + count + "','" + price + "')";

DBA dba = new DBA();

dba.nonSelect(sql);

MessageBox.Show("上架成功！");

dba.close();

refresh();//列表刷新

}

else if (radioButton2.Checked)//下架功能实现

{

string sql = "select \* from Object\_table where name='" + name + "'";

DBA dba = new DBA();

SqlDataReader sdr = dba.selectOneRow(sql);

if (sdr.Read())

{

DBA dda = new DBA();

string sql2 = "delete Object\_table where name='" + name + "'";

dda.nonSelect(sql2);

MessageBox.Show("下架成功！");

dda.close();

refresh();//列表刷新

}

else

{

MessageBox.Show("下架失败！该饮品不存在！");

}

dba.close();

}

else

{

MessageBox.Show("请选择‘上架’或‘下架’选项！");

}

}

**商品修改属性功能代码：**

private void button3\_Click(object sender, EventArgs e)

{

string NA = comboBox2.Text;

string name = textBox4.Text;

string count = textBox5.Text;

string price = textBox6.Text;

//bug调整

if (name == "" || count == "" || price == "")

{

MessageBox.Show("亲，数据不能为空哦！");

}

else

{

string sql = "update Object\_table set name='" + name + "' ,count='" + count + "' ,price='" + price + "' where name='" + NA + "'";

DBA ed = new DBA();

ed.nonSelect(sql);

MessageBox.Show("修改成功！");

ed.close();

refresh();//列表刷新

}

}

其效果如图5-5所示：

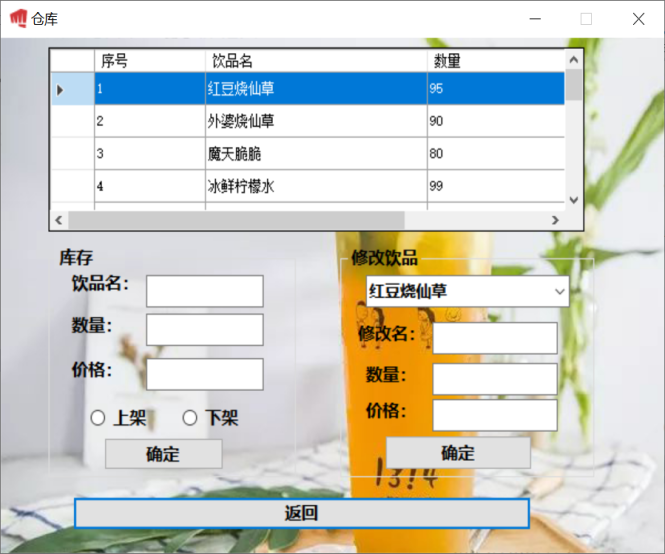


图5-5 通讯信息效果图

## 5.8 本章小结

介绍了茗品销售管理系统的各个功能模块流程图，代码实现，以及所展示出的效果图。

# 第6章 系统测试

系统测试其实就是对已经开发的系统进行验证的过程，需要对系统的各个模块进行数据测试，一般情况下，通过对模块的测试能够及时发现系统设计过程中由于程序员的疏忽导致的潜在问题，这样在没有交给用户的前提下及时的更正错误，一是对客户负责的态度，二是也可以减小因为系统问题导致的风险。所以，系统测试环节是必须要有的环节，为了系统的更加稳定，需要仔仔细细的对系统进行一番测试，使其达到设计要求，给客户一个满意的成果。

## 6.1 测试原则

1、测试工作应避免由原来开发软件的个人和小组承担。测试工作应由专门人员来进行，会更客观、更有效。

2、设计测试用例不仅要包括有效合理的输入数据，也要包含不合理、无效的输入数据。

## 6.2 功能测试

### 6.2.1 商品购买模块功能测试

商品购买模块测试，首先是购买的实现。其次是商品与详细信息的匹配是否正确，然后是对商品购买进行一个测试，看能否成功的跳转到扫码购买页面。

本次主要对购买商品进行测试，测试内容为，购买成功或者购买失败所显示的结果。

如表6-1所示。

表6-1 商品购买功能测试用例表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试名称 | 测试数据 | 测试输入 | 预期结果 | 实际结果 |
| 商品购买模块测试 | 输入空商品名 | null | 购买失败 | 购买失败 |
| 输入不存在商品名 | aaaa | 购买失败 | 购买失败 |
| 输入存在商品名 | 柠檬红茶 | 购买成功 | 购买成功 |

### 6.2.2 管理员管理模块功能测试

管理员管理模块功能测试，首先是注销用户和修改用户权限的实现。其次是上下架商品和修改商品属性测试，看看能否实现功能。

本次主要对管理用户进行测试，测试内容为，注销用户和修改用户权限成功或失败所显示的结果。

如表6-2所示。

表6-2 管理仓库功能测试用例表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试名称 | 测试数据 | 测试输入 | 预期结果 | 实际结果 |
| 管理仓库模块测试 | 输入空商品名 | null | 下架失败 | 下架失败 |
| 输入不存在商品名 | aaaa | 下架失败 | 下架失败 |
| 输入存在商品名 | 柠檬红茶 | 下架成功 | 下架成功 |
| 输入空商品名 | null | 上架失败 | 上架失败 |
| 输入不存在商品名 | 柠檬水 | 上架成功 | 上架成功 |
| 输入空商品名 | null | 修改失败 | 修改失败 |
| 输入不存在商品名 | aaaa | 修改失败 | 修改失败 |
| 输入存在商品名 | 柠檬红茶 | 修改成功 | 修改成功 |

本次主要对管理仓库进行测试，测试内容为，上下架商品和修改商品属性成功或失败所显示的结果。

如表6-3所示。

表6-3 管理用户功能测试用例表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试名称 | 测试数据 | 测试输入 | 预期结果 | 实际结果 |
| 管理用户模块测试 | 输入空用户名 | null | 注销失败 | 注销失败 |
| 输入不存在用户名 | aaaa | 注销失败 | 注销失败 |
| 输入存在用户名 | 张三 | 注销成功 | 注销成功 |
| 输入空用户名 | null | 修改失败 | 修改失败 |
| 输入不存在用户名 | aaaa | 修改失败 | 修改失败 |
| 输入存在用户名 | 张三 | 修改成功 | 修改成功 |

## 6.3 性能测试

本章节性能测试主要是为了用户在使用该系统的时候能够满足所有用户所要求的性能标准，在系统中实现各个功能的响应时间以及效率甚至说是否有无bug等一些问题。主要分为几个方面，描述如下：

（1）在Visual Studio 2012中启动后端代码，在启动过程中查看是否能够启动成功，是否启动的顺畅。

（2）需要对数据库服务状态进行开启，是否开启顺利。

（3）在整个平台运行的过程中关注Visual Studio 2012 中控制台查看是否有一些bug，以及在展示页面中是否存在调用失败或者返回内容不完整等情况。

（4）最后，是关于硬件方面的要求，就是看CPU的利用效果是否很低，关于内存方面的使用空间是否很少，还有磁盘和网络的速率是不是也不高。

## 6.4 本章小结

测试了系统的用户购买功能，管理员管理用户的功能和管理员管理仓库的功能能否成功实现。

# 第7章 总结和展望

## 7.1 总结

茗品销售管理系统从普通用户和管理员的角度，设计了不同的功能，既提高了消费者购买的效率，也提高了管理者管理用户和仓库的效率。

## 7.2 展望

管理系统的开发与实施不是一个纯技术工程，而是一个涉及企业产，供，销，存，财各项经营活动的企业行为。如何在企业管理信息系统建设中体现的先进管理哲理和功能，需要开发商提供全方位的服务。全方位的服务包括:方案咨询，提供软件，培训实施，用户化开发和可持续改进。因此。管理系统建设会越来越向模块化，人性化，个性化的方向发展。信息系统开发体现着企业管理思路。最佳业务实践是成功企业的管理模式和业务模式中提炼出的精华，应体现到信息管理软件中去。这种软件往往大而全，企业在实施时一般要先做业务流程重组和管理创新，然后再进行管理信息系统的安装调试。管理信息系统开发应按照不同的功能分成很多很小的模块，用户可以自由选择，并随时追加。

随着社会的进步，管理系统也延伸到了更多的领域。只要立足系统本身，并有序地开拓创新，就能取得长足的发展。基于此，我们有理由相信管理系统在未来会有更广阔的天地。

# 参考文献

# 致 谢