三、详细设计说明书

[1．引言 1](#_Toc506973482)

[1.1编写目的 1](#_Toc506973483)

[1.2项目背景 1](#_Toc506973484)

[1.3定义 2](#_Toc506973485)

[1.4参考资料 2](#_Toc506973486)

[2．总体设计 2](#_Toc506973487)

[2.1需求概述 2](#_Toc506973488)

[2.2软件结构 2](#_Toc506973489)

[3．程序描述 2](#_Toc506973490)

[3.1功能 3](#_Toc506973491)

[3.2性能 3](#_Toc506973492)

[3.3输入项目 3](#_Toc506973493)

[3.4输出项目 3](#_Toc506973494)

[3.5算法 3](#_Toc506973495)

[3.6程序逻辑 3](#_Toc506973496)

[3.7接口 3](#_Toc506973497)

[3.8存储分配 3](#_Toc506973498)

[3.9限制条件 3](#_Toc506973499)

[3.10测试要点 3](#_Toc506973500)

# 1．引言

## 1.1编写目的

【阐明编写详细设计说明书的目的，指明读者对象。】

为本系统开发人员、测试人员和使用人员作为参考。

## 1.2项目背景

【应包括项目的来源和主管部门等。】

委托单位：公寓内小型超市

开发者：葛成

该软件是商品交易软件的后台，为其提供商品管理信息。

## 1.3定义

【列出文档中所用到的专门术语的定义和缩写词的原文。】

## 1.4参考资料

【列出有关资料的作者、标题、编号、发表日期、出版单位或资料来源，可包括：

1. 项目的计划任务书、合同或批文；
2. 项目开发计划；
3. 需求规格说明书；
4. 概要设计说明书；
5. 测试计划（初稿）；
6. 用户操作手册（初稿）；
7. 文档中所引用的其他资料、软件开发标准或规范。】

# 2．总体设计

## 2.1需求概述

实现一个小型的商品管理系统，提供对商品管理的最基本的操作。

## 2.2软件结构

【如给出软件系统的结构图。】



主结构：

管理系统

商品录入

商品删除

价格修改

库存管理

功能模块:商品录入

商品录入

输入新的商品及相关信息

录入数据库

功能模块:商品删除

商品删除

删除商品及相关信息

录入数据库

功能模块:商品价格修改

商品价格修改

输入新的商品价格信息

录入数据库

功能模块:商品库存管理

库存管理

输入商品数目的相关信息

录入数据库

# 3．程序描述

【逐个模块给出以下的说明：】

添加商品信息：主要功能添加新进的商品。

修改商品信息：主要功能是修改商品的信息，包括它的库存数量。

删除商品信息：主要功能是删除不再销售的商品信息。

修改商品库存：主要功能是修改商品的库存量。

## 3.1功能

**1.系统登录模块**

功能：本模块的主要功能是对用户身份进行，验证只有系统的合法用户才能进入系统。在进行系统登录过程中，登录模块将调用数据库里的用户清单，并对账号和密码进行验证，只有输入了正确的账号和密码后，系统登录才会成功。

代码：import java.awt.Dimension;

import java.awt.Graphics;

import java.awt.GridBagConstraints;

import java.awt.GridBagLayout;

import java.awt.Insets;

import java.awt.Toolkit;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import javax.swing.ImageIcon;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

import javax.swing.JPasswordField;

import javax.swing.JTextField;

public class LoginJFrame implements ActionListener

{

public JFrame jframe;

public JLabel jlabel, jlabel1;

public GridBagLayout gridbag;

public GridBagConstraints constraints;

public JTextField jtfield1;

public JPasswordField jpfield1;

public JButton jbutton1, jbutton2, jbutton3;

public JPanel jpanel;

public Connection con;

public Statement sql; //声明Statement对象

public ResultSet rs;

public static void main(String[] args) {

new LoginJFrame();

}

public LoginJFrame(){

jframe = new JFrame();

jframe.setTitle("登陆");

jlabel = new JLabel();

jlabel.setText("用户名：");

jlabel1 = new JLabel();

jlabel1.setText("密 码：");

jtfield1 = new JTextField(7);

jpfield1 = new JPasswordField(7);

gridbag = new GridBagLayout();

jbutton1 = new JButton();

jbutton1.setText("登 录");

jbutton2 = new JButton();

jbutton2.setText("退 出");

jbutton3 = new JButton();

jbutton3.setText("帮 助");

jpanel = new JPanel()

{

public void paintComponent(Graphics g) {

super.paintComponent(g);

ImageIcon img = new ImageIcon("src/img/j01.jpg");

g.drawImage(img.getImage(), 0, 0, null);

}

};

//设置窗体属性

Dimension sr=Toolkit.getDefaultToolkit().getScreenSize();

jframe.setBounds((sr.width-460)/2, (sr.height-340)/2, 460, 340);

jframe.setDefaultCloseOperation(3);

// 设置JPanel为透明，且使用GridBagLayout布局管理器

// jpanel.setOpaque(true);

jpanel.setLayout(gridbag);

// 用户名文本框显示

constraints = getGridBagConstraints(0, 0, 1, 1, 0, 0,

GridBagConstraints.CENTER, GridBagConstraints.NONE, new Insets(

10, 0, 10, 0), 0, 0);

gridbag.setConstraints(jlabel, constraints);

jpanel.add(jlabel);

// 用户名输入框显示

constraints = getGridBagConstraints(1, 0, 1, 1, 0, 0,

GridBagConstraints.CENTER, GridBagConstraints.NONE, new Insets(

10, 0, 10, 0), 100, 0);

gridbag.setConstraints(jtfield1, constraints);

jpanel.add(jtfield1);

// 密码文本框显示

constraints = getGridBagConstraints(0, 1, 1, 1, 0, 0,

GridBagConstraints.CENTER, GridBagConstraints.NONE, new Insets(

10, 0, 10, 0), 0, 0);

gridbag.setConstraints(jlabel1, constraints);

jpanel.add(jlabel1);

// 密码输入框显示

constraints = getGridBagConstraints(1, 1, 1, 1, 0, 0,

GridBagConstraints.CENTER, GridBagConstraints.NONE, new Insets(

10, 0, 10, 0), 100, 0);

gridbag.setConstraints(jpfield1, constraints);

jpanel.add(jpfield1);

// 帮助按钮显示

constraints = getGridBagConstraints(0, 2, 1, 1, 0, 0,

GridBagConstraints.CENTER, GridBagConstraints.NONE, new Insets(

10, 0, 10, 0), 0, 0);

gridbag.setConstraints(jbutton3, constraints);

jpanel.add(jbutton3);

// 登录按钮显示

constraints = getGridBagConstraints(1, 2, 1, 1, 0, 0,

GridBagConstraints.CENTER, GridBagConstraints.NONE, new Insets(

10, 0, 10, 0), 0, 0);

gridbag.setConstraints(jbutton1, constraints);

jpanel.add(jbutton1);

// 退出按钮显示

constraints = getGridBagConstraints(2, 2, 1, 1, 0, 0,

GridBagConstraints.CENTER, GridBagConstraints.NONE, new Insets(

10, 0, 10, 0), 0, 0);

gridbag.setConstraints(jbutton2, constraints);

jpanel.add(jbutton2);

//获得ActionEvent的监听器

jbutton1.addActionListener(this);

jbutton2.addActionListener(this);

jbutton3.addActionListener(this);

jframe.add(jpanel);

jframe.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

jframe.setResizable(false);

jframe.setVisible(true);

}

public void actionPerformed(ActionEvent e)

{

if(e.getSource()==jbutton1){

if(jtfield1.getText().equals("")||jpfield1.getPassword().equals("")){

JOptionPane.showMessageDialog(null,"输入不能为空！");

}

else{

try{

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

} catch(ClassNotFoundException a){

System.out.println(""+a);

}

try{

con=DriverManager.getConnection("jdbc:odbc:mysource","","");

sql=con.createStatement();

rs=sql.executeQuery("Select \* FROM login\_1"); //查询数据库

boolean check = false;

while(rs.next()){

String name=rs.getString(2); //获得数据库第二列

String password=rs.getString(3); //获得数据库第三列

char[] s=jpfield1.getPassword();

String psw=new String(s);

if(jtfield1.getText().equals(name) && psw.equals(password))

{ //判断语句

check = true;

new MainFrame(jtfield1.getText());

jframe.dispose();

break;

}

}

if(check == false){

JOptionPane.showMessageDialog(null,"登陆失败，请重新输入！");

}

con.close();

} catch(SQLException el){}

}

}

else if(e.getSource()==jbutton2)

{

System.exit(0);

}

if(e.getSource()==jbutton3)

{

JOptionPane.showMessageDialog(null, "您需要注册一个用户名和密码才可以进入系统！");

}

}

private static GridBagConstraints getGridBagConstraints(int gridx,

int gridy, int gridwidth, int gridheight, double weightx,

double weighty, int anchor, int fill, Insets insets, int ipadx,

int ipady) {

return new GridBagConstraints(gridx, gridy, gridwidth, gridheight,

weightx, weighty, anchor, fill, insets, ipadx, ipady);

}

}

**2.系统管理模块**

功能：实现对商品的管理。

代码：

import java.awt.event.\*;

import java.awt.\*;

import javax.swing.\*;

public class MainFrame {

private final static javax.swing.ImageIcon icon = new javax.swing.ImageIcon("src/img/logo1.jpg");

private JFrame frame = new javax.swing.JFrame();

private JMenuBar menuBar = null;

private JToolBar toolBar = null;

private JSplitPane splitPane = null;

private JTabbedPane tabbedPane = new javax.swing.JTabbedPane();

private JButton Retail = null;//商品零售

private JButton stock = null;//库存管理

private JButton help = null;//帮助

private JButton about = null;//关于

private JButton exit\_system = null;//退出系统

private JLabel leftLabel=null;//左部标签

private String sql = null;

public MainFrame(String name){

try

{

UIManager.setLookAndFeel(UIManager.getCrossPlatformLookAndFeelClassName());

}

catch(Exception exe)

{System.err.print(exe.getMessage());}

java.awt.Toolkit tool = frame.getToolkit();

Image image = tool.getImage("res/logo.jpg");

Dimension dimn = tool.getScreenSize();

String title = "超市管理系统--"+name;

frame.setTitle(title);

frame.setIconImage(image);

frame.setFocusable(true);

frame.setLayout(new BorderLayout());

frame.setJMenuBar(createJMenuBar());

frame.add(createJToolBar(),"North");

frame.add(createSplitPane(),"Center");

frame.addWindowListener(new WindowListener(){

public void windowActivated(WindowEvent e){}

//因对窗口调用dispose()而将其关闭时调用

public void windowClosed(WindowEvent e){}

//用户试图从窗口的系统菜单中关闭窗口时调用

public void windowClosing(WindowEvent e){

frame.dispose();

System.exit(0);

}

//当窗口不再是活动窗口时调用

public void windowDeactivated(WindowEvent e){}

//窗口从最小化状态变为正常状态时调用

public void windowDeiconified(WindowEvent e){}

//窗口从正常状态变为最小状态时调用

public void windowIconified(WindowEvent e){}

//窗口首次变为可见时调用

public void windowOpened(WindowEvent e){}

});

Dimension sr=new Dimension(750,640);

frame.setPreferredSize(sr);

frame.setLocation((dimn.width-750)/2, (dimn.height-640)/2);

frame.setResizable(false);

frame.setBackground(java.awt.Color.pink);

frame.pack();

frame.validate();

frame.setVisible(true);

}

private JMenuBar createJMenuBar()

{

menuBar = new JMenuBar()

{

public void paintComponent(Graphics g)

{

super.paintComponent(g);

if(!isOpaque()){return;}

Graphics2D g2d = (Graphics2D) g;

AlphaComposite opaque = AlphaComposite.SrcOver;

int width = getWidth();

int height = getHeight();

GradientPaint gradientPaint = new GradientPaint(0,0,java.awt.Color.green,width/2,height/2,java.awt.Color.yellow,false);

g2d.setComposite(opaque);

g2d.setPaint(gradientPaint);

g2d.fillRect(0, 0, width,height);

g2d.setRenderingHint(RenderingHints.KEY\_ANTIALIASING,RenderingHints.VALUE\_ANTIALIAS\_OFF);

}

};

menuBar.add(createSystem\_Manage\_Menu());

menuBar.add(createRetail\_Manage\_Menu());

menuBar.add(createStock\_Manage\_Menu());

menuBar.add(createHelp\_Menu());

menuBar.setBackground(java.awt.Color.pink);

return menuBar;

}

private JToolBar createJToolBar(){

toolBar = new javax.swing.JToolBar(){

public void paintComponent(Graphics g){

super.paintComponent(g);

if(!isOpaque()){return;}

Graphics2D g2d = (Graphics2D) g;

AlphaComposite opaque = AlphaComposite.SrcOver;

int width = getWidth();

int height = getHeight();

GradientPaint gradientPaint = new GradientPaint(0,0,java.awt.Color.green,width/2,height/2,java.awt.Color.yellow,false);

g2d.setComposite(opaque);

g2d.setPaint(gradientPaint);

g2d.fillRect(0, 0, width,height);

g2d.setRenderingHint(RenderingHints.KEY\_ANTIALIASING,RenderingHints.VALUE\_ANTIALIAS\_OFF);

}

};

Retail = new javax.swing.JButton("商品零售");

Retail.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e){

retailManage("零 售");

}

});

stock = new javax.swing.JButton("库存管理");

stock.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e){

stockManage("库存信息");

}

});

help = new javax.swing.JButton("帮 助");

help.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e){

JOptionPane.showMessageDialog(null,"帮助");

//AddHelp();

}

});

about = new javax.swing.JButton("关 于");

about.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e){

JOptionPane.showMessageDialog(null,"关于");

//new UserAbout();

}

});

exit\_system = new javax.swing.JButton("退出系统");

exit\_system.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e){

frame.dispose();

System.exit(0);

}

});

toolBar.add(Retail);

toolBar.addSeparator();

toolBar.add(stock);

toolBar.addSeparator();

toolBar.add(help);

toolBar.addSeparator();

toolBar.add(about);

toolBar.addSeparator();

toolBar.add(exit\_system);

toolBar.setBackground(java.awt.Color.pink);

return toolBar;

}

private JSplitPane createSplitPane(){

leftLabel=new JLabel();

leftLabel.setBackground( Color.green);

leftLabel.setBounds(0, 0, 400, 600);

leftLabel.setOpaque(true);

splitPane = new JSplitPane(JSplitPane.HORIZONTAL\_SPLIT,true,leftLabel,createHandlePane()){

public void paintComponent(Graphics g){

super.paintComponent(g);

if(!isOpaque()){return;}

Graphics2D g2d = (Graphics2D) g;

AlphaComposite opaque = AlphaComposite.SrcOver;

int width = getWidth();

int height = getHeight();

GradientPaint gradientPaint = new GradientPaint(0,0,java.awt.Color.green,width/2,height/2,java.awt.Color.yellow,false);

g2d.setComposite(opaque);

g2d.setPaint(gradientPaint);

g2d.fillRect(0, 0, width,height);

g2d.setRenderingHint(RenderingHints.KEY\_ANTIALIASING,RenderingHints.VALUE\_ANTIALIAS\_OFF);

}

};

splitPane.setDividerLocation(138);

splitPane.setDividerSize(5);

splitPane.setOneTouchExpandable(true);

splitPane.setBackground(java.awt.Color.pink);

return splitPane;

}

private JTabbedPane createHandlePane(){

javax.swing.ImageIcon image = new javax.swing.ImageIcon("src/img/welcome1.jpg");

javax.swing.JLabel component = new javax.swing.JLabel(image);

tabbedPane.addTab("欢迎",icon ,component);

tabbedPane.setBackgroundAt(0, java.awt.Color.pink);

return tabbedPane;

}

private void retailManage(String s){

int i = tabbedPane.getTabCount();

if(i>0)tabbedPane.removeAll();

tabbedPane.addTab(s, icon,new creatRetailFace());

int b = tabbedPane.getTabCount();

tabbedPane.setSelectedIndex(b-1);

}

private void stockManage(String s){

int i = tabbedPane.getTabCount();

if(i>0)tabbedPane.removeAll();

tabbedPane.addTab(s, icon,new creatstockFace());

int b = tabbedPane.getTabCount();

tabbedPane.setSelectedIndex(b-1);

}

private JMenu createSystem\_Manage\_Menu(){

javax.swing.JMenu systemMenu = new javax.swing.JMenu("系统管理");

javax.swing.JMenuItem login = new javax.swing.JMenuItem("注 册");

login.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e){

JOptionPane.showMessageDialog(null,"注册");

//new AddUser();

}

});

javax.swing.JMenuItem logout = new javax.swing.JMenuItem("删除用户");

logout.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e){

String jp = new JOptionPane().showInputDialog(frame,"请输入要删除的用户名！","输入用户名",JOptionPane.YES\_NO\_CANCEL\_OPTION).trim();

}

});

javax.swing.JMenuItem exit = new javax.swing.JMenuItem("退出系统");

exit.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e){

System.exit(0);

}

});

systemMenu.add(login);

systemMenu.add(logout);

systemMenu.add(exit);

return systemMenu;

}

private JMenu createRetail\_Manage\_Menu(){

javax.swing.JMenu retailMenu = new javax.swing.JMenu("零售管理");

javax.swing.JMenuItem retail = new javax.swing.JMenuItem("零 售");

retail.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e){

retailManage("零 售");

}

});

retailMenu.add(retail);

return retailMenu;

}

private JMenu createStock\_Manage\_Menu(){

javax.swing.JMenu stockMenu = new javax.swing.JMenu("库存管理");

javax.swing.JMenuItem purcharse = new javax.swing.JMenuItem("库存信息");

purcharse.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e){

stockManage("库存信息");

}

});

javax.swing.JMenuItem updata = new javax.swing.JMenuItem("库存更新");

updata.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e){

}

});

stockMenu.add(purcharse);

stockMenu.add(updata);

return stockMenu;

}

private JMenu createHelp\_Menu(){

javax.swing.JMenu helpMenu = new javax.swing.JMenu("帮助");

javax.swing.JMenuItem help = new javax.swing.JMenuItem("帮 助");

help.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e){

JOptionPane.showMessageDialog(null,"帮助");

}

});

javax.swing.JMenuItem about = new javax.swing.JMenuItem("关 于");

about.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e){

JOptionPane.showMessageDialog(null,"关于");

}

});

helpMenu.add(help);

helpMenu.add(about);

return helpMenu;

}

}

## 3.2性能

可扩展，易维护。

## 3.3输入项目

商品名称、价格、生产日期、保质期等。

## 3.4输出项目

商品剩余数目。

## 3.5算法

【模块所选用的算法。】

连接数据库后进行查找。

## 3.6程序逻辑

【详细描述模块实现的算法，可采用：

1. 标准流程图；
2. PDL语言；
3. N－S图；
4. PAD；
5. 判定表等描述算法的图表。】

## 3.7接口

## 3.8存储分配

## 3.9限制条件

## 3.10测试要点

【给出测试模块的主要测试要求。】