 ****

**软 件 学 院**

**二级课题报告**

**课题名称：** 会员管理系统开发

**专业：** 软件工程

**班级：** 卓越161

**学号：** 201619140220

**学生姓名：** 崔烁豪

**指导老师：** 韩玉民

2017年 10月

**中原工学院软件学院**

**二级实践课**题任务书

|  |  |  |  |
| --- | --- | --- | --- |
| **姓名** | 崔烁豪 | **班级** | 卓越161班 |
| **题目** | **会员管理系统开发（PC版）** | | |
| **设**  **计**  **任**  **务** | 随着社会信息化的日益广泛，服务业的会员管理基本上都采用计算机管理，为了提高会员管理效率，开发一通用的PC版会员管理系统，训练数据库管理实践能力。  系统主要功能包括：  1.会员管理  对会员基本信息进行管理，包括会员基本信息的增删改查，会员基本信息主要包括会员号、卡号、姓名、联系方式、会员级别、办卡时间、积分、推荐人（经办人）等。  2.会员级别管理  会员级别信息主要包括级别编号、级别说明、折扣等。  3.储值管理  对会员充值记录进行管理，包括充值时间、金额、余额等，可按卡号、手机号或姓名查询会员。  4.消费管理  对会员消费记录进行管理，包括消费时间、金额、余额等，可按卡号、手机号或姓名查询会员  5.统计汇总  可统计系统中会员数、总储值余额等。  6.日报月报  可按天、月统计汇总储值、消费情况。  7.短信提示  对会员储值、消费、店面优惠活动等进行短信提示。  开发工具：MySQL和 Eclipse | | |
| **时**  **间**  **进**  **度** | 第1周（2017-09-11~2016-09-17）：对需求进行深入分析，并设计出架构与数据库，仔细理解需求，规划功能。  第2周（2017-09-18~2017-09-24）：学习并完成每一步的开发，在过程中发现问题并改进问题，学习使用第三方的API来完成系统，并进行下一步的功能该进，并优化界面。  第3周（2017-09-25~2017-09-30）：调试测试，通过实际用户的操作，提出建议并修改，撰写报告。 | | |
| **原主**  **始要**  **资参**  **料考**  **与文**  **献** | [01]Java swing日历控件[EB/OL].  https://www.cnblogs.com/lzy1991/p/5714935.html.2017-9-13  [02]JAVA中获取系统当前时间[EB/OL].  https://www.cnblogs.com/Matrix54/archive/2012/05/01/2478158.html.2017-9-14  [03]getLayeredPane与getContentPane的区别[EB/OL].  http://blog.csdn.net/dyf4281/article/details/17091677.2017-9-16  [04]刘瑞新.SQL Server数据库技术及应用教程[M].电子工业出版社,2012.05 | | |

# 摘 要

在当今经济的形式下，人民币不断贬值，很多人都选择了投资。但亦是由于货币政策的印象，很多店铺在开店的时候资金短缺，这个时候很多店铺选择充值消费，以达到回收资金的目的。亦可通过打折充值来吸引客户。会员管理系统的模式深受国人欢迎，无论是消费者还是中小店铺的老板，都热衷于利用会员管理系统来谋求更大的利益。所以设计开发一个会员管理系统很有必要的。

该系统主要包括查询会员的个人信息，会员级别，充值记录和消费记录，会员数、总储值余额。会员信息以及储值记录消费记录的增删改查等功能。本文共5章，第1章主要介绍了会员管理系统开发的背景，研究意义及系统功能，第2章是对会员管理系统的分析，第3章是分析本系统数据库的设计，第4章是对系统的几大功能的具体实现进行阐述，第5章介绍在软工实践课题一完成过程中的一些心得体会，以及对此次实践成果的评价，进行总结。

关键字： 会员；管理系统；实践课题

至少3个关键字，要能反映题目

# 目录

[第1章 概述 1](#_Toc24012)

[1.1 课题背景 1](#_Toc1884)

[1.2 会员管理系统的现状 1](#_Toc5152)

[1.3 会员管理系统的功能 1](#_Toc17902)

[1.4 会员管理系统的作用 1](#_Toc2792)

[第2章 系统分析 3](#_Toc30001)

[2.1 问题描述 3](#_Toc8562)

[2.2 技术分析 3](#_Toc7542)

[2.3 需求分析 3](#_Toc14849)

[第3章 数据库分析与设计 5](#_Toc14792)

[3.1 实体分析 5](#_Toc25773)

[3.2关系设计 6](#_Toc9733)

[3.3数据表设计 6](#_Toc6617)

[第4章系统实现 8](#_Toc23800)

[4.1 用户登录 8](#_Toc9524)

[4.2 会员信息查询 13](#_Toc3938)

[4.3 用户信息管理 23](#_Toc24869)

[4.4 系统的发布 29](#_Toc28714)

[第5章 结束语 30](#_Toc22177)

[参考文献 32](#_Toc18260)

[指导老师评语 33](#_Toc11492)

# 第1章 概述

本章主要阐述会员管理系统的开发背景、应用现状及开发意义。

## 1.1 课题背景

在当今经济的形式下，人民币不断贬值，很多人都选择了投资。但亦是由于货币政策的印象，很多店铺在开店的时候资金短缺，这个时候很多店铺选择充值消费，以达到回收资金的目的。亦可通过打折充值来吸引客户。

## 1.2 会员管理系统的现状

会员管理系统的模式深受国人欢迎，无论是消费者还是中小店铺的老板，都热衷于利用会员管理系统来谋求更大的利益，由此，也催生了一批会员管理系统的产业链。

## 1.3 会员管理系统的功能

会员管理系统主要是研究管理方面的应用。第一部分，你得先登录之后才能进行一下的操作，然后，你就可以查看全部用户的基本信息和用户的消费记录和储值记录，还可以对用户进行管理，比如增加新的用户，或者删除用户，再或者更改已有的用户的信息。

## 1.4 会员管理系统的作用

会员管理系统的开发意义就在于方便商户或管理者，给商户或管理者带来方便。弥补之前的不足，给商户或管理者提供不同的方式来管理自己的用户，尝试着用一个简单的管理系统来完善不足。这样一个程序安装到自己打的电脑上就可以长时间的访问。

# 第2章 系统分析

本章主要是对会员管理系统项目进行合理性的分析，通过问题描述、技术分析、需求分析这三个方面连描述这个项目。

## 2.1 问题描述

现如今，会员管理系统的模式深受国人欢迎。经过分析，决定设计一个会员管理系统。为什么要设计开发这样的一个程序？因为无论是消费者还是中小店铺的老板都热衷于利用会员管理系统来谋求更大的利益。这个系统比较小规模，容易实现。所以综合考虑接设计开发了这个系统。

## 2.2 技术分析

系统主要运用Eclipse，它同时也能开发跨平台的应用程序，如开发使用微软操作系统的手机的程序等。总体来说是一个非常庞大的软件，甚至包含代码测试功能。同时还需要MySQL创建数据库 ,和java连接数据库的操作。

## 2.3 需求分析

会员管理系统的模式深受国人欢迎，无论是消费者还是中小店铺的老板，都热衷于利用会员管理系统来谋求更大的利益。所以设计开发一个会员管理系统很有必要的。

该系统主要包括以下功能：

（1）用户登陆，密码修改。

（2）查询会员的个人信息，会员级别，充值记录和消费记录。

（3）会员数、总储值余额

（4）会员信息以及储值记录消费记录的增删改查系统功能结构图如图2-1所示。

会员管理系统

数据管理

据管理据管理

增删改查用户

增删改差储值、消费记录

数据查询

查询用户信息

查询用户储

值消费记录

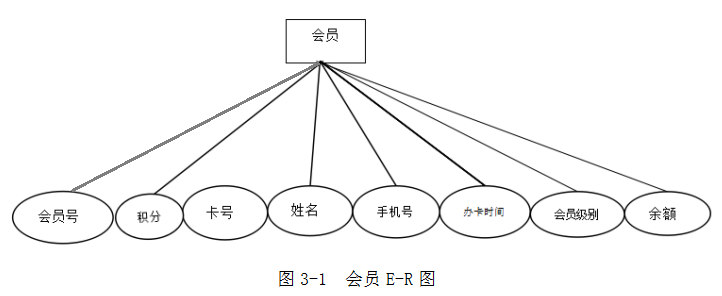
图2-1 会员管理系统功能结构图

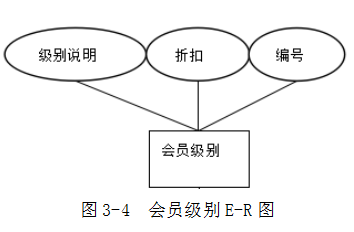
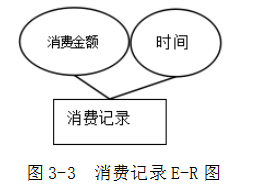
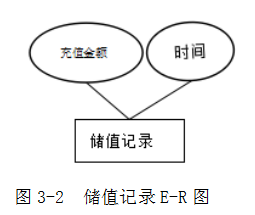
# 第3章 数据库分析与设计

数据库拥有很强的功能，可以建立一个数据库把我们所需要的数据给存储起来，并可以进行增删改查的操作，还可以对数据库进行保存和他的安全性等等多的工能。我们采用MySQL数据库来保存会员的信息及储值记录、消费记录。

## 3.1 实体分析

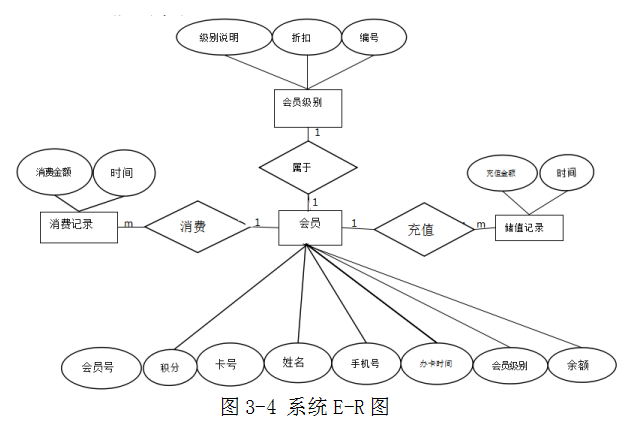
本系统涉及的实体包括会员、储值记录、消费记录、会员级别等。经过分析，会员实体E-R图如图3-1，储值记录实体E-R图如图3-2，消费记录实体E-R图如图3-3，会员级别实体E-R图如图3-4。





## 3.2关系设计

这里有四个实体，分别是：会员、储值记录、消费记录、会员级别，它们之间的关系分别是：会员级别属于会员，一对一;会员充值后有充值记录，一对多;会员储值后有储值记录，一对多。



## 3.3数据表设计

会员管理系统的数据库设计总共有四个表：会员管理表、会员级别表、储值记录表、消费记录表，表的具体结构见表3-1、表3-2、表3-3和表3-4。

表3-1 会员管理表(VipAdminis)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **列名** | **描述** | **数据类型及长度** | **是否为空** | **主键** |
| Vip\_No | 卡号 | varchar(10) | 否 | 是 |
| VipAdminis\_No | 会员号 | varchar(10) | 是 |  |
| VipAdminis\_Name | 姓名 | varchar(10) | 是 |  |
| VipAdminis\_Tel | 手机号 | varchar(15) | 是 |  |
| VipAdminis\_Time | 办卡时间 | date | 是 |  |
| VipAdminis\_Money | 账户余额 | float | 是 |  |
| VipAdminis\_Level | 会员级别 | int | 是 |  |
| VipAdminis\_Score | 会员积分 | int | 是 |  |

表3-2 会员级别表(VipLevel)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **列名** | **描述** | **数据类型及长度** | **是否为空** | **主键** |
| VipLevel\_No | 级别编号 | int | 是 | 是 |
| VipLevel\_Discount | 折扣 | float | 是 |  |
| VipLevel\_Instructions | 级别说明 | varchar(10) | 是 |  |

表3-3 储值记录表(Vipin)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **列名** | **描述** | **数据类型及长度** | **是否为空** | **主键** |
| Vipin\_Money | 充值金额 | float | 是 |  |
| Vipin\_Time | 时间 | date | 是 |  |

表3-4 消费纪录表(Vipout)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **列名** | **描述** | **数据类型及长度** | **是否为空** | **主键** |
| Vipout\_Money | 消费金额 | float | 是 |  |
| Vipout\_Time | 时间 | date | 是 |  |

# 第4章系统实现

本章主要描述个人所完成的系统模块的实现过程，包括会员登录、消费表、储值表等模块，按模块阐述了系统功能的实现技术、流程和主要代码。

会员管理系统的功能，在设计上分为四个层次。具体如图4-0-1所示：

用户登录模块可以分为：(1)用户登录界面. (2)登录后的管理员界面.两部分。

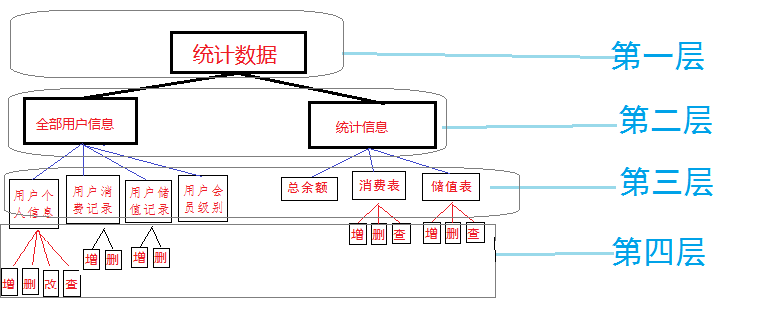
会员管理系统的登录界面流程图如图4-1所示。

图4-0-1设计思路

## 4.1 用户登录

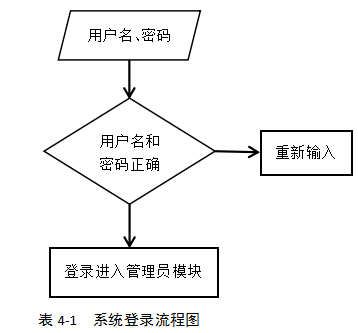
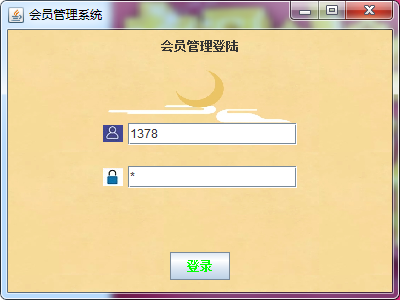


图4-2为登录界面。这个界面主要是用了button 按钮用于指令和Label按钮用于提示信息和textbox 文本框用于显示文本。



**图4-2 登陆界面**

登录界面的主要代码如下：

**public** **class** Login {//登录窗口

**public** Login() {

Login.*jiemian*();**public** **static** **void** jiemian(){

JFrame jframe=**new** JFrame("会员管理系统");

JPanel jp1=**new** JPanel();

JPanel jp2=**new** JPanel();

JPanel jp3=**new** JPanel();

JPanel jp4=**new** JPanel();

JPanel jp5=**new** JPanel();

JPanel jp6=**new** JPanel();

jp1.setOpaque(**false**);

jp2.setOpaque(**false**);

jp3.setOpaque(**false**);

jp4.setOpaque(**false**);

jp5.setOpaque(**false**);

jp6.setOpaque(**false**);

JLabel lab1=**new** JLabel("会员管理登陆",JLabel.***CENTER***);

JLabel lab2=**new** JLabel("",JLabel.***CENTER***);

JLabel lab3=**new** JLabel("",JLabel.***CENTER***);

JLabel lab4=**new** JLabel("",JLabel.***CENTER***);

JLabel lab5=**new** JLabel("身份: ",JLabel.***CENTER***);

JButton b=**new** JButton("登陆");

// String str[]={"管理员"};

// JComboBox<String> box=new JComboBox<String>(str);

JTextField t1=**new** JTextField("",15);

JPasswordField t2=**new** JPasswordField("",15);

t2.setEchoChar('\*');

ImageIcon icon1=**new** ImageIcon("头.jpg");

ImageIcon icon2=**new** ImageIcon("密.jpg");

lab3.setIcon(icon1);

lab4.setIcon(icon2);

jp1.add(lab1);

jp2.add(lab2);

jp3.add(lab3);

jp3.add(t1);

jp4.add(lab4);

jp4.add(t2);

jp6.add(b);

FreedomPane freedompane=**new** FreedomPane();

freedompane.setLayout(**new** GridLayout(6,1));

freedompane.add(jp1);

freedompane.add(jp2);

freedompane.add(jp3);

freedompane.add(jp4);

freedompane.add(jp5);

freedompane.add(jp6);

JPanel mainpane=**new** JPanel();

mainpane.add(freedompane);

mainpane.add(freedompane);

mainpane.setLayout(**new** CardLayout());

jframe.setContentPane(mainpane);

//使窗口居中显示

Toolkit tk=Toolkit.*getDefaultToolkit*();

Dimension sc=tk.getScreenSize();//获取屏幕尺寸

jframe.setBounds(400, 300,400,300);

b.setForeground(Color.***GREEN***);

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

jframe.setVisible(**true**);

b.addActionListener(**new** ActionListener() {//

**public** **void** actionPerformed(ActionEvent e) {

String user=t1.getText();

String pass=**new** String(t2.getPassword());

//String identity=(String)box.getSelectedItem();

Login\_Check\_Function date=**new** Login\_Check\_Function(user,pass);

**boolean** flag = date.prepared();

**if**(user.equals("")){

lab2.setForeground(Color.***RED***);

lab2.setText("请输入帐号！！！");

}**else** **if**(pass.equals("")){

lab2.setForeground(Color.***RED***);

lab2.setText("请输入密码！！！");

}**else** **if** (flag) { //判断是否为合法用户

**new** Manager\_Login();

jframe.dispose(); //隐藏当前窗体

lab2.setForeground(Color.***GREEN***);

lab2.setText("登录成功！！！");

} **else** **if** (!flag){

lab2.setForeground(Color.***RED***);

lab2.setText("账号或密码错误！！！");

}

}

});

}

}

图4-3 为管理员界面，这个界面主要是用了菜单的功能，中间加了一个图，每个菜单项对应相应的功能。



图4-3管理员模块界面

管理员模块界面主要代码如下：

**public** **class** Manager\_Login {

**public** Manager\_Login(){

JFrame jframe=**new** JFrame("管理员模块");

ImageIcon icon=**new** ImageIcon("timg.jpg");

JLabel la=**new** JLabel(icon);

jframe.add(la);

jframe.setResizable(**false**);

JMenuBar mb=**new** JMenuBar();

jframe.setJMenuBar(mb);

JMenu m1=**new** JMenu("用户");

JMenu m2=**new** JMenu("统计数据");

JMenuItem m21=**new** JMenuItem("总余额");

JMenuItem m22=**new** JMenuItem("消费表");

JMenuItem m23=**new** JMenuItem("储值表");

JMenu m3=**new** JMenu("短信");

JMenuItem m31=**new** JMenuItem("短信");

JMenu m4=**new** JMenu("重新登录");

JMenuItem m41=**new** JMenuItem("重新登录");

m2.add(m21);

m2.add(m22);

m2.add(m23);

m3.add(m31);

m4.add(m41);

JMenuItem mi6=**new** JMenuItem("全部用户");

m1.add(mi6);

mb.add(m1);

mb.add(m2);

mb.add(m3);

mb.add(m4);

jframe.setLocation(400, 100);

jframe.setSize(600,600);

mi6.addActionListener(**new** ActionListener() {//"全部用户"

**public** **void** actionPerformed(ActionEvent e) {

**new** Manager\_User\_AllUser();

jframe.dispose();

}

});

m21.addActionListener(**new** ActionListener() {//"总余额"

**public** **void** actionPerformed(ActionEvent e) {

Manager\_Tongjishuju\_AllMoney.*allmoney*();

}

});

m22.addActionListener(**new** ActionListener() {//"消费表"

**public** **void** actionPerformed(ActionEvent e) {

**new** Manager\_Tongjishuju\_OutMoney();

jframe.dispose();

}

});

m23.addActionListener(**new** ActionListener() {//"储值表"

**public** **void** actionPerformed(ActionEvent e) {

**new** Manager\_Tongjishuju\_InMoney();

jframe.dispose();

}

});

m31.addActionListener(**new** ActionListener() {//"短信"

**public** **void** actionPerformed(ActionEvent e) {

**new** Manager\_Duanxin();

}

});

m41.addActionListener(**new** ActionListener() {//"重新登录"

**public** **void** actionPerformed(ActionEvent e) {

**new** Login();

jframe.dispose();

}

});

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

jframe.setVisible(**true**);

}

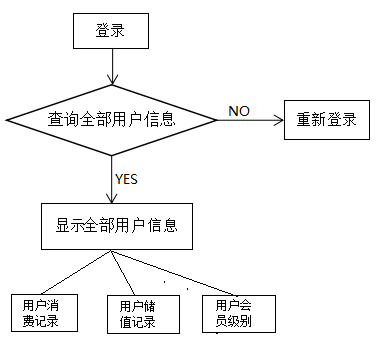
}

## 4.2 会员信息查询

会员信息查询模块分为：(1)全部用户信息查询.(2)统计信息查询.两部分。

全部用户信息查询主要是对会员信息的查询，用户可以通过窗口显示会员个人信息，会员储值记录、会员消费记录和会员级别，也可以查询所有会员在某个时间段的储值记录和消费记录，以及全部用户的人数和余额。

全部用户信息查询的流程图如图4-4所示。



**图4-4 会员信息查询流程图**

全部会员信息的界面如图4-5：



**图4-5 全部会员信息**

全部会员信息的关键代码如下：

public class Manager\_User\_AllUser {//全部用户

public Manager\_User\_AllUser() {

JFrame f = new JFrame("会员\_用户\_全部用户");

JButton b1=new JButton("返");

JButton b2=new JButton("增");

JButton b3=new JButton("删");

JButton b4=new JButton("改");

JButton b5=new JButton("查");

JButton b6=new JButton("消费记录");

b6.setOpaque(false);

JButton b7=new JButton("储值记录");

JButton b8=new JButton("会员级别");

JPanel p2=new JPanel();

p2.setOpaque(false);

p2.add(b1);

p2.add(b2);

p2.add(b3);

p2.add(b4);

p2.add(b5);

JPanel p3=new JPanel();

p3.setOpaque(false);

p3.add(b6);

p3.add(b7);

p3.add(b8);

Vector<Object> vv=Manager\_User\_AllUser.query();

Vector<Object> Names = new Vector<Object>();

Names.add( "卡号");

Names.add("会员号");

Names.add("姓名");

Names.add("电话");

Names.add("办卡时间");

Names.add("余额");

Names.add("会员级别");

Names.add("积分");

Names.add("推荐人");

DefaultTableModel model=new DefaultTableModel(vv, Names);

JTable table = new JTable(model);

table.setOpaque(false);

table.setPreferredScrollableViewportSize(new Dimension(55, 30));// 将表格加入到滚动条组件中

JScrollPane scrollPane = new JScrollPane(table);

scrollPane.setOpaque(false);

JPanel jpanel=new JPanel();

jpanel.setOpaque(false);

jpanel.add(scrollPane);

f.getContentPane().add(scrollPane, BorderLayout.CENTER);// 再将滚动条组件添加到中间容器中

f.add(p3,BorderLayout.NORTH);

f.add(p2,BorderLayout.SOUTH);

f.pack();

b1.addActionListener(new ActionListener() {//"返"

public void actionPerformed(ActionEvent e) {

new Manager\_Login();

f.dispose();

}

});

b2.addActionListener(new ActionListener() {//"增"

public void actionPerformed(ActionEvent e) {

new Manager\_AllUser\_Add\_Function();

f.dispose();

}

});

b3.addActionListener(new ActionListener() {//"删"

public void actionPerformed(ActionEvent e) {

int rows = table.getSelectedRow();

if(rows<0){

JOptionPane.showMessageDialog(null, "请选中一行", "☆★提示信息☆★",

JOptionPane.INFORMATION\_MESSAGE);

}else{

String row=(String)table.getValueAt(rows, 0);

model.removeRow(rows);

Manager\_User\_AllUser.deleteDao(row);

JOptionPane.showMessageDialog(null, "删除成功", "☆★提示信息☆★",

JOptionPane.INFORMATION\_MESSAGE);

}

}

});

b4.addActionListener(new ActionListener() {//"改"

public void actionPerformed(ActionEvent e) {

int rows = table.getSelectedRow();

if(rows<0){

JOptionPane.showMessageDialog(null, "请选中一行", "☆★提示信息☆★",

JOptionPane.INFORMATION\_MESSAGE);

}else{

String row=(String)table.getValueAt(rows, 0);

new Manager\_AllUser\_Change\_Function(row);

f.dispose();

}

}

});

b5.addActionListener(new ActionListener() {//"查"

public void actionPerformed(ActionEvent e) {

new Manager\_AllUser\_Query\_Function();

f.dispose();

}

});

b6.addActionListener(new ActionListener() {//"用户消费记录"

public void actionPerformed(ActionEvent e) {

int rows = table.getSelectedRow();

if(rows<0){

JOptionPane.showMessageDialog(null, "请选中一行", "☆★提示信息☆★",

JOptionPane.INFORMATION\_MESSAGE);

}else{

String id=(String)table.getValueAt(rows, 0);

String name=(String)table.getValueAt(rows, 2);

new Manager\_Tongjishuju\_OutMoney(id,name);

f.dispose();

}

}

});

b7.addActionListener(new ActionListener() {//"用户储值记录"

public void actionPerformed(ActionEvent e) {

int rows = table.getSelectedRow();

if(rows<0){

JOptionPane.showMessageDialog(null, "请选择要查询储值记录的会员", "☆★提示信息☆★",

JOptionPane.INFORMATION\_MESSAGE);

}else{

String id=(String)table.getValueAt(rows, 0);

String name=(String)table.getValueAt(rows, 2);

new Manager\_Tongjishuju\_InMoney(id,name);

f.dispose();

}

}

});

b8.addActionListener(new ActionListener() {//"用户会员级别"

public void actionPerformed(ActionEvent e) {

int rows = table.getSelectedRow();

if(rows<0){

JOptionPane.showMessageDialog(null, "请选中一行", "☆★提示信息☆★",

JOptionPane.INFORMATION\_MESSAGE);

}else{

String level=""+table.getValueAt(rows, 6);

String id=(String)table.getValueAt(rows, 0);

new Manager\_User\_AllUser\_Level\_Function(level, id);

f.dispose();

}

}

});

Container contain = f.getContentPane();

((JPanel) contain).setOpaque(false);

// 将内容面板设为透明。将LayeredPane面板中的背景显示出来。

Tool\_Function.bjpic(f);

f.setSize(600, 600);

f.setLocation(400, 100);

f.setVisible(true);

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

}

public static Vector<Object> query(){//全部会员户表格

Vector<Object> vv=new Vector<Object>();

Vector<Object> v = new Vector<Object>();

Connection conn=null;

PreparedStatement ps=null;

ResultSet rs=null;

try{

conn=DBConnection.getConnection();

String sql="select Vip\_No,VipAdminis\_No,VipAdminis\_Name,VipAdminis\_Tel,VipAdminis\_Time,VipAdminis\_Money,VipAdminis\_Level,VipAdminis\_Score,Vip\_peo from VipAdminis";

ps=conn.prepareStatement(sql);

rs=ps.executeQuery();

while(rs.next()){

v.clear();

v.add(rs.getObject(1));

v.add(rs.getObject(2));

v.add(rs.getObject(3));

v.add(rs.getObject(4));

v.add(rs.getObject(5));

v.add(rs.getObject(6));

v.add(rs.getObject(7));

v.add(rs.getObject(8));

v.add(rs.getObject(9));

vv.add(v.clone());

}

}catch (SQLException e){

System.out.println("查询数据时发生异常！！！");

e.printStackTrace();

}finally{

DBConnection.close(conn);

}return vv;

}

public static int deleteDao(String id) {//删除会员行

Connection conn = null; //连接数据库对象

PreparedStatement pstmt = null; //预编译对象

int rows = 0; //更新语句行数变量

try {

conn = DBConnection.getConnection(); //获取数据库连接

String sql = "DELETE FROM vipAdminis WHERE Vip\_No=?";

pstmt = conn.prepareStatement(sql); //使用预编译语句进行插入操作

pstmt.setString(1, id); //将名称参数设置到SQL语句中

rows = pstmt.executeUpdate(); //执行更新操作

} catch (SQLException e) { //判断是否发生异常

JOptionPane.showMessageDialog(null,"删除数据时发生异常","Message",

JOptionPane.INFORMATION\_MESSAGE); //异常提示

e.printStackTrace();

} finally {

DBConnection.close(conn); //关闭连接数据库对象资源

}

return rows;

}

}

用户储值记录查询

会员储值记录信息的界面如图4-5：

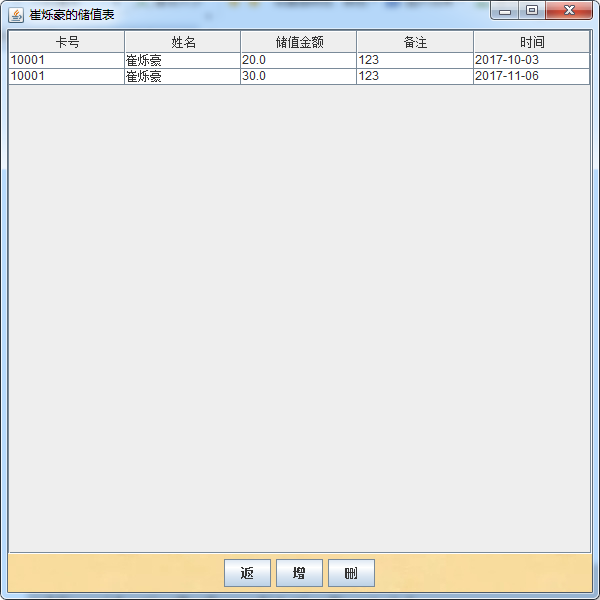


图4-5 会员储值记录

用户储值记录的关键代码如下：

b7.addActionListener(new ActionListener() {//"用户储值记录"

public void actionPerformed(ActionEvent e) {

int rows = table.getSelectedRow();

if(rows<0){

JOptionPane.showMessageDialog(null, "请选中一行", "☆★提示信息☆★",

JOptionPane.INFORMATION\_MESSAGE);

}else{

String id=(String)table.getValueAt(rows, 0);

String name=(String)table.getValueAt(rows, 2);

new Manager\_Tongjishuju\_InMoney(id,name);

f.dispose();

}

}

});

public Manager\_Tongjishuju\_InMoney(String id,String name) {//用户储值表

JFrame f = new JFrame(name+"的储值表");//调用于Vip\_User\_AllUser

JButton b1=new JButton("返");

JButton b2=new JButton("增");

JButton b3=new JButton("删");

JPanel p1=new JPanel();

p1.add(b1);

p1.add(b2);

p1.add(b3);

Vector<Object> vv=Manager\_Tongjishuju\_InMoney.idinquery(id);

Vector Names = new Vector();

Names.add("卡号");

Names.add("姓名");

Names.add("储值金额");

Names.add("时间");

DefaultTableModel model=new DefaultTableModel(vv, Names);

JTable table = new JTable(model);

table.setPreferredScrollableViewportSize(new Dimension(550, 30));// 将表格加入到滚动条组件中

JScrollPane scrollPane = new JScrollPane(table);

f.getContentPane().add(scrollPane, BorderLayout.CENTER);// 再将滚动条组件添加到中间容器中

f.add(p1,BorderLayout.SOUTH);

f.pack();

b1.addActionListener(new ActionListener() {//返

public void actionPerformed(ActionEvent e) {

new Manager\_User\_AllUser();

f.dispose();

}

});

b2.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {//增 储值记录

new Manager\_In\_Add\_Function(id,name);

f.dispose();

}

});

b3.addActionListener(new ActionListener() {//删 储值记录

public void actionPerformed(ActionEvent e) {

int rows = table.getSelectedRow();

if(rows<0){

JOptionPane.showMessageDialog(null, "请选中一行", "☆★提示信息☆★",

JOptionPane.INFORMATION\_MESSAGE);

}else{

String row=(String)table.getValueAt(rows, 0);

model.removeRow(rows);

Manager\_Tongjishuju\_InMoney.delete(row);

JOptionPane.showMessageDialog(null, "删除成功", "☆★提示信息☆★",

JOptionPane.INFORMATION\_MESSAGE);

}

}

});

f.setSize(600, 600);

f.setLocation(400, 100);

f.setVisible(true);

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

}

用户消费记录查询

用户消费记录信息的界面如图4-6：



图4-6 会员消费记录

用户储值记录的关键代码如下：

b6.addActionListener(new ActionListener() {//"用户消费记录"

public void actionPerformed(ActionEvent e) {

int rows = table.getSelectedRow();

if(rows<0){

JOptionPane.showMessageDialog(null, "请选中一行", "☆★提示信息☆★",

JOptionPane.INFORMATION\_MESSAGE);

f.dispose();

}else{

String id=(String)table.getValueAt(rows, 0);

String name=(String)table.getValueAt(rows, 2);

new Manager\_Tongjishuju\_OutMoney(id,name);

f.dispose();

}

}

});

public Manager\_Tongjishuju\_OutMoney(String id,String name) {//用户消费表

JFrame f = new JFrame(name+"的消费表");//调用于Vip\_User\_AllUser

JButton b1=new JButton("返");

JButton b2=new JButton("增");

JButton b3=new JButton("删");

JPanel p1=new JPanel();

p1.add(b1);

p1.add(b2);

p1.add(b3);

Vector<Object> vv=Manager\_Tongjishuju\_OutMoney.idoutquery(id);

Vector Names = new Vector();

Names.add("卡号");

Names.add("姓名");

Names.add("消费金额");

Names.add("时间");

DefaultTableModel model=new DefaultTableModel(vv, Names);

JTable table = new JTable(model);

table.setPreferredScrollableViewportSize(new Dimension(550, 30));// 将表格加入到滚动条组件中

JScrollPane scrollPane = new JScrollPane(table);

f.getContentPane().add(scrollPane, BorderLayout.CENTER);// 再将滚动条组件添加到中间容器中

f.add(p1,BorderLayout.SOUTH);

f.pack();

b1.addActionListener(new ActionListener() {//返 消费记录

public void actionPerformed(ActionEvent e) {

new Manager\_User\_AllUser();

f.dispose();

}

});

b2.addActionListener(new ActionListener() {//增 消费记录

public void actionPerformed(ActionEvent e) {

new Manager\_Out\_Add\_Function(id,name);

f.dispose();

}

});

b3.addActionListener(new ActionListener() {//删 消费记录

public void actionPerformed(ActionEvent e) {

int rows = table.getSelectedRow();

if(rows<0){

JOptionPane.showMessageDialog(null, "请选中一行", "☆★提示信息☆★",

JOptionPane.INFORMATION\_MESSAGE);

}else{

String row=(String)table.getValueAt(rows, 0);

model.removeRow(rows);

Manager\_Tongjishuju\_OutMoney.delete(row);

JOptionPane.showMessageDialog(null, "删除成功", "☆★提示信息☆★",

JOptionPane.INFORMATION\_MESSAGE);

}

}

});

f.setSize(600, 600);

f.setLocation(400, 100);

f.setVisible(true);

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

}

**用户会员级别查询：**

会员级别信息的界面如图4-7：

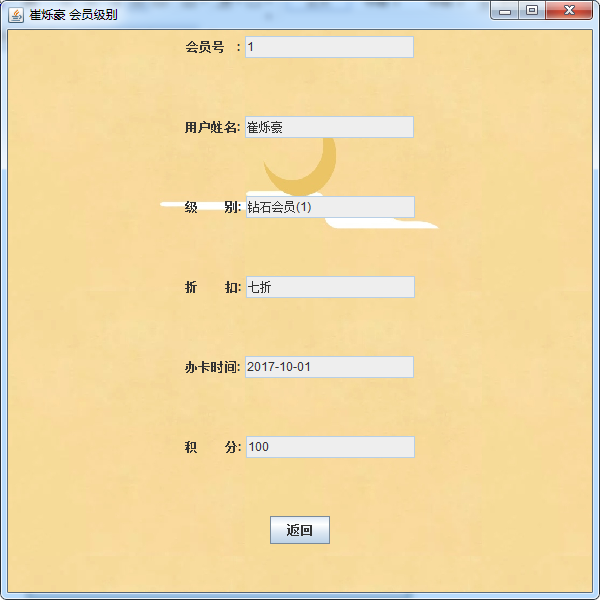


图4-7 会员级别信息

用户会员储值记录的关键代码如下：

public Manager\_User\_AllUser\_Level\_Function(String level,String vip\_no){//根据级别号，会员号 查询 会员级别信息

JFrame jframe=new JFrame();

JPanel p1=new JPanel();

JPanel p6=new JPanel();

JPanel p2=new JPanel();

JPanel p3=new JPanel();

JPanel p4=new JPanel();

JPanel p5=new JPanel();

JPanel p=new JPanel();

p1.setOpaque(false);

p2.setOpaque(false);

p3.setOpaque(false);

p4.setOpaque(false);

p5.setOpaque(false);

p6.setOpaque(false);

p.setOpaque(false);

JLabel lab1=new JLabel("会员号 :",JLabel.CENTER);

JLabel lab6=new JLabel("用户姓名:",JLabel.CENTER);

JLabel lab2=new JLabel("级 别:",JLabel.CENTER);

JLabel lab3=new JLabel("折 扣:",JLabel.CENTER);

JLabel lab4=new JLabel("办卡时间:",JLabel.CENTER);

JLabel lab5=new JLabel("积 分:",JLabel.CENTER);

JButton b3=new JButton("返回");

JTextField t1=new JTextField(15);

JTextField t6=new JTextField(15);

JTextField t2=new JTextField(15);

JTextField t3=new JTextField(15);

JTextField t4=new JTextField(15);

JTextField t5=new JTextField(15);

Object[] os=Manager\_User\_AllUser\_Level\_Function.select\_Level\_ID(level);

String zhe=null;

if("0.7".equals(os[1])){

zhe="七折";

}else if("0.9".equals(os[1])){

zhe="九折";

}else{

zhe="不打折";

}

t3.setText(zhe);

t2.setText((String)os[2]+"("+level+")");

Object[] os1=Manager\_User\_AllUser\_Level\_Function.select\_User\_ID(vip\_no);

t1.setText((String)os1[3]);

t6.setText((String)os1[1]);

t4.setText((String)os1[5]);

t5.setText((String)os1[8]);

t1.setEditable(false);

t6.setEditable(false);

t2.setEditable(false);

t3.setEditable(false);

t4.setEditable(false);

t5.setEditable(false);

p1.add(lab1);

p1.add(t1);

p6.add(lab6);

p6.add(t6);

p2.add(lab2);

p2.add(t2);

p3.add(lab3);

p3.add(t3);

p4.add(lab4);

p4.add(t4);

p5.add(lab5);

p5.add(t5);

p.add(b3);

FreedomPane freedompane=new FreedomPane();

freedompane.setLayout(new GridLayout(7,1));

freedompane.add(p1);

freedompane.add(p6);

freedompane.add(p2);

freedompane.add(p3);

freedompane.add(p4);

freedompane.add(p5);

freedompane.add(p);

JPanel mainpane=new JPanel();

mainpane.add(freedompane);

mainpane.setLayout(new CardLayout());

jframe.setContentPane(mainpane);

jframe.setTitle(os1[1]+" 会员级别");

jframe.setBounds(400,100,600,600);

b3.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

new Manager\_User\_AllUser();

jframe.dispose();

}

});

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

jframe.setVisible(true);

}

//根据级别号 查询 会员级别

public static Object[] select\_Level\_ID(String VipLevel\_No){//传入级别序列 //查询指定用户的所有信息

Object[] os = new Object[3]; //创建对象数组

Connection conn = null; //连接数据库对象

PreparedStatement pstmt = null; //预编译语句对象

ResultSet rs = null; //结果集对象

try {

conn = DBConnection.getConnection(); //获取数据库连接

String sql = "SELECT VipLevel\_No,VipLevel\_Discount,VipLevel\_Instructions FROM VipLevel where VipLevel\_No=?"; //创建sql语句

pstmt = conn.prepareStatement(sql); //使用预编译语句进行查询操作

pstmt.setString(1,VipLevel\_No); //将name参数设置到SQL语句中

rs = pstmt.executeQuery(); //获取结果集

if (rs.next()){ //判断结果集中是否有数据

os[0] = rs.getString(1); //级别编号

os[1] = rs.getString(2); //级别

os[2] = rs.getString(3); //说明

}

} catch (SQLException e){ //判断是否发生异常

JOptionPane.showMessageDialog(null, "查询level指定用户信息时发生异常", "☆★提示信息☆★",

JOptionPane.INFORMATION\_MESSAGE); //提示查询指定用户信息时发生异常

} finally {

DBConnection.close(conn); //关闭连接数据库连接

}

return os;

}

## 4.3 用户信息管理

用户信息管理模块主要实现对用户信息进行增、删、改、查等功能，并且可以对数据进行实时刷新。

会员管理系统的用户信息管理流程图如图4-8所示。

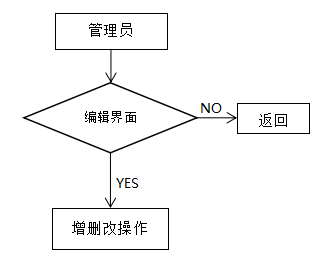


图4-8 会员编辑流程图

增加会员界面如图4-9所示：

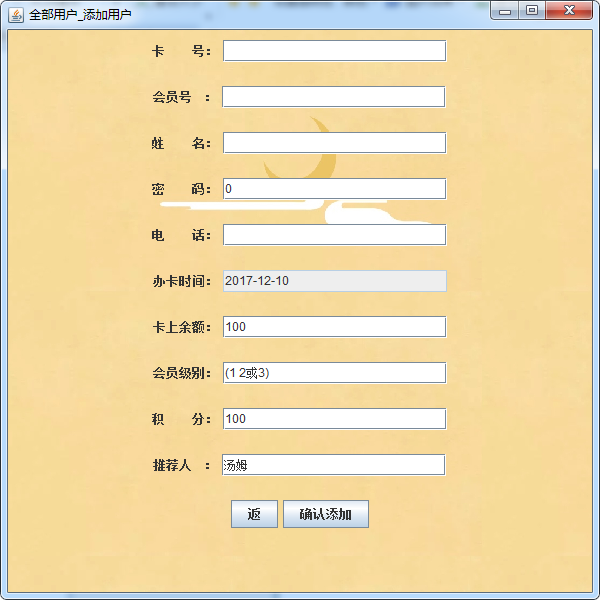


图4-9 增加会员

增加会员主要代码如下：

b2.addActionListener(**new** ActionListener() {//"增"

**public** **void** actionPerformed(ActionEvent e) {

**new** Manager\_AllUser\_Add\_Function();

f.dispose();

}

});

**public** **class** Manager\_AllUser\_Add\_Function {

**public** Manager\_AllUser\_Add\_Function(){

JFrame jframe=**new** JFrame("全部用户\_添加用户");

JPanel p1=**new** JPanel();

JPanel p2=**new** JPanel();

JPanel p3=**new** JPanel();

JPanel p4=**new** JPanel();

JPanel p5=**new** JPanel();

JPanel p6=**new** JPanel();

JPanel p7=**new** JPanel();

JPanel p8=**new** JPanel();

JPanel p9=**new** JPanel();

JPanel p10=**new** JPanel();

JPanel p11=**new** JPanel();

p1.setOpaque(**false**);

p2.setOpaque(**false**);

p3.setOpaque(**false**);

p4.setOpaque(**false**);

p5.setOpaque(**false**);

p6.setOpaque(**false**);

p7.setOpaque(**false**);

p8.setOpaque(**false**);

p9.setOpaque(**false**);

p10.setOpaque(**false**);

p11.setOpaque(**false**);

JLabel u11 = **new** JLabel("卡 号：");

JLabel u12 = **new** JLabel("会员号 ：");

JLabel u13 = **new** JLabel("姓 名：");

JLabel u14 = **new** JLabel("密 码：");

JLabel u15 = **new** JLabel("电 话：");

JLabel u16 = **new** JLabel("办卡时间：");

JLabel u17 = **new** JLabel("卡上余额：");

JLabel u18 = **new** JLabel("会员级别：");

JLabel u19 = **new** JLabel("积 分：");

JLabel u20 = **new** JLabel("推荐人 ：");

JTextField t11 = **new** JTextField("", 20);

JTextField t12 = **new** JTextField("", 20);

JTextField t13 = **new** JTextField("", 20);

JTextField t14 = **new** JTextField("0", 20);

JTextField t15 = **new** JTextField("", 20);

JTextField t16 = **new** JTextField(Tool\_Function.*data*(), 20);

JTextField t17 = **new** JTextField("100", 20);

JTextField t18 = **new** JTextField("(1 2或3)", 20);

JTextField t19 = **new** JTextField("100", 20);

JTextField t20 = **new** JTextField("汤姆", 20);

t16.setEditable(**false**);

JButton b1=**new** JButton("确认添加");

JButton b2=**new** JButton("返");

p1.add(u11);

p1.add(t11);

p2.add(u12);

p2.add(t12);

p3.add(u13);

p3.add(t13);

p4.add(u14);

p4.add(t14);

p5.add(u15);

p5.add(t15);

p6.add(u16);

p6.add(t16);

p7.add(u17);

p7.add(t17);

p8.add(u18);

p8.add(t18);

p9.add(u19);

p9.add(t19);

p10.add(u20);

p10.add(t20);

p11.add(b2);

p11.add(b1);

FreedomPane freedompane=**new** FreedomPane();

freedompane.setLayout(**new** GridLayout(12, 1));

freedompane.add(p1);

freedompane.add(p2);

freedompane.add(p3);

freedompane.add(p4);

freedompane.add(p5);

freedompane.add(p6);

freedompane.add(p7);

freedompane.add(p8);

freedompane.add(p9);

freedompane.add(p10);

freedompane.add(p11);

JPanel mainpane=**new** JPanel();

mainpane.add(freedompane);

mainpane.setLayout(**new** CardLayout());

jframe.setContentPane(mainpane);

jframe.setBounds(400,100, 600, 600);

b1.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

String s1=t11.getText();

String s2=t12.getText();

String s3=t13.getText();

String s4=t14.getText();

String s5=t15.getText();

String s6=t16.getText();

String s7=t17.getText();

String s8=t18.getText();

String s9=t19.getText();

String s10=t20.getText();

**if**(s1==**null**||"".equals(s1)||

s2==**null**||"".equals(s2)||

s3==**null**||"".equals(s3)||

s4==**null**||"".equals(s4)||

s5==**null**||"".equals(s5)||

s6==**null**||"".equals(s6)||

s7==**null**||"".equals(s7)||

s8==**null**||"".equals(s8)||

s9==**null**||"".equals(s9)||

s10==**null**||"".equals(s10)){ //判断属性信息是否正确

JOptionPane.*showMessageDialog*(**null**, "用户信息输入错误！！！",

"☆★提示信息☆★", JOptionPane.***INFORMATION\_MESSAGE***);

**return**;

}

**if**(Manager\_AllUser\_Add\_Function.*check*(s1)){

JOptionPane.*showMessageDialog*(**null**, "该卡号已经存在！！！",

"☆★提示信息☆★", JOptionPane.***INFORMATION\_MESSAGE***);

}

**if**(Manager\_AllUser\_Add\_Function.*add*(s1,s2,s3,s4,s5,s6,s7,s8,s9,s10)>0){

JOptionPane.*showMessageDialog*(**null**, "插入用户信息成功！",

"☆★提示信息☆★", JOptionPane.***INFORMATION\_MESSAGE***);

}**else**{

JOptionPane.*showMessageDialog*(**null**, "插入用户信息失败！！！",

"☆★提示信息☆★", JOptionPane.***INFORMATION\_MESSAGE***);

}

**new** Manager\_User\_AllUser();

jframe.dispose();

}

});

b2.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

**new** Manager\_User\_AllUser();

jframe.dispose();

}

});

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

jframe.setVisible(**true**);

}

修改会员界面如图4-10所示：

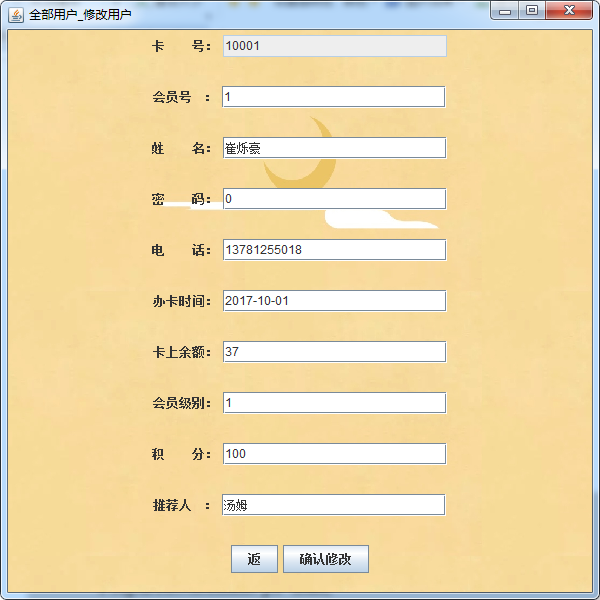


图4-10 修改会员

修改会员主要代码如下：

public static Object[] iddquery(String id){

Object[] os=new Object[10];

Connection conn=null;

PreparedStatement ps=null;

ResultSet rs=null;

try{

conn=DBConnection.getConnection();

String sql="select Vip\_No,VipAdminis\_No,VipAdminis\_Name,Vip\_passwd,VipAdminis\_Tel,"

+ "VipAdminis\_Time,VipAdminis\_Money,VipAdminis\_Level,VipAdminis\_Score,Vip\_peo from VipAdminis WHERE Vip\_No=?";

ps=conn.prepareStatement(sql);

ps.setString(1, id);

rs=ps.executeQuery();

if(rs.next()){

os[0] = rs.getString(1); //卡号

os[1] = rs.getString(2); //会员号

os[2] = rs.getString(3); //姓名

os[3] = rs.getString(4); //密码

os[4] = rs.getString(5); //电话

os[5] = rs.getString(6); //办卡时间

os[6] = rs.getString(7); //余额

os[7] = rs.getString(8); //级别

os[8] = rs.getString(9); //积分

os[9] = rs.getString(10);

}

}catch (SQLException e){

JOptionPane.showMessageDialog(null, "查询user指定用户信息时发生异常", "☆★提示信息☆★",

JOptionPane.INFORMATION\_MESSAGE);

e.printStackTrace();

}finally{

DBConnection.close(conn);

}

return os;

}

增加用户消费记录界面如图4-11所示：



**图4-11** 增加用户消费记录

增加用户消费记录主要代码如下：

**publicstaticint** add(String s1,String s2,String s3,String s4){

Connection conn=**null**;//增添消费记录

PreparedStatement ps=**null**;

**int**rows=0;

**try**{

conn=DBConnection.*getConnection*();

String sql="insert into vipout(Vip\_No,VipAdminis\_Name,Vip\_Money,Vip\_Time) "

+ "values(?,?,?,?)";

ps=conn.prepareStatement(sql);

ps.setString(1,s1);

ps.setString(2,s2);

ps.setString(3,s3);

ps.setString(4,s4);

rows=ps.executeUpdate();

}**catch**(Exception e){

JOptionPane.*showMessageDialog*(**null**, "插入消费记录时发生异常！！！",

"☆★提示信息☆★", JOptionPane.***INFORMATION\_MESSAGE***);

}**finally**{

DBConnection.*close*(ps);

DBConnection.*close*(conn);

}

**return**rows;

}

增加用户储值记录界面如图4-12所示：



**图4-12** 增加用户储值记录

增加用户储值记录主要代码如下：

public Manager\_In\_Add\_Function(String id,String name){//用户储值表

JFrame jframe=new JFrame("添加"+name+"的储值记录");

JPanel p1=new JPanel();

JPanel p2=new JPanel();

JPanel p3=new JPanel();

JPanel p4=new JPanel();

JPanel p5=new JPanel();

JPanel p6=new JPanel();

p1.setOpaque(false);

p2.setOpaque(false);

p3.setOpaque(false);

p4.setOpaque(false);

p5.setOpaque(false);

p6.setOpaque(false);

JLabel u11 = new JLabel("卡 号：");

JLabel u12 = new JLabel("姓 名：");

JLabel u13 = new JLabel("储值金额：");

JLabel u14 = new JLabel("备 注：");

JLabel u15 = new JLabel("时 间：");

JTextField t11 = new JTextField(id, 20);

JTextField t12 = new JTextField(name, 20);

JTextField t13 = new JTextField("", 20);

JTextField t14 = new JTextField("", 20);

JTextField t15 = new JTextField(Tool\_Function.data(), 20);

t11.setEditable(false);

t12.setEditable(false);

t15.setEditable(false);

JButton b1=new JButton("确认添加");

JButton b2=new JButton("返回");

p1.add(u11);

p1.add(t11);

p2.add(u12);

p2.add(t12);

p3.add(u13);

p3.add(t13);

p4.add(u14);

p4.add(t14);

p5.add(u15);

p5.add(t15);

p6.add(b2);

p6.add(b1);

FreedomPane freedompane=new FreedomPane();

freedompane.setLayout(new GridLayout(6, 1));

freedompane.add(p1);

freedompane.add(p2);

freedompane.add(p3);

freedompane.add(p4);

freedompane.add(p5);

freedompane.add(p6);

JPanel mainpane=new JPanel();

mainpane.add(freedompane);

mainpane.setLayout(new CardLayout());

jframe.setContentPane(mainpane);

b1.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

String s3=t13.getText();

String s4=t14.getText();

String s5=t15.getText();

if(Manager\_In\_Add\_Function.add(id,name,s3,s4,s5)>0){

JOptionPane.showMessageDialog(null, "插入用户信息成功！",

"☆★提示信息☆★", JOptionPane.INFORMATION\_MESSAGE);

if(!Manager\_In\_Add\_Function.In\_Add\_Yuetongbu(id, s3)){

JOptionPane.showMessageDialog(null, "储值表同步失败！",

"☆★提示信息☆★", JOptionPane.INFORMATION\_MESSAGE);

}

}else{

JOptionPane.showMessageDialog(null, "插入用户信息失败！！！",

"☆★提示信息☆★", JOptionPane.INFORMATION\_MESSAGE);

}

new Manager\_Tongjishuju\_InMoney(id,name);

jframe.dispose();

}

});

b2.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

new Manager\_Tongjishuju\_InMoney(id,name);

jframe.dispose();

}

});

jframe.setSize(600,600);

jframe.setLocation(400,100);

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

jframe.setVisible(true);

}

## 4.4 系统的发布

会员管理系统的发布，就是将本java应用程序进行发布，之后就可以安装。这样自己就可以不再带着冗余的应用程序来回的奔波，省了很多的事，当用到它的时候直接安装就可以的。这个不足之处就是发布的没那么的规范，还有待于改进。

# 第5章 结束语

本章主要是对二级实践课题的总结，首先介绍课题完成情况和心得、然后就是这个系统的不足和改进。

本次实践课题在韩玉民老师的悉心指导和严格要求下已完成，经过两个月的努力，最终较好地完成的课题开发，实现了要求的系统功能。

两个月以来忙碌紧张而又有条不紊的实践使我有机会对本专业的基本理论、专业知识和基本技术有了更深入的了解和体会，使我在这学期中所学到的知识得到了系统和升华，真正达到了学以致用。

（1）从心态方面来说，在整个课程设计的过程中我学到了做任何事情所要有的态度经过这项二级课题的亲身研究，我从中学习到了很多。在整个课程设计的过程中我学到了做任何事情所要有的态度和心态，首先我明白了做学问要一丝不苟，对于出现的任何问题和偏差都不要轻视，要通过正确的途径去解决，在做事情的过程中要有耐心和毅力，不要一遇到困难就打退堂鼓，只要坚持下去就可以找到思路去解决问题的。在工作中要学会与人合作的态度，认真听取别人的意见，这样做起事情来就可以事倍功半。

（2）从实践方面来说，我充分的体会到了“实践出真知”这一点，书本上的知识是不够的，只有把理论与实践相结合才能够真正的学到知识。在一个系统开发的过程中编码不是重要的，重要的是分析系统、建立起系统模型。经过三周的努力，个人通讯录管理系统终于完成了。这个课题不像以往所做的课题，这里不仅仅包括了书本上的知识，这里大部分运用了课外知识，所以在做课题之前，我从图书馆借了许多关于系统开发的书本，在做课题之中，这些课本对我的帮助很大，这里有课本以外的知识，所以我认为学习不仅仅是书本上的知识，只有你通过实践和训练才能把书本上的知识扩展和应用 ，才能学到课本以外的知识，实践出真知这句话真的不错。还有就是做课题并不是那么的容易，在成功之前我经过了许多次的更改，每一次都是一种提高。我就得在错误中学到的知识更加的有用。在做课题之中，我也遇到了许多的问题，有问题是好事，有问题才有提高，解决问题找老师，在韩老师的指导下，问题解决了。

（3）系统的不足之处。

1）查询功能较为单一，没有按时间段查询储值记录和消费记录。

2）由于时间比较紧迫，在做开发设计的时候，考虑不够充分，导致系统在开发过程中要对系统的流程结构做修改，这对开发流程的影响是非常大的。

3）因为项目开发经验不足，在代码的精简上做的有些欠缺，代码的重利用率不搞，重复的地方比较多，而且系统的可读性也不是好。

4）由于没有测试经验，对系统的抗压能力，稳定性，安全性不是很了解。以后会把系统的抗压性、稳定性调到最高

（4）下一步系统改进计划：增加按时间段查询消费记录和储值记录的功能。

一个系统的开发不可能是一下子就成功的，经过了两个月的实战，这个会员管理系统终于完成了。这一过程。刚开始时，为了在做课题就对这个课题开始研究，借了许多的课外书，有做了一些想法，但是在做课题之中，还是遇到了一些问题，虽说这个会员管理系统不是多么的麻烦，但是还是遇到了问题，还好问题不是多大。而且一个成功的系统要符合用户的需要，要以用户为中心，开发出用户想要的系统，而不能自以为是，想当然。

总之，这次课题实践让对所学知识有了更深的利用，学会了应用，实践能力有了很大提高。

在此非常感谢韩老师的指导和同学们的帮助。

# 指导老师评语

成绩：

指导老师签字：

年 月 日