**中北大学软件学院**

中期报告

实训名称: 面向对象与数据库课程设计

题目名称：基于用户管理员的2048小游戏的课设实现

专 业： 软件工程

班 级： 20130401班

小组成员

学号： 2013040111 姓名：田梓萱

学号： 2013040110 姓名：王超强

学号： 2013040139 姓名：杜庭巍

学号： 2013040136 姓名： 袁圣淇

指导教师： 李华玲

2022 年 1 月

**1 详细设计**

**1.1 2048主界面**

用户在点击运行程序时，会弹出该界面，选择进行“国服”或“外服”，如图1.1所示。



图1.1 2048主界面

该界面实现代码如下：

public class GServiceLChoose extends JFrame implements IData {

static final long serialVersionUID = -6718310314384392563L;

Toolkit tk = Toolkit.getDefaultToolkit();

Image img = getTk().getImage(GPictureIcon);

JFrame jf;

public GServiceLChoose(String st) {

init(st);

}

@Override

public void init(String st) {

jf = new JFrame(st);

jf.setIconImage(getImg());

jf.setSize(500, 500);

jf.setLocationRelativeTo(null);

jf.setLayout(null);

ImageIcon image = new ImageIcon(GBackupPicture);

image.setImage(image.getImage().getScaledInstance(500, 500, Image.SCALE\_DEFAULT));

JLabel jLable = new JLabel(image);

jLable.setBounds(0, 0, 500, 500);

JPanel jp1 = new JPanel();

JPanel jp2 = new JPanel();

JPanel jp3 = new JPanel();

JLabel lable = new JLabel("2048游戏中心");

JLabel lable1 = new JLabel("2048GameCenter");

lable.setFont(font2);

lable1.setFont(font3);

jp1.add(lable);

jp1.add(lable1);

jp1.setBounds(75, 85, 300, 300);

jf.add(jp1);

JButton jb1 = new JButton("国服(CN service)");

JButton jb2 = new JButton("外服(US service)");

jb1.setFont(font0);

jb2.setFont(font0);

Dimension dms = new Dimension(190, 50);

jb1.setPreferredSize(dms);

jb2.setPreferredSize(dms);

jp2.add(jb1);

jp3.add(jb2);

jp2.setBounds(15, 310, 210, 300);

jp3.setBounds(255, 310, 210, 300);

jf.add(jp2);

jf.add(jp3);

jp1.setOpaque(false);

jp2.setOpaque(false);

jp3.setOpaque(false);

jf.add(jLable);

jf.setResizable(false);

jf.setVisible(true);

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

**1.2 用户注册界面**

用户在注册时，需要通过该界面输入用户名、真实姓名、密码、身份证号、性别，然后点击“注册”按钮进行注册，如图1.2所示。

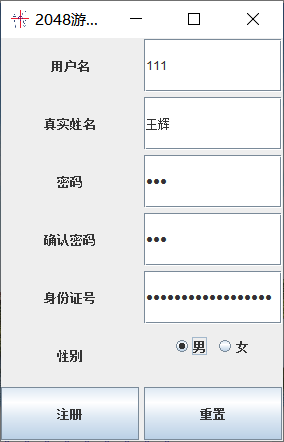


图1.2 注册界面

注册实现代码如下：

/\*数据库开始\*/

try {

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

Connection conn = DriverManager.getConnection(SqlMessage.Sqlmessage\_URL,

SqlMessage.Sqlmessage\_USER, SqlMessage.Sqlmessage\_PASSWORD);

Statement stmt = conn.createStatement();

ResultSet rs = stmt.executeQuery("select count(\*) from t\_user where Tname='" + getT\_id().getText() + "';");

while (rs.next()) {

int panduan = rs.getInt(1);

if (panduan == 1) {

setPanduancf("cf");

break;

} else {

setPanduancf("ko");

}

}

} catch (Exception e\_2) {

JOptionPane.showMessageDialog(null, "数据库综合错误，如果你看到此弹窗，请立刻联系我。",

"ERROR!", JOptionPane.ERROR\_MESSAGE);

e\_2.printStackTrace();

}

/\*数据库结束\*/

if (getPanduancf().equals("cf")) {

JOptionPane.showMessageDialog(null, "用户名重复，请重新注册！",

"ERROR", JOptionPane.ERROR\_MESSAGE);

getT\_id().setText("");

getT\_name().setText("");

getP\_password1().setText("");

getP\_password2().setText("");

getP\_card().setText("");

} else {

if (GIdCard.isIDNumber(String.valueOf(getP\_card().getPassword()))) {

if (GIdCard.isIDNumberSex(String.valueOf(getP\_card().getPassword())).equals(String.valueOf(getSex1()))) {

if (GIdCard.isIDNumberAge(String.valueOf(getP\_card().getPassword())) <= 85) {

if (String.valueOf(getP\_password1().getPassword()).equals(String.valueOf((getP\_password2().getPassword())))) {

UserCN u = new UserCN(String.valueOf(getT\_id().getText()), String.valueOf(getT\_name().getText()),

String.valueOf(getP\_password1().getPassword()), String.valueOf(getSex1()).charAt(0),

String.valueOf(getP\_card().getPassword()));

String datime = new SimpleDateFormat("yyyy-MM-dd|HH-mm-ss").format(new Date());

List<String> listuser = new ArrayList<>();

listuser.add(u.getId());

listuser.add(u.getName());

listuser.add(u.getPassword());

listuser.add(u.getCard());

listuser.add(String.valueOf(u.getSex()));

listuser.add(datime);

String data = "insert into t\_user(Tname, Trname, Tpasswd, Tcard, Tsex,Tage, Ttime,Status,StatusTime) values ('" +

listuser.get(0) + "','" + listuser.get(1) + "','" + listuser.get(2) + "','" +

listuser.get(3) + "','" + listuser.get(4) + "','" + GIdCard.isIDNumberAge(String.valueOf(getP\_card().getPassword()))

+ "','" + listuser.get(5) + "','" + 111 + "','" + listuser.get(5) + "');";

String data1 = "insert into t\_user1(Tname) values ('" +

listuser.get(0) + "');";

/\*数据库开始\*/

try {

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

Connection conn1 = DriverManager.getConnection(SqlMessage.Sqlmessage\_URL,

SqlMessage.Sqlmessage\_USER, SqlMessage.Sqlmessage\_PASSWORD);

Statement stmt1 = conn1.createStatement();

stmt1.execute(data);

stmt1.execute(data1);

JOptionPane.showMessageDialog(null, "注册成功！", " ", -1);

} catch (Exception e\_3) {

JOptionPane.showMessageDialog(null, "数据库综合错误，如果你看到此弹窗，请立刻联系我。",

"ERROR!", JOptionPane.ERROR\_MESSAGE);

e\_3.printStackTrace();

}

/\*数据库结束\*/

getJfm().dispose();

int nnn = JOptionPane.showConfirmDialog(null, "您的所有游戏数据将被自动保存" +

"在位于中国北京的服务器中,继续游戏默认同意我们的隐私协议,是否同意我们的隐私协议?", "隐私申明"

, JOptionPane.YES\_NO\_OPTION);

if (nnn == JOptionPane.YES\_OPTION) {

JOptionPane.showMessageDialog(null, "注意:" + "\n" +

"1.请通过菜单栏退出游戏." + "\n" + "2.默认会接着上一次记录." + "\n" +

"3.如果想重新开始请按Esc或R." + "\n" + "4.请勿在游戏中随意从主界面点X" +

"退出，否则会概率报错." + "\n" + "5.务必保证网络的状态良好，否则切换会卡顿，" +

"谢谢理解.",

"欢迎用户" + getT\_id().getText() + ",的登录提示", JOptionPane.CANCEL\_OPTION);

new Thread(new Runnable() {

@Override

public void run() {

GMusic gm = new GMusic(GMusicMessagePath);

while (true) {

gm.run();

}

}

}).start();

if (GIdCard.isIDNumberAge(String.valueOf(getP\_card().getPassword())) < 18) {

Timer timer = new Timer();

timer.schedule(new TimerTask() {

@Override

public void run() {

LoginCN.setFcm("stop");

JOptionPane.showMessageDialog(null, "您是未成年人，今日游玩时间已达上限，程序将会退出！",

"ERROR!", JOptionPane.ERROR\_MESSAGE);

}

}, GPlayTime);

}

new GView4CN(listuser.get(0)).showView();

} else if (nnn == JOptionPane.NO\_OPTION) {

System.exit(0);

}

} else {

JOptionPane.showMessageDialog(null, "在此我们很感谢您游玩我们的2048小游戏，但是由于您年事已高，为了您的健康，本游戏您将无法游玩，感谢您的谅解！",

"WARN!", JOptionPane.ERROR\_MESSAGE);

getT\_id().setText("");

getT\_name().setText("");

getP\_password1().setText("");

getP\_password2().setText("");

getP\_card().setText("");

System.exit(0);

}

} else {

getP\_password1().setText("");

getP\_password2().setText("");

}

} else {

JOptionPane.showMessageDialog(null, "身份证号码与性别不符！请重新输入！！！",

"ERROR", JOptionPane.ERROR\_MESSAGE);

getT\_id().setText("");

getT\_name().setText("");

getP\_password1().setText("");

getP\_password2().setText("");

getP\_card().setText("");

}

} else {

JOptionPane.showMessageDialog(null, "身份证号码不合法！请重新输入！！！",

"ERROR", JOptionPane.ERROR\_MESSAGE);

getT\_id().setText("");

getT\_name().setText("");

getP\_password1().setText("");

getP\_password2().setText("");

getP\_card().setText("");

}

}

}

});

setB\_reset(new JButton("重置"));

getB\_reset().addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

getT\_id().setText("");

getT\_name().setText("");

getP\_password1().setText("");

getP\_password2().setText("");

getP\_card().setText("");

JOptionPane.showMessageDialog(null, "重置成功！", " ", 1);

}

});

**1.3 用户登录界面**

用户在进行系统登录时，需要通过该界面输入用户名、密码，然后点击“登录”按钮进行登录，登录界面设计如图1.3.1所示。用户类主要用于封装用户的信息：帐号、密码和用户类型等信息，类图如图1.3.2所示。

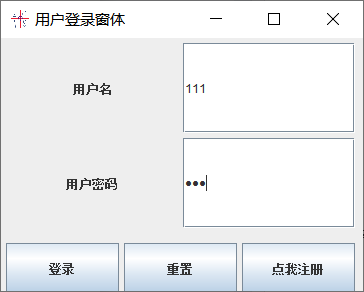


图1.3.1 用户登录界面

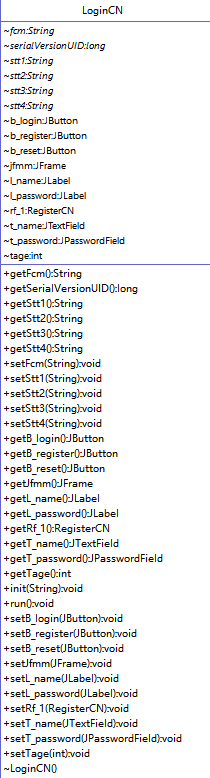


图1.3.2 用户uml类图

**1.4 管理员登录界面**

管理员在进行系统登录时，需要通过该界面输入用户名、密码，然后点击“登录”按钮进行登录，登录界面设计如图1.4.1所示，管理员界面类图如图1.4.2。

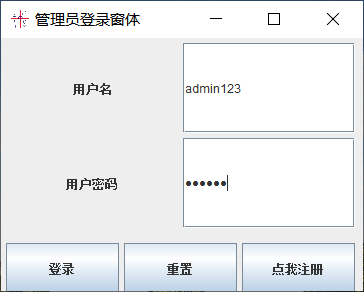
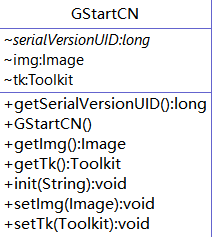


图1.4.1 管理员登录界面

图1.4.2 管理员uml类图

管理员 实现代码如下：

public class UserInfoCN extends JFrame implements IData, ActionListener {  
 static final long serialVersionUID = -4598519373915560300L;  
 int line = 0;  
 String admin = null;  
 JTable table\_book;  
 JFrame JFm = this;  
 JMenuBar jmb;  
 JMenu jm;  
 JMenuItem Moadd;  
 JMenuItem Delete;  
  
 UserInfoCN(String st) {  
 setAdmin(st);  
 init(st);  
 setVisible(true);  
 setSize(800, 600);  
 setIconImage(new ImageIcon(GMessage.GPictureIcon).getImage());  
 getJFm().setLocationRelativeTo(null);  
 setTitle("管理员" + st + "的用户信息展示区");  
 }  
  
 public static long getSerialVersionUID() {  
 return serialVersionUID;  
 }  
  
 @Override  
 public void init(String st) {  
 setLayout(null);  
 Object[] title = {"账号", "密码", "姓名", "性别", "身份证号", "注册时间"};//1.3.2.5.4.6  
 setJmb(new JMenuBar());  
 setJm(new JMenu("功能"));  
 setMoadd(new JMenuItem("更改"));  
 getMoadd().addActionListener(this);  
 setDelete(new JMenuItem("删除"));  
 getDelete().addActionListener(this);  
 getJm().add(getMoadd());  
 getJm().add(getDelete());  
 getJmb().add(getJm());  
 getJFm().setJMenuBar(getJmb());  
 List<String> list1 = new ArrayList<>();  
 /\*数据库开始\*/  
 try {  
 Class.forName("com.mysql.cj.jdbc.Driver");  
 Connection conn = DriverManager.getConnection(SqlMessage.Sqlmessage\_URL,  
 SqlMessage.Sqlmessage\_USER, SqlMessage.Sqlmessage\_PASSWORD);  
 Statement stmt = conn.createStatement();  
 ResultSet rs = stmt.executeQuery("select count(\*) from t\_user ;");  
 while (rs.next()) {  
 setLine(rs.getInt(1));  
 if (getLine() != 0) {  
 break;  
 }  
 }  
 } catch (Exception e) {  
 JOptionPane.showMessageDialog(null, "数据库综合错误，如果你看到此弹窗，请立刻联系我。",  
 "ERROR!", JOptionPane.ERROR\_MESSAGE);  
 e.printStackTrace();  
 }  
 /\*数据库结束\*/  
 try {  
 Class.forName("com.mysql.cj.jdbc.Driver");  
 Connection conn1 = DriverManager.getConnection(SqlMessage.Sqlmessage\_URL,  
 SqlMessage.Sqlmessage\_USER, SqlMessage.Sqlmessage\_PASSWORD);  
 Statement stmt1 = conn1.createStatement();  
 ResultSet rs1 = stmt1.executeQuery("select \* from t\_user ;");  
 while (rs1.next()) {  
 list1.add(String.valueOf(rs1.getString(1)));  
 list1.add(String.valueOf(rs1.getString(3)));  
 list1.add(String.valueOf(rs1.getString(2)));  
 list1.add(String.valueOf(rs1.getString(5)));  
 list1.add(String.valueOf(rs1.getString(4)));  
 list1.add(String.valueOf(rs1.getString(7)));  
 }  
 } catch (Exception e) {  
 JOptionPane.showMessageDialog(null, "数据库综合错误，如果你看到此弹窗，请立刻联系我。",  
 "ERROR!", JOptionPane.ERROR\_MESSAGE);  
 e.printStackTrace();  
 }  
 /\*数据库结束\*/  
 Object[][] bookList = new String[getLine()][6];  
 for (int j = 0; j < 6; j++)  
 for (int b = 0, k = 0; b < getLine(); b++, k += 6)  
 switch (j) {  
 case 0:  
 bookList[b][j] = list1.get(k);  
 break;  
 case 1:  
 bookList[b][j] = list1.get(k + 1);  
 break;  
 case 2:  
 bookList[b][j] = list1.get(k + 2);  
 break;  
 case 3:  
 bookList[b][j] = list1.get(k + 3);  
 break;  
 case 4:  
 bookList[b][j] = list1.get(k + 4);  
 break;  
 case 5:  
 bookList[b][j] = list1.get(k + 5);  
 break;  
 }  
 setTable\_book(new JTable(bookList, title));  
 JScrollPane pane = new JScrollPane();  
 pane.setBounds(20, 20, 750, 500);  
 pane.getViewport().add(getTable\_book());  
 getContentPane().add(pane);  
 }  
  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 if (e.getSource() == getMoadd()) {  
 getJFm().dispose();  
 new ModedCN(getAdmin());  
 } else if (e.getSource() == getDelete()) {  
 getJFm().dispose();  
 new DeletedCN(getAdmin());  
 }  
 }

**1.5 防沉迷游戏申明及隐私申明界面**

用户在点击开始游戏时，会弹出防沉迷游戏申明以及隐私申明的弹窗进行提示，点击“是”可以继续游戏，“否”则推出程序。如图1.5.1所示。

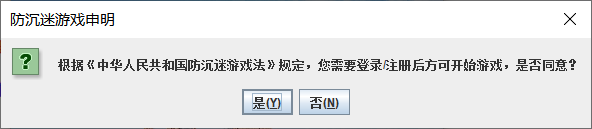
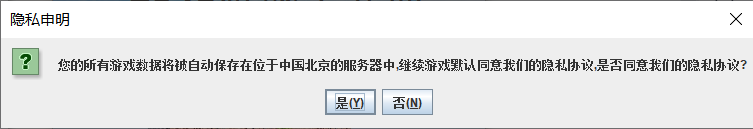


图1.5.1 防沉迷游戏申明界面

用户在进入游戏时按照应用上架开发规范守则，进行提示用户查看我们的隐私协议，同意后方可继续，如图1.5.2所示。

图1.5.2 隐私申明界面

该功能的实现代码：

jb1.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

int nnn = JOptionPane.showConfirmDialog(null, "根据您的选择，请您务必遵守中国(包括香港、澳门、台湾地区)" +

"及您所在地区的法律法规才可以继续游戏，本游戏为3+分类，您是否知悉？", "游戏申明"

, JOptionPane.YES\_NO\_OPTION);

if (nnn == JOptionPane.YES\_OPTION) {

SqlMessage.Sqlmessage\_URL = "jdbc:mysql://152.136.182.127:6133/game2048";

SqlMessage.Sqlmessage\_USER = "game2048";

SqlMessage.Sqlmessage\_PASSWORD = "12233434";

new GStartCN("2048游戏主界面");

jf.dispose();

} else if (nnn == JOptionPane.NO\_OPTION) {

System.exit(0);

}

}

});

jb2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

int nnn = JOptionPane.showConfirmDialog(null, "According to your choice, you must" +

" abide by the laws and regulations of the United States, the European Union and your region" +

" before you can continue the game. This game is classified as E. this game only retains " +

"the basic user data for storing your game information. Do you know?", "Game statement"

, JOptionPane.YES\_NO\_OPTION);

if (nnn == JOptionPane.YES\_OPTION) {

SqlMessage.Sqlmessage\_URL = "jdbc:mysql://107.148.206.54:6133/g2048";

SqlMessage.Sqlmessage\_USER = "g2048";

SqlMessage.Sqlmessage\_PASSWORD = "12233434";

new GStartEN("2048Game Main Interface");

jf.dispose();

} else if (nnn == JOptionPane.NO\_OPTION) {

System.exit(0);

}

}

});

}

int nnn = JOptionPane.showConfirmDialog(null, "您的所有游戏数据将被自动保存" +

"在位于中国北京的服务器中,继续游戏默认同意我们的隐私协议,是否同意我们的隐私协议?", "隐私申明"

, JOptionPane.YES\_NO\_OPTION);

if (nnn == JOptionPane.YES\_OPTION) {

JOptionPane.showMessageDialog(null, "注意:" + "\n" +

"1.请通过菜单栏退出游戏." + "\n" + "2.默认会接着上一次记录." + "\n" +

"3.如果想重新开始请按Esc或R." + "\n" + "4.请勿在游戏中随意从主界面点X" +

"退出，否则下次会无法登录." + "\n" + "5.务必保证网络的状态良好，否则切换会卡顿，" +

"谢谢理解." + "\n" + "6.您上次登录时间为:" + getStt4() + ".",

"欢迎用户" + getT\_name().getText() + ",的登录提示", JOptionPane.CANCEL\_OPTION);

**1.6 2048游戏运行界面**

用户可以通过键盘W,A,S,D或上,下,左,右来控制方块的移动，使其进行“2+2”、“4+4”、“8+8”等操作。不同的人有不同的习惯，通过多种方式操作游戏，可以符合每个用户的需求。图1.6所示模式为 4\*4 的2048游戏界面。

图1.6 4\*4游戏界面

1.6.1 功能

该程序存在着不同的玩法，可以通过“功能”这个选项来进行不同的选择。如图1.6.1所示



图1.6.1 功能

1.6.2 3\*3游戏界面

“3\*3”模式的2048如图1.6.2所示。

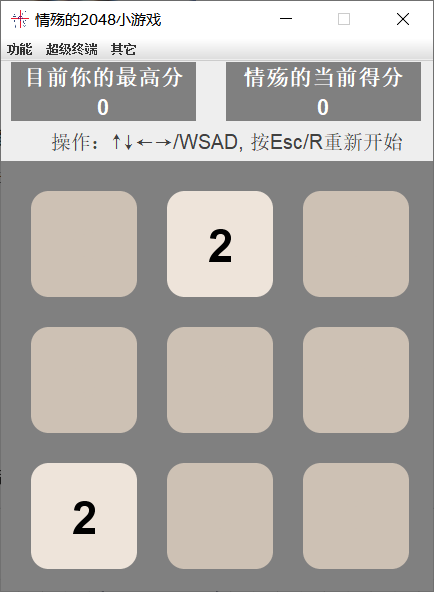


图1.6.2 3\*3游戏界面