**2020-2021第2学期《面向对象编程实训》项目设计总结表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 姓 名 | 呆萌小新@渊洁 | 学 号 | 201530404 | 专业班级 | 20软工4班 |
| 项目名称 | 网银财务管理 | | | 成 绩 |  |
| 项目需求描述 | 描述项目所包含的功能，列出并简单描述用到的主要技术  该项目的主要功能及其主要技术：   1. 可视化界面功能：   用户功能：  （1）登录 （2）注册 （3）首页（4）修改密码  （5）充值 （6）提现  （7）个人信息 （8）添加银行卡（9）银行卡信息  本项目类似于余额宝,微信账户等第三方网银平台.能实现用户的信息修改,不同卡号之间的转账功能.   1. 使用技术：   （1）MVC分包编程思想。  （2）java多态，继承，封装的思想。 （3）Java.GUI可视化技术。  （4）IO流序列化/反序列化的数据持久化技术。 | | | | |
| 设计过程 | 1.运用MVC思想  实体类创建用户类Users类型名称解释StringTel手机号（账户）StringPwd密码Stringname昵称StringRelName真实姓名StringcardNum身份证号Stringemail邮箱doublemoney余额银行卡Banks类型名称解释StringbankNum银行卡号Doublemoney余额StringBankName开户行管理员Admin类型名称解释StringAccount账户StringPwd密码给项目分层MVC开始是存在于桌面程序中的，M是指业务模型，V是指用户界面，C则是控制器，使用MVC的目的是将M和V的实现代码分离，从而使同一个程序可以使用不同的表现形式。比如一批统计数据可以分别用柱状图、饼图来表示。C存在的目的则是确保M和V的同步，一旦M改变，V应该同步更新。创建以下5个层com.baizhi.entity 实体类层 放置实体类Com.baizhi.dao dao层 操作数据Com.baizhi.service 服务层 处理业务Com.baizhi.controller 控制层 控制操作Com.baizhi.view 视图层 展示信息Com.baizhi.test 测试创建实体类部分代码用户public class Users implements Serializable {  private String tel;  private String pwd;  private String name;  private String relName;  private String cardNum;  private String email;  private double money;银行卡public class Banks implements Serializable {  private String bankNum;  private double money;  private String bankName;  private String tel;管理员/\*\*管理员\*/ public class Admin implements Serializable {//Serializable序列化接口  private String account;  private String pwd;创建dao层package com.baizhi.dao;  import com.baizhi.entity.Admin;  import java.io.\*; import java.util.ArrayList; import java.util.List;   /\*\*操作admin的数据\*/ public class AdminDao {  /\*\*读取数据\*/  public List<Admin> read(){  List<Admin> list=new ArrayList<>();  try {  FileInputStream file=new FileInputStream("admin.data");  ObjectInputStream in=new ObjectInputStream(file);  list= (List<Admin>) in.readObject();  in.close();  } catch (Exception e) {  e.printStackTrace();  }  return list;  }   /\*\*写入数据\*/  public boolean write(List<Admin> list){  try {  FileOutputStream file=new FileOutputStream("admin.data");  ObjectOutputStream out=new ObjectOutputStream(file);  out.writeObject(list);  out.close();  return true;//写入成功  } catch (Exception e) {  e.printStackTrace();  return false;//写入失败  }  }  }完成基本功能用户登陆UsersService private UsersDao dao=new UsersDao(); @Override public Users findUsersByTelAndPwd(String tel, String pwd) {  List<Users> list=dao.read();//读取数据  for(Users u:list){  String tels=u.getTel();  String pwds=u.getPwd();  if(tels.equals(tel)&&pwds.equals(pwd)){//手机号，密码相同  System.out.println("登陆成功！");  return u;  }  }  return null;//失败 }UsersControllerprivate UsersService usersService=new UsersServiceImpl(); /\*\*登陆\*/ public Users usersLogin(String tel,String pwd){  return usersService.findUsersByTelAndPwd(tel,pwd); }UsersLoginView/\*\*用户登陆界面\*/ public class UsersLoginView extends JFrame implements MouseListener {  private JPanel jp;//大组件  private JLabel top;//顶部  private JPanel bom;//底部  private JLabel nameJL,pwdJL;  private JTextField nameJF;//用户  private JPasswordField pwdJF;//密码框  private JButton login,reg;//按钮  public UsersLoginView(){  //设置大组件布局方式  jp=new JPanel();  jp.setLayout( new BorderLayout());  //顶部  top=new JLabel();  ImageIcon icon=new ImageIcon("img/logo.png");  icon.setImage(icon.getImage().getScaledInstance(300,50, Image.SCALE\_DEFAULT));  top.setIcon(icon);  top.setBorder(new EmptyBorder(20,70,20,30));  jp.add(top,BorderLayout.NORTH);  //底部  bom=new JPanel();  bom.setLayout(new GridLayout(3,2,20,20));//表格布局  bom.setBorder(new EmptyBorder(0,40,30,40));  //输入框  nameJL=new JLabel("手机号:",JLabel.RIGHT);  nameJL.setIcon(new ImageIcon("img/username\_icon.png"));  nameJL.setBorder(new EmptyBorder(0,0,0,20));  nameJF=new JTextField(20);  pwdJL=new JLabel("密 码:",JLabel.RIGHT);  pwdJL.setIcon(new ImageIcon("img/password\_icon.png"));  pwdJL.setBorder(new EmptyBorder(0,0,0,20));  pwdJF=new JPasswordField(20);  login=new JButton("登陆");  login.setName("login");  login.addMouseListener(this);//添加监听事件  reg=new JButton("注册");  reg.setName("reg");  reg.addMouseListener(this);//添加监听事件  bom.add(nameJL);  bom.add(nameJF);  bom.add(pwdJL);  bom.add(pwdJF);  bom.add(login);  bom.add(reg);  jp.add(bom,BorderLayout.CENTER);  this.add(jp);  this.setTitle("欢迎使用百知转账管理系统！");//标题  this.setSize(450,300);//大小  this.setLocationRelativeTo(null);//居中  this.setResizable(false);//不允许改变文件大小  this.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);//关闭方式  this.setVisible(true);//显示  }  @Override  public void mouseClicked(MouseEvent e) {  //发生事件的组件名称  String str=e.getComponent().getName();  if(str.equals("login")){  String tel=nameJF.getText();  String pwd=new String(pwdJF.getPassword());  if(tel.equals("")){  JOptionPane.showMessageDialog(null,"手机号不能为空");  }else if(pwd.equals("")){  JOptionPane.showMessageDialog(null,"密码不能为空");  }else {  UsersController usersController=new UsersController();  Users users=usersController.usersLogin(tel,pwd);  if(users==null){//失败  JOptionPane.showMessageDialog(null,"登陆失败！！！");  }else{  System.out.println("成功，进入主界面");  MainView mainView=new MainView(users);  dispose();//隐藏登陆页  }  }  }else{//注册  AddUsersView ad=new AddUsersView(this);  }  }注册功能Servicezh@Override public boolean addUsers(Users users) {  List<Users> list=dao.read();//读取数据  list.add(users);  return dao.write(list); }  @Override public boolean isExistTel(String tel) {  List<Users> list=dao.read();//读取数据  for (Users u:list){  if(u.getTel().equals(tel)){  return true;//手机号相同  }  }  return false; } @Override public boolean isExistEmail(String email) {  List<Users> list=dao.read();//读取数据  for (Users u:list){  if(u.getEmail().equals(email)){  return true;//邮箱相同  }  }  return false; }Controller/\*\*注册\*/ public boolean addUsers(String tel, String pwd, String name, String relName,  String cardNum, String email){  Users u=new Users(tel,pwd,name,relName,cardNum,email,0);  return usersService.addUsers(u); } /\*\*验证手机号是否存在\*/ public boolean isExistTel(String tel){  return usersService.isExistTel(tel); } /\*\*验证邮箱是否存在\*/ public boolean isExistEmail(String email){  return usersService.isExistEmail(email); }View/\*\*用户注册\*/ public class AddUsersView extends Dialog implements MouseListener {  private JPanel jp;//顶级组件   private JLabel telJL,pwdJL,nameJL,relNameJL,cardNumJL,emailJL;//左侧提示  private JTextField telJF,nameJF,relNameJF,cardNumJF,emailJF;//文本  private JPasswordField pwdJF;//密码  private JButton sub,clo; //按钮  public AddUsersView(Frame owner) {  super(owner);  jp=new JPanel();  jp.setBackground(Color.GRAY);  jp.setBorder(new EmptyBorder(20,20,20,20));  jp.setLayout(new GridLayout(7,2,20,20));   EmptyBorder jbo=new EmptyBorder(0,20,0,0);  EmptyBorder fbo=new EmptyBorder(0,0,0,20);   telJL=new JLabel("手机号:",JLabel.LEFT);  telJL.setIcon(new ImageIcon("img/phone\_icon.png"));  telJL.setBorder(jbo);  telJF=new JTextField(20);  telJF.setBorder(fbo);  jp.add(telJL);  jp.add(telJF);   pwdJL=new JLabel("密码:",JLabel.LEFT);  pwdJL.setIcon(new ImageIcon("img/password\_icon.png"));  pwdJL.setBorder(jbo);  pwdJF=new JPasswordField(20);  pwdJF.setBorder(fbo);  jp.add(pwdJL);  jp.add(pwdJF);   nameJL=new JLabel("昵称:",JLabel.LEFT);  nameJL.setIcon(new ImageIcon("img/name\_icon.png"));  nameJL.setBorder(jbo);  nameJF=new JTextField(20);  nameJF.setBorder(fbo);  jp.add(nameJL);  jp.add(nameJF);   relNameJL=new JLabel("真实姓名:",JLabel.LEFT);  relNameJL.setIcon(new ImageIcon("img/username\_icon.png"));  relNameJL.setBorder(jbo);  relNameJF=new JTextField(20);  relNameJF.setBorder(fbo);  jp.add(relNameJL);  jp.add(relNameJF);   cardNumJL=new JLabel("身份证:",JLabel.LEFT);  cardNumJL.setIcon(new ImageIcon("img/idc\_icon.png"));  cardNumJL.setBorder(jbo);  cardNumJF=new JTextField(20);  cardNumJF.setBorder(fbo);  jp.add(cardNumJL);  jp.add(cardNumJF);   emailJL=new JLabel("邮箱:",JLabel.LEFT);  emailJL.setIcon(new ImageIcon("img/msg\_icon.png"));  emailJL.setBorder(jbo);  emailJF=new JTextField(20);  emailJF.setBorder(fbo);  jp.add(emailJL);  jp.add(emailJF);   sub=new JButton("提交");  sub.setName("sub");  sub.addMouseListener(this);  jp.add(sub);  clo=new JButton("取消");  clo.setName("clo");  clo.addMouseListener(this);  jp.add(clo);  this.add(jp);  this.setTitle("百知转账管理系统-->用户注册");//标题  this.setSize(400,400);//大小  this.setLocationRelativeTo(null);//居中  this.setResizable(false);//不允许改变文件大小  this.setVisible(true);//显示  }   @Override  public void mouseClicked(MouseEvent e) {  String str=e.getComponent().getName();  switch (str){  case "sub":  //获取输入的内容  String tel=telJF.getText();  String pwd=new String(pwdJF.getPassword());  String name=nameJF.getText();  String relName=relNameJF.getText();  String cardNum=cardNumJF.getText();  String email=emailJF.getText();  if(tel.equals("")){  JOptionPane.showMessageDialog(null,"手机号不能为空！");  break;  }  if(pwd.equals("")){  JOptionPane.showMessageDialog(null,"密码不能为空！");  break;  }  if(name.equals("")){  JOptionPane.showMessageDialog(null,"昵称不能为空！");  break;  }  if(relName.equals("")){  JOptionPane.showMessageDialog(null,"真实姓名不能为空！");  break;  }  if(cardNum.equals("")){  JOptionPane.showMessageDialog(null,"身份证不能为空！");  break;  }  if(email.equals("")){  JOptionPane.showMessageDialog(null,"邮箱不能为空！");  break;  }  UsersController con=new UsersController();  if(con.isExistTel(tel)){  JOptionPane.showMessageDialog(null,"手机号已注册！");  break;  }  if(con.isExistEmail(email)){  JOptionPane.showMessageDialog(null,"邮箱已注册！");  break;  }  boolean flag=con.addUsers(tel,pwd,name,relName,cardNum,email);  if(flag){  JOptionPane.showMessageDialog(null,"注册成功！");  dispose();  }else{  JOptionPane.showMessageDialog(null,"注册失败！");  }   break;  case "clo":  dispose();  break;  }  }主页面设计package com.baizhi.view;  import com.baizhi.entity.Users;  import javax.swing.\*; import javax.swing.border.EmptyBorder; import java.awt.\*; import java.awt.event.MouseEvent; import java.awt.event.MouseListener;  /\*\*主界面\*/ public class MainView extends JFrame implements MouseListener {  private JPanel jpJP,topJP,leftJP,conJP,indexJP;  private JLabel logoJL,welJL,indexJL;  private JButton indexJB,showUserJB,pwdJB,banksJB,addMoneyJB,deleteMoneyJB,moveMoneyJB,updateUserJB;  Users users;  public MainView(Users users){  this.users=users;  jpJP=new JPanel();  jpJP.setLayout(new BorderLayout());//边框布局  //顶部  topJP=new JPanel();  logoJL=new JLabel();  ImageIcon lo=new ImageIcon("img/logo.png");  lo.setImage(lo.getImage().getScaledInstance(200,30,Image.SCALE\_DEFAULT));  logoJL.setIcon(lo);  welJL=new JLabel("欢迎用户: "+users.getName());  topJP.add(logoJL);  topJP.add(welJL);  jpJP.add(topJP,BorderLayout.NORTH);  //左侧导航  leftJP=new JPanel();  leftJP.setLayout(new GridLayout(8,1,20,20));  EmptyBorder jb=new EmptyBorder(10,0,10,20);  indexJB=new JButton("首页");  indexJB.setName("indexJB");  indexJB.setBorder(jb);  indexJB.addMouseListener(this);  leftJP.add(indexJB);   showUserJB=new JButton("用户信息");  showUserJB.setName("showUserJB");  showUserJB.setBorder(jb);  showUserJB.addMouseListener(this);  leftJP.add(showUserJB);   pwdJB=new JButton("修改密码");  pwdJB.setName("pwdJB");  pwdJB.setBorder(jb);  pwdJB.addMouseListener(this);  leftJP.add(pwdJB);   banksJB=new JButton("银行卡");  banksJB.setName("banksJB");  banksJB.setBorder(jb);  banksJB.addMouseListener(this);  leftJP.add(banksJB);   addMoneyJB=new JButton("充值");  addMoneyJB.setName("addMoneyJB");  addMoneyJB.setBorder(jb);  addMoneyJB.addMouseListener(this);  leftJP.add(addMoneyJB);   deleteMoneyJB=new JButton("提现");  deleteMoneyJB.setName("deleteMoneyJB");  deleteMoneyJB.setBorder(jb);  deleteMoneyJB.addMouseListener(this);  leftJP.add(deleteMoneyJB);   moveMoneyJB=new JButton("转账");  moveMoneyJB.setName("moveMoneyJB");  moveMoneyJB.setBorder(jb);  moveMoneyJB.addMouseListener(this);  leftJP.add(moveMoneyJB);   updateUserJB=new JButton("管理用户");  updateUserJB.setName("updateUserJB");  updateUserJB.setBorder(jb);  updateUserJB.addMouseListener(this);  leftJP.add(updateUserJB);   jpJP.add(leftJP,BorderLayout.WEST);   conJP=new JPanel();  conJP.setBorder(new EmptyBorder(10,10,10,0));  indexJP=new JPanel();  indexJP.setSize(500,450);  indexJL=new JLabel();  ImageIcon icon=new ImageIcon("img/banner1.jpg");  icon.setImage(icon.getImage().getScaledInstance(500,350,Image.SCALE\_DEFAULT));  indexJL.setIcon(icon);   indexJP.add(indexJL);  conJP.add(indexJP);  jpJP.add(conJP);  this.add(jpJP);   this.setSize(600,500);//大小  this.setLocationRelativeTo(null);//居中  this.setResizable(false);//不允许改变文件大小  this.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);//关闭方式  this.setVisible(true);//显示  }  //切换view  public void updateView(JPanel jp){  conJP.removeAll();  conJP.add(jp);  conJP.validate();  conJP.updateUI();  }  @Override  public void mouseClicked(MouseEvent e) {  String str=e.getComponent().getName();  switch (str){  case "indexJB":  updateView(indexJP);  break;  case "showUserJB":  ShowUserView showUserView=new ShowUserView(users);  updateView(showUserView);  break;  case "pwdJB":  break;  case "banksJB":  break;  case "addMoneyJB":  break;  case "deleteMoneyJB":  break;  case "moveMoneyJB":  break;  case "updateUserJB":  break;   }  }   }用户信息显示public class ShowUserView extends JPanel {  private JPanel top,bom;  private JLabel telJL,pwdJL,nameJL,relNameJL,cardNumJL,emailJL;//左侧提示  private JTextField telJF,nameJF,relNameJF,cardNumJF,emailJF;//文本  private JPasswordField pwdJF;//密码  public ShowUserView(Users u){  this.setLayout(new BorderLayout());  top=new JPanel();  JLabel topJl=new JLabel("个人信息展示");  top.add(topJl);  this.add(top,BorderLayout.NORTH);   //下边  bom=new JPanel();  bom.setLayout(new GridLayout(6,2,10,10));   //下边是复制的  EmptyBorder jbo=new EmptyBorder(10,50,0,0);  EmptyBorder fbo=new EmptyBorder(0,0,0,50);  telJL=new JLabel("手机号:",JLabel.CENTER);  telJL.setIcon(new ImageIcon("img/phone\_icon.png"));  telJL.setBorder(jbo);  telJF=new JTextField(20);  telJF.setText(u.getTel());  telJF.setBorder(fbo);  bom.add(telJL);//jp.add  bom.add(telJF);   pwdJL=new JLabel("密码:",JLabel.CENTER);  pwdJL.setIcon(new ImageIcon("img/password\_icon.png"));  pwdJL.setBorder(jbo);  pwdJF=new JPasswordField(20);  pwdJF.setBorder(fbo);  pwdJF.setText(u.getPwd());  bom.add(pwdJL);  bom.add(pwdJF);   nameJL=new JLabel("昵称:",JLabel.CENTER);  nameJL.setIcon(new ImageIcon("img/name\_icon.png"));  nameJL.setBorder(jbo);  nameJF=new JTextField(20);  nameJF.setBorder(fbo);  nameJF.setText(u.getName());  bom.add(nameJL);  bom.add(nameJF);   relNameJL=new JLabel("真实姓名:",JLabel.CENTER);  relNameJL.setIcon(new ImageIcon("img/username\_icon.png"));  relNameJL.setBorder(jbo);  relNameJF=new JTextField(20);  relNameJF.setBorder(fbo);  relNameJF.setText(u.getRelName());  bom.add(relNameJL);  bom.add(relNameJF);   cardNumJL=new JLabel("身份证:",JLabel.CENTER);  cardNumJL.setIcon(new ImageIcon("img/idc\_icon.png"));  cardNumJL.setBorder(jbo);  cardNumJF=new JTextField(20);  cardNumJF.setBorder(fbo);  cardNumJF.setText(u.getCardNum());  bom.add(cardNumJL);  bom.add(cardNumJF);   emailJL=new JLabel("邮箱:",JLabel.CENTER);  emailJL.setIcon(new ImageIcon("img/msg\_icon.png"));  emailJL.setBorder(jbo);  emailJF=new JTextField(20);  emailJF.setBorder(fbo);  emailJF.setText(u.getEmail());  bom.add(emailJL);  bom.add(emailJF);  this.add(bom,BorderLayout.CENTER);   }添加银行卡public class BanksServiceImpl implements BanksService {  private BanksDao dao=new BanksDao();  @Override  public boolean addBanks(Banks banks) {  List<Banks> list=dao.read();  list.add(banks);  return dao.write(list);  }public class BanksController {  private BanksService service=new BanksServiceImpl();  /\*\*添加\*/  public boolean addBanks(Banks banks){   return service.addBanks(banks);  }viewpublic class AddBanksView extends JPanel implements MouseListener {  private JPanel top,bom;  private JLabel bankNumJL,moneyJL, bankNameJL;  private JTextField bankNumJF,moneyJF, bankNameJF;  private JButton sub;  private Users users;  public AddBanksView(Users users){  this.users=users;  this.setLayout(new BorderLayout());  top=new JPanel();  top.setLayout(new GridLayout(3,2,20,20));   bankNumJL=new JLabel("卡号：",JLabel.CENTER);  bankNumJF=new JTextField(20);  top.add(bankNumJL);  top.add(bankNumJF);   bankNameJL=new JLabel("开户行：",JLabel.CENTER);  bankNameJF=new JTextField(20);  top.add(bankNameJL);  top.add(bankNameJF);   moneyJL=new JLabel("余额：",JLabel.CENTER);  moneyJF=new JTextField(20);  top.add(moneyJL);  top.add(moneyJF);  this.add(top,BorderLayout.CENTER);   bom=new JPanel();  sub=new JButton("提交");  sub.setName("sub");  sub.addMouseListener(this);  bom.add(sub);  this.add(bom,BorderLayout.SOUTH);    }   @Override  public void mouseClicked(MouseEvent e) {  String bankNum=bankNumJF.getText();  String money=moneyJF.getText();  String bankName=bankNameJF.getText();  if(bankNum.equals("")){  JOptionPane.showMessageDialog(null,"卡号不能为空！");  }  if(money.equals("")){  JOptionPane.showMessageDialog(null,"余额不能为空！");  }  if(bankName.equals("")){ | | | | |
| 关键问题 | 1. 对于刷新金额的问题   封装一个用户方法,其中有get(),set()方法  将提现前将要写入文件的用户数据获取,提供给显示金额界面.  重新设定文本框的值 | | | | |
| 总结展望 | 让我进一步巩固了java基础,虽然用到的技术很简单,很基础.但是对于java基础的巩固很全面.本次项目也让我深刻感受到设计思路应当优先编程.无目的的编程就是白费功夫.此后我将会少走弯路.路漫漫其修远兮,吾将上下求索. | | | | |

提交日期： 年 月 日