2023060832065
2023060932082

PG DIPLOMA IN FULL STACK JAVA DEVELOPEMENT

CRANES VARSITY, BANGALORE

SEPTEMBER - 2023

A

PROJECT REPORT

on

ATM SOFTWARE

Submitted in partial fulfilment of the requirements of the degree of

POST GRADUATE DIPLOMA

Under Guidance of :

Mrs Chama Tiwari, Trainer
Cranes Varsity, Bangalore

Submitted by:

Vivekanand (R2023060932082)
Akanksha (R2023060832065)

PG DIPLOMA IN FULL STACK JAVA DEVELOPEMENT

CRANES VARSITY, BANGALORE

SEPTEMBER- 2023

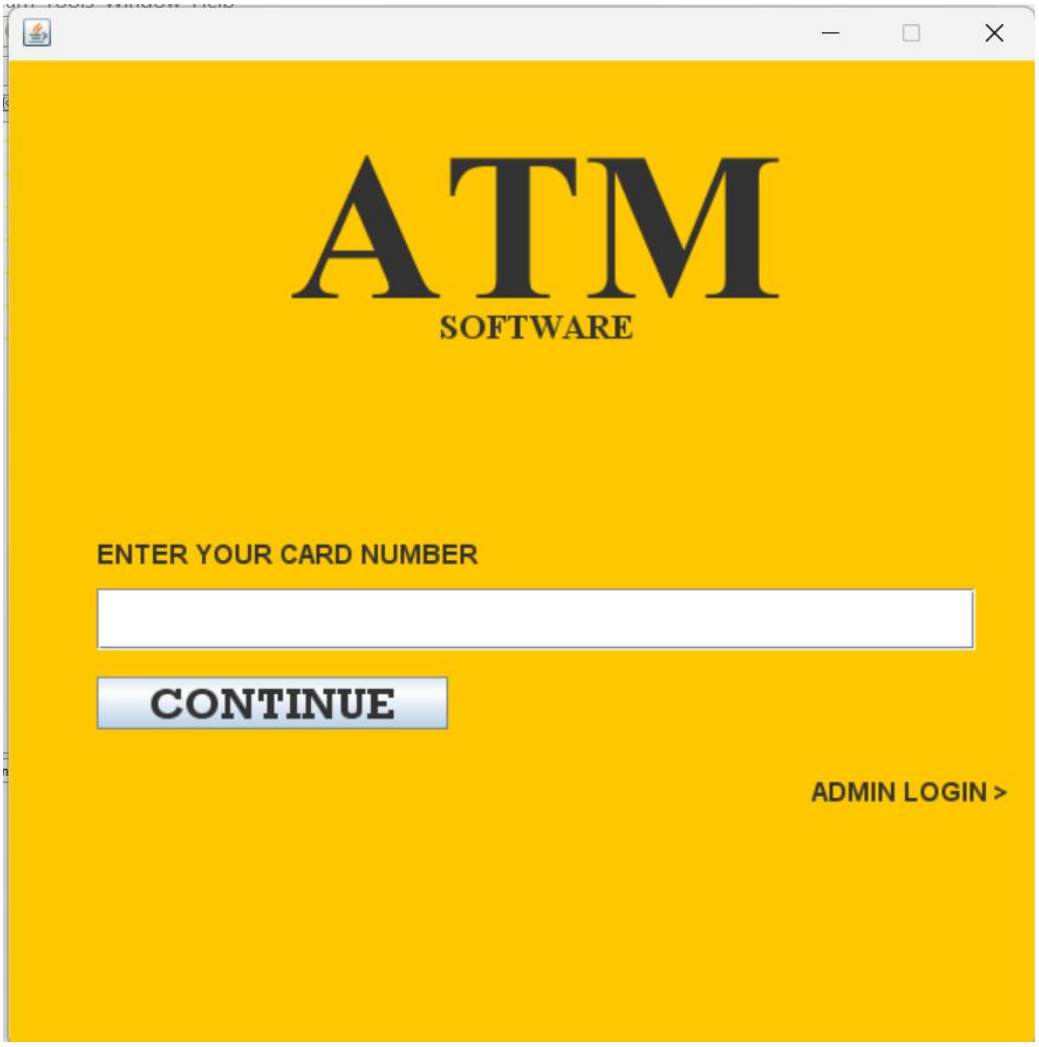
TABLE OF CONTENT

Abstract

This report outlines the development and implementation of a simple Atm Software application using the java swings,database and jdbc, package in Java. The Atm Software is capable ofperforming the logics like ,withdraw, entering card no, balance enquiry, feteching mini statement and etc operations can be done. The application has been designed with a user-friendly that makes it easy for the banking operations.

Chapter 1: INTRODUCTION

This report outlines the development and implementation of a simple software of atm machine application using the java swings,jdbc and rdbms .This application is capable of checking the balance,withdrawing money,deposit money and etc..operations. The application has been designed with a user-friendly that makes it easy for the banking operations.



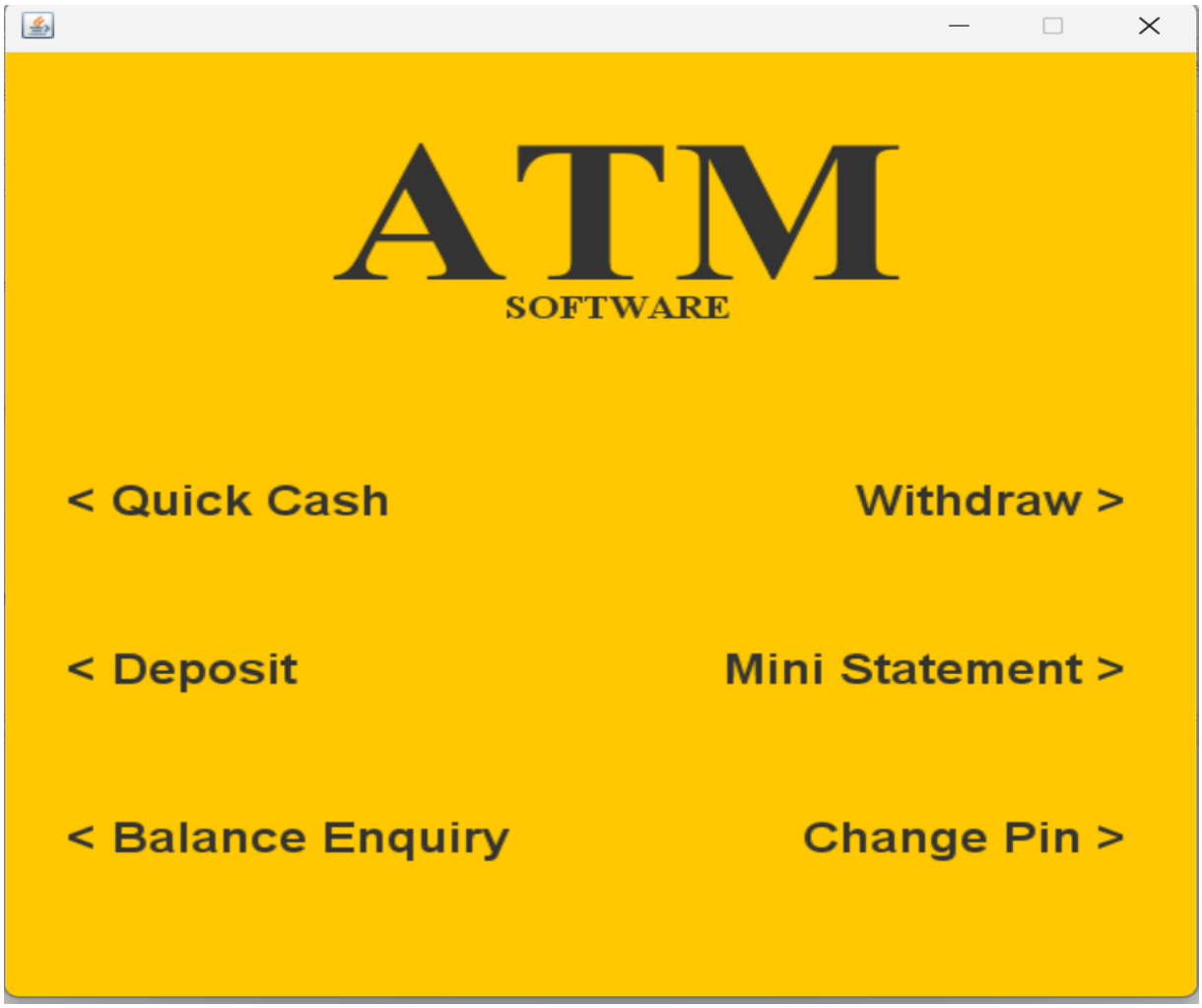


Fig. 1.1: Operation Pane

Chapter 2: SYSTEM REQUIREMENTS

To run the Simple ATM software application, the following system requirements must be met:

- Operating System: Windows 7 or later, macOS 10.10 or later, Linux
- Processor: 1.8 GHz or faster
- Memory: 4 GB RAM or more
- Hard Disk: 800 MB of free space or more
- Java SE Development Kit (JDK) 8 or later

Chapter 3: DESIGN

The Simple ATM software has been designed with a clean and intuitive interfacethat makes it easy for Banking Operations. The main components of the application include a JFrame,JPasswordfield, JPanel, and JTextField, all of which are part of the java swing package. The JFrame component provides the main window for the application, while the JPanel component provides the area for the buttons and text field. The JTextField component is used to display the results of the calculations.

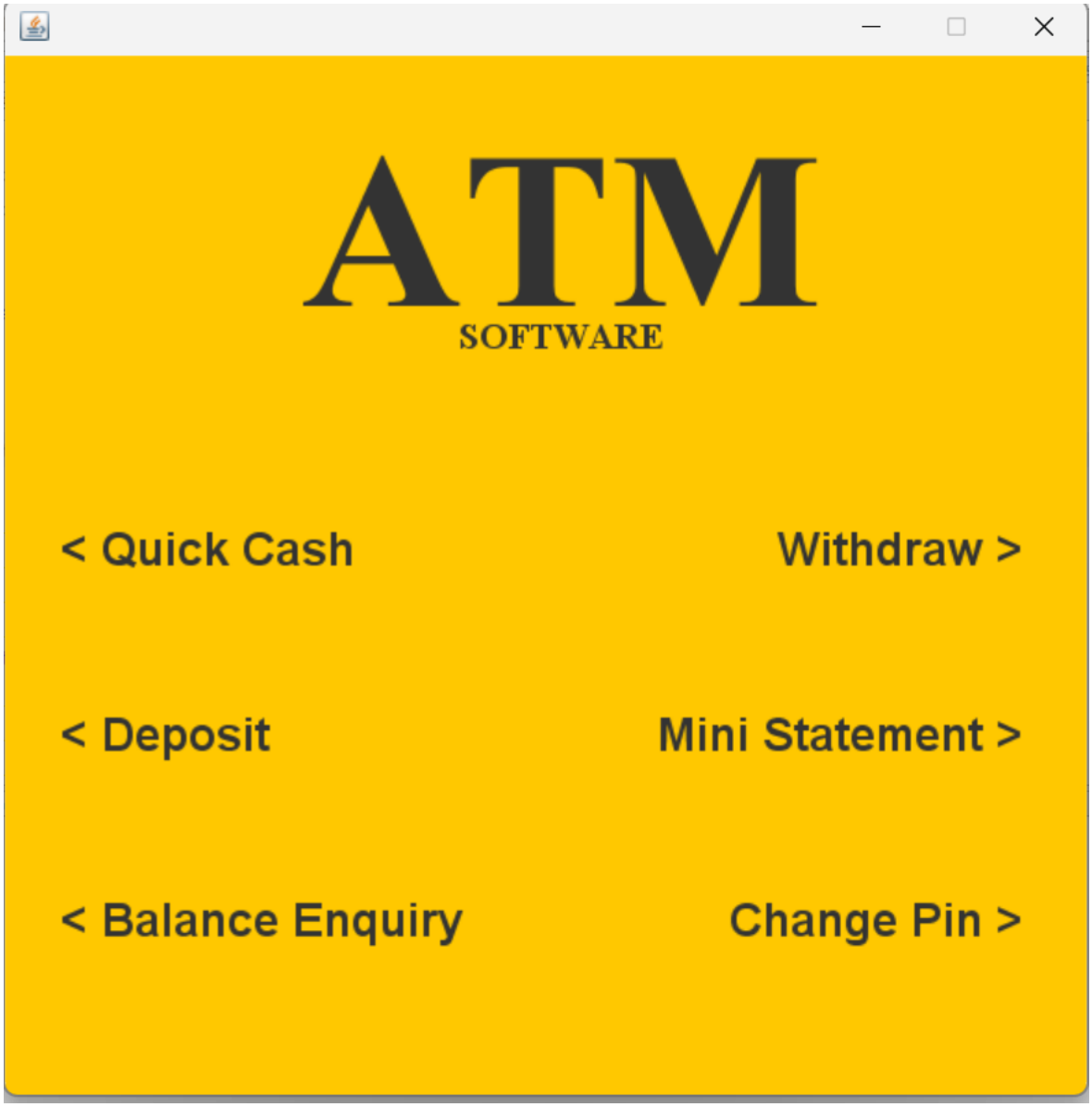


Fig. 3.1: operation pane

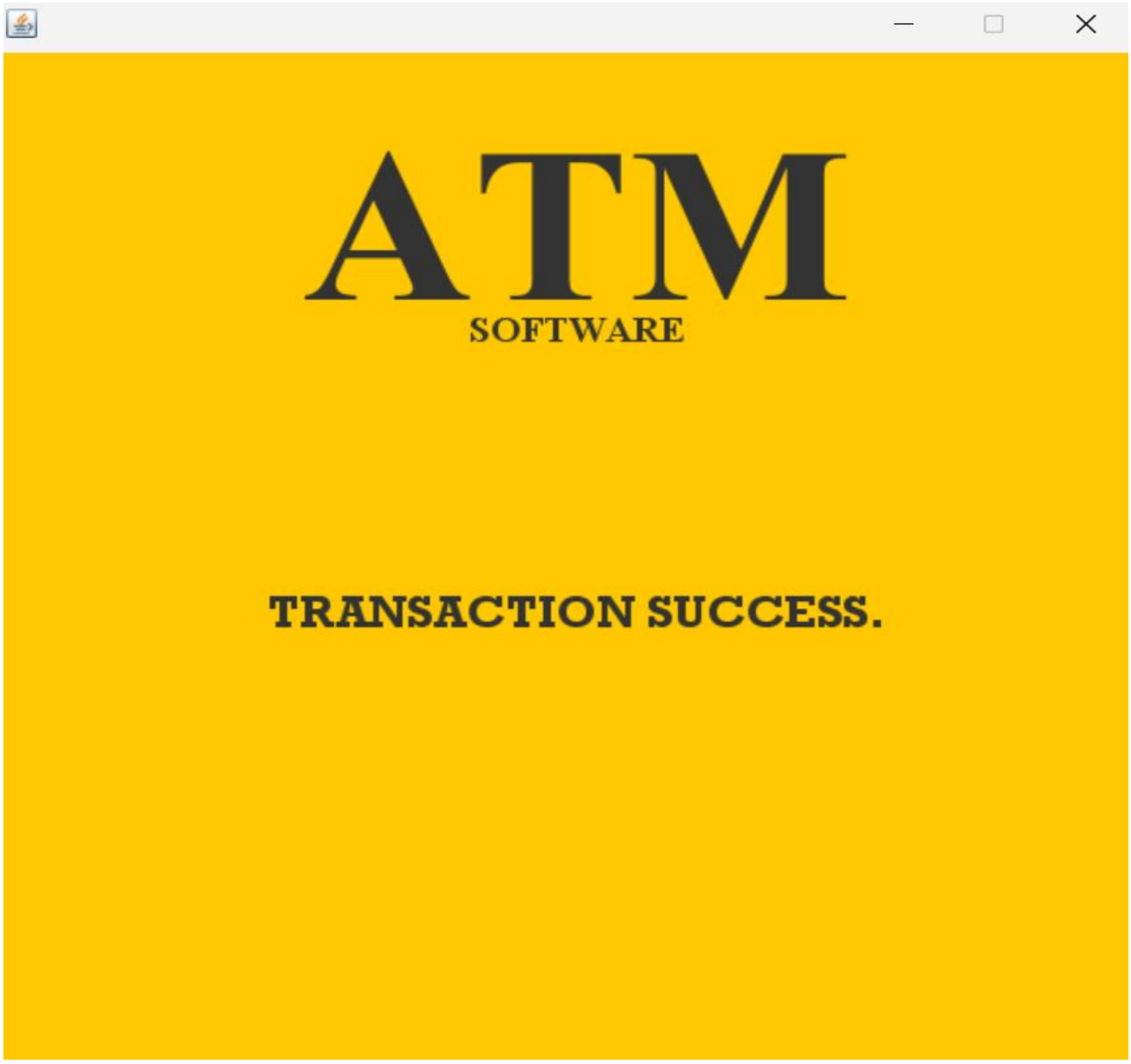


Fig.3.2 : Transaction

Chapter 4:

WORKING OF PROJECT

The Simple ATM software has been implemented in Java using the java swing, jdbc and database. The application is event-driven, with each button press generating an event that is handled by an ActionListener object. The ActionListener objects are responsible for updating the contents of the text field in response to button presses, as well as performing the appropriate operation when the suitable button is pressed.

The ATM software performs the following operations as follows:

- **Admin:** The only admin can add the card no they want and then the user can make his operations with respect to the software.

- **Pin change:** The user can change the pin by selecting this operation by entering the proper card no and old pin than he can change his pin.

- **Withdraw**: The user can withdraw the amount

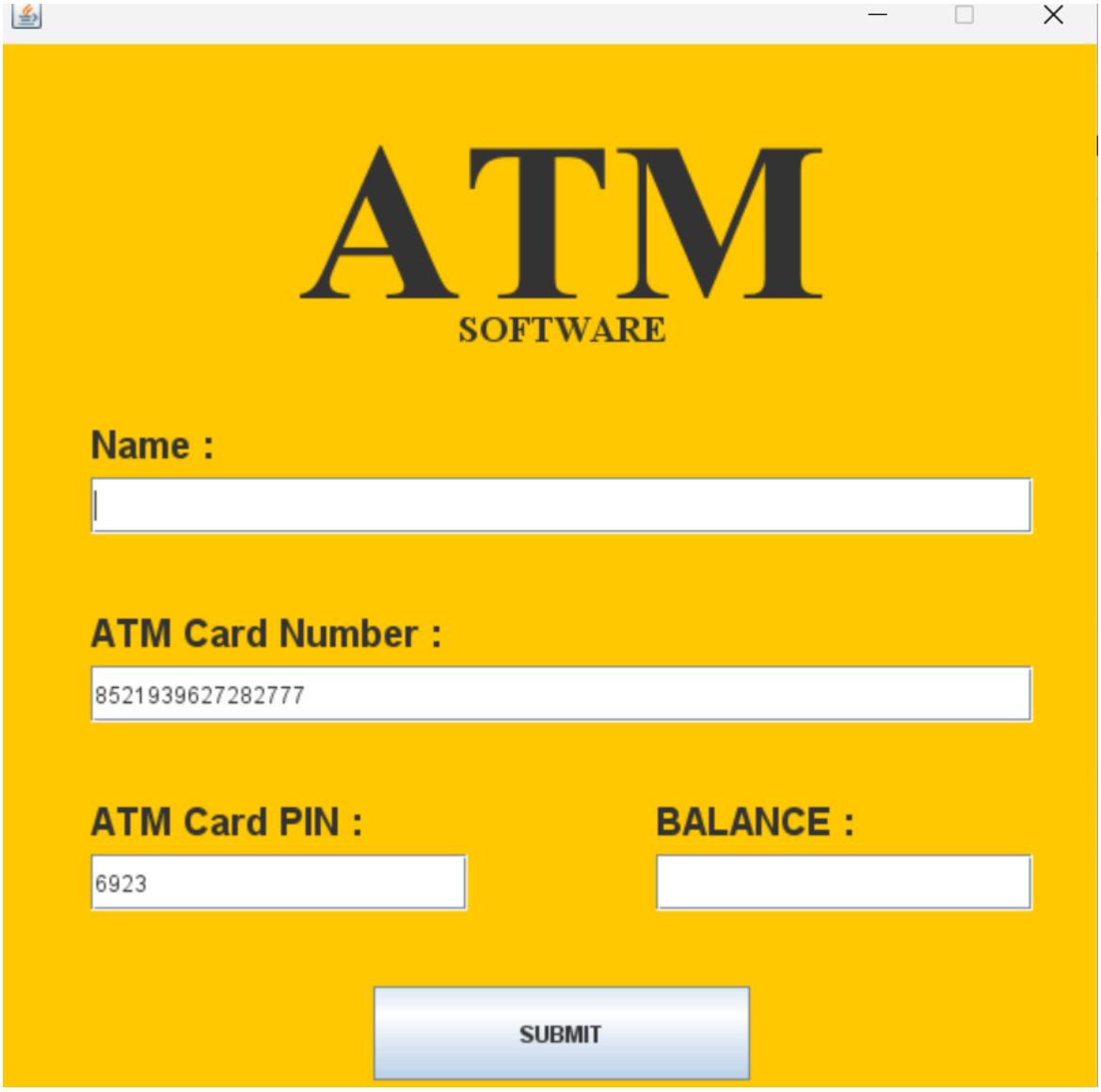
- **Deposit:** The user can deposit the amount to his bank account by using the card number.

- **Mini statement**: here the transaction history is fetched.

- **Balance Enquiry:** User can see balance of there account.

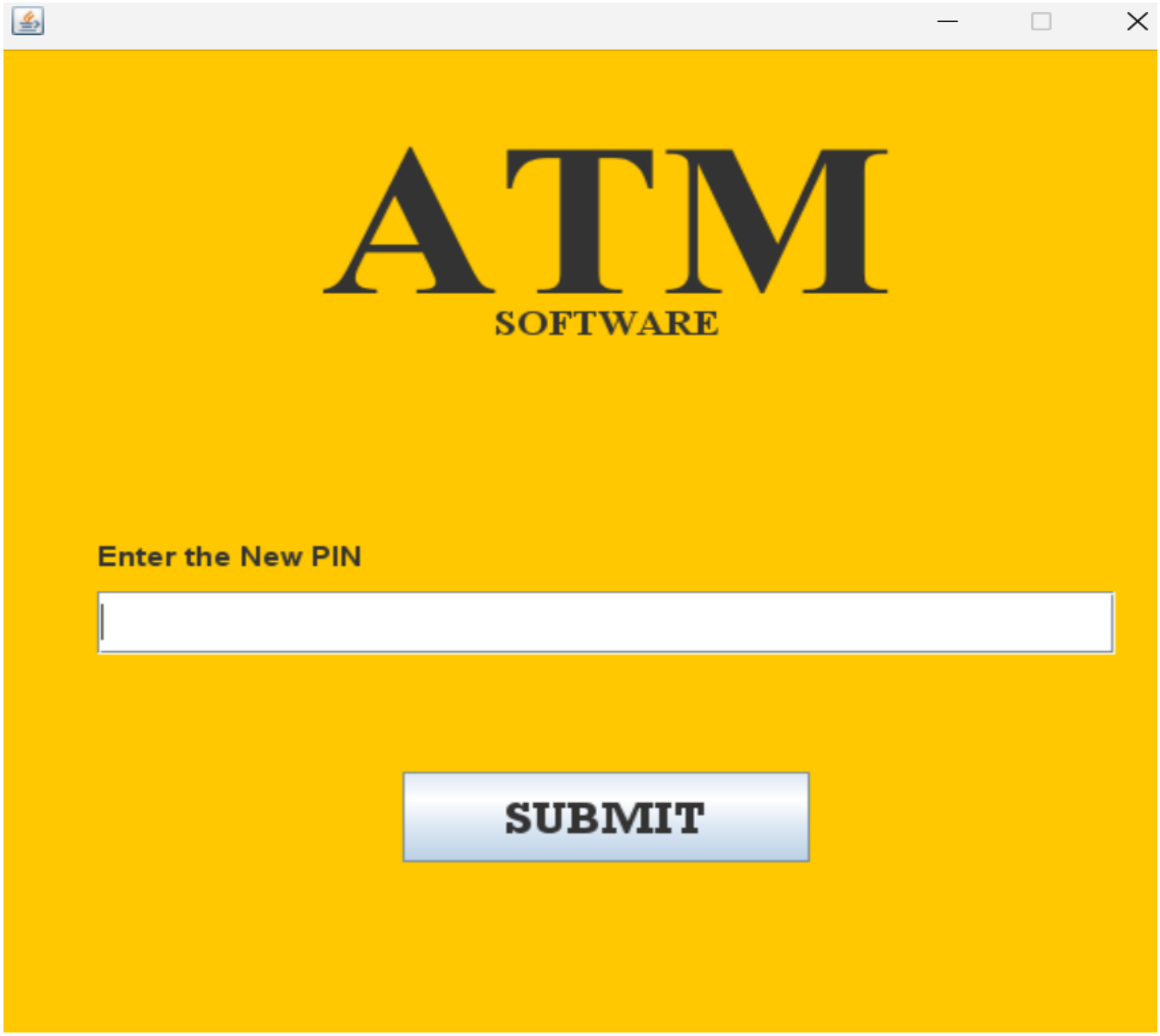
Admin:

Here the Admin button is to add the users details like card no, name, pin and etc.. then by clicking on continue button in the admin page then the details will be saved in database successfully.



Pin Change:

Here the user can change his pin according to the need.



Withdraw:

Here user can withdraw the amount how much he want.

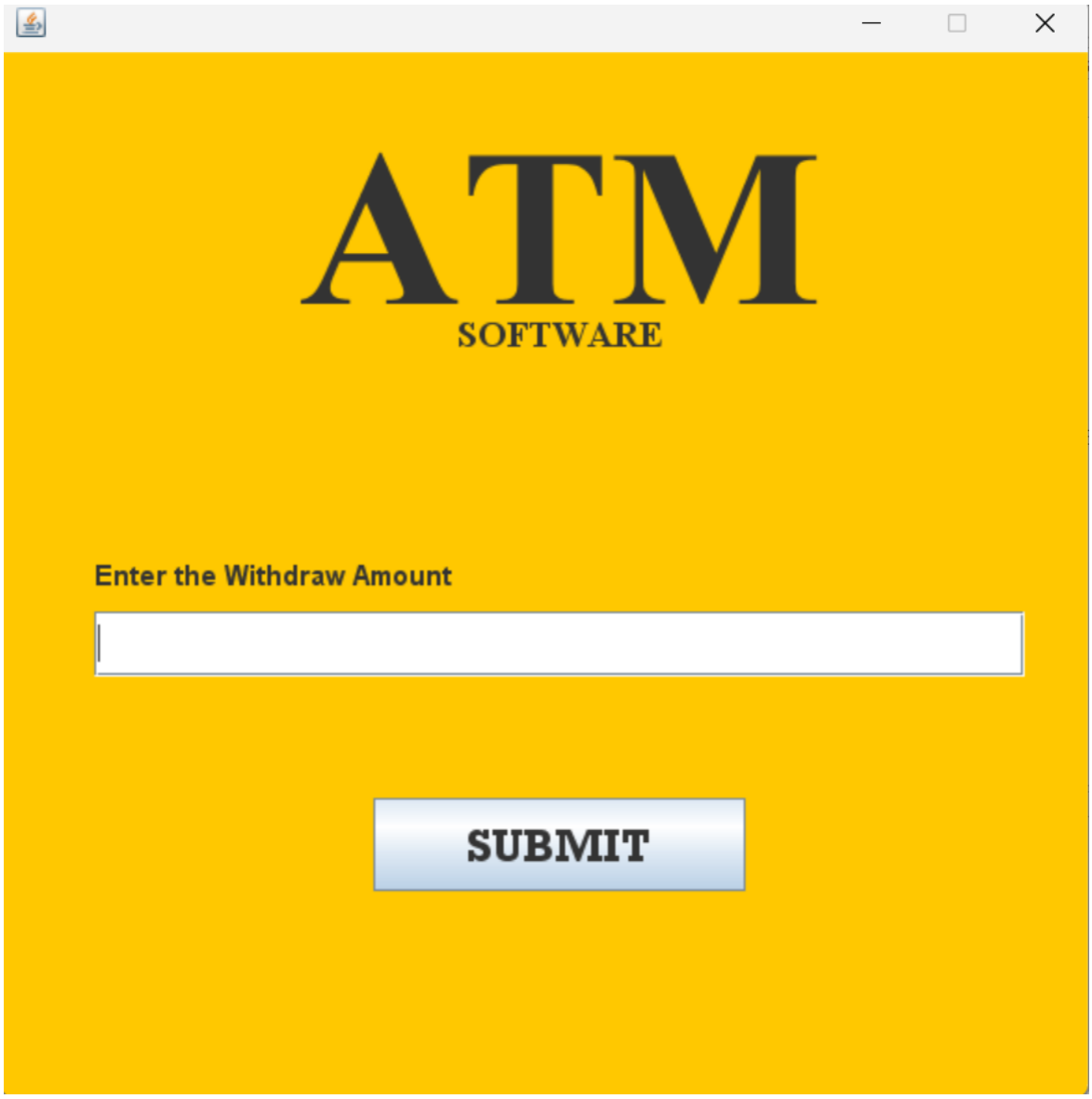


Fig: Withdraw

Deposit:

- Here the user can deposit the amount to his bank account by using the card number .

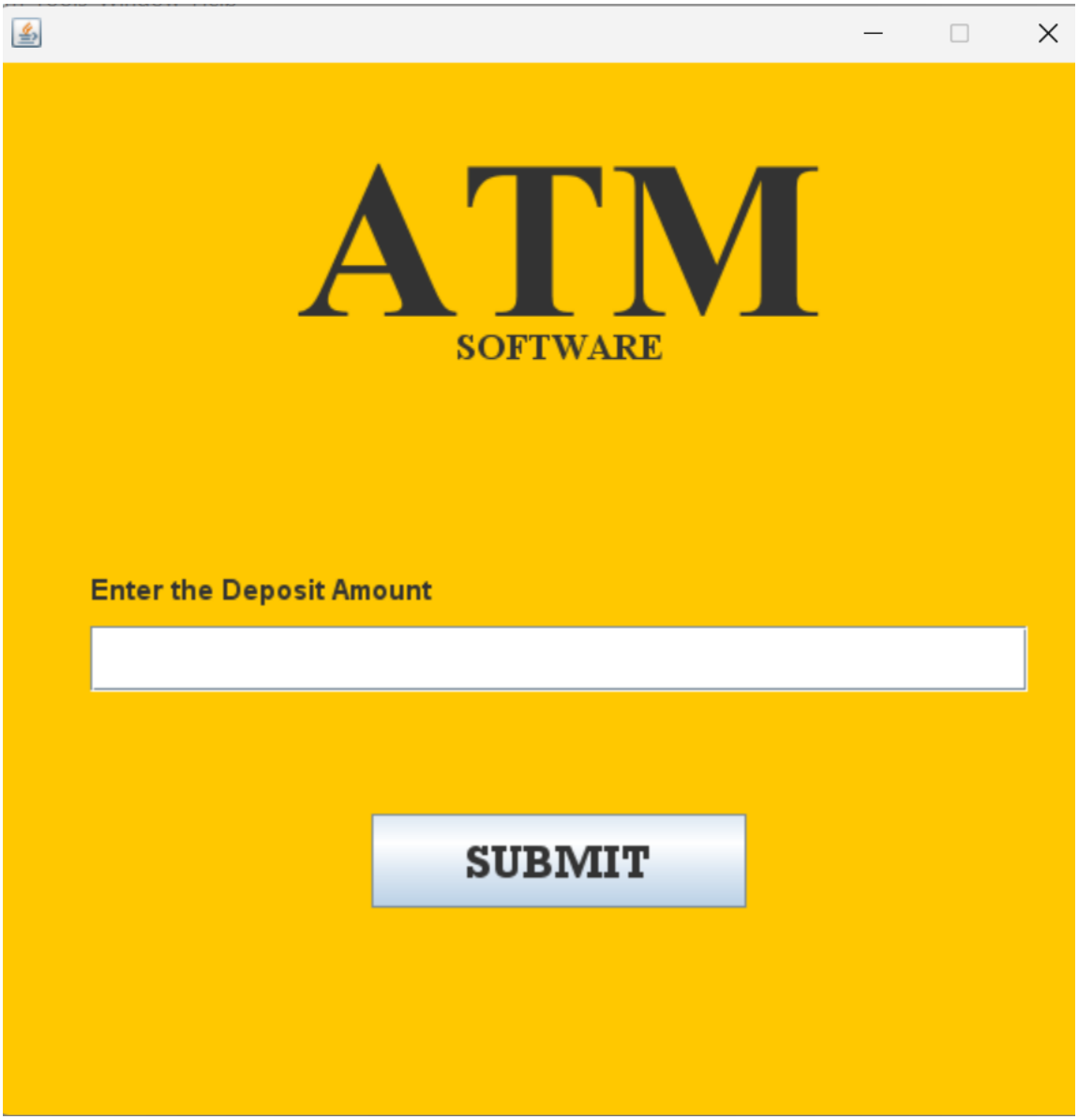


Fig:Deposit

-

- **Mini statement:** : here the transaction history is fetched.

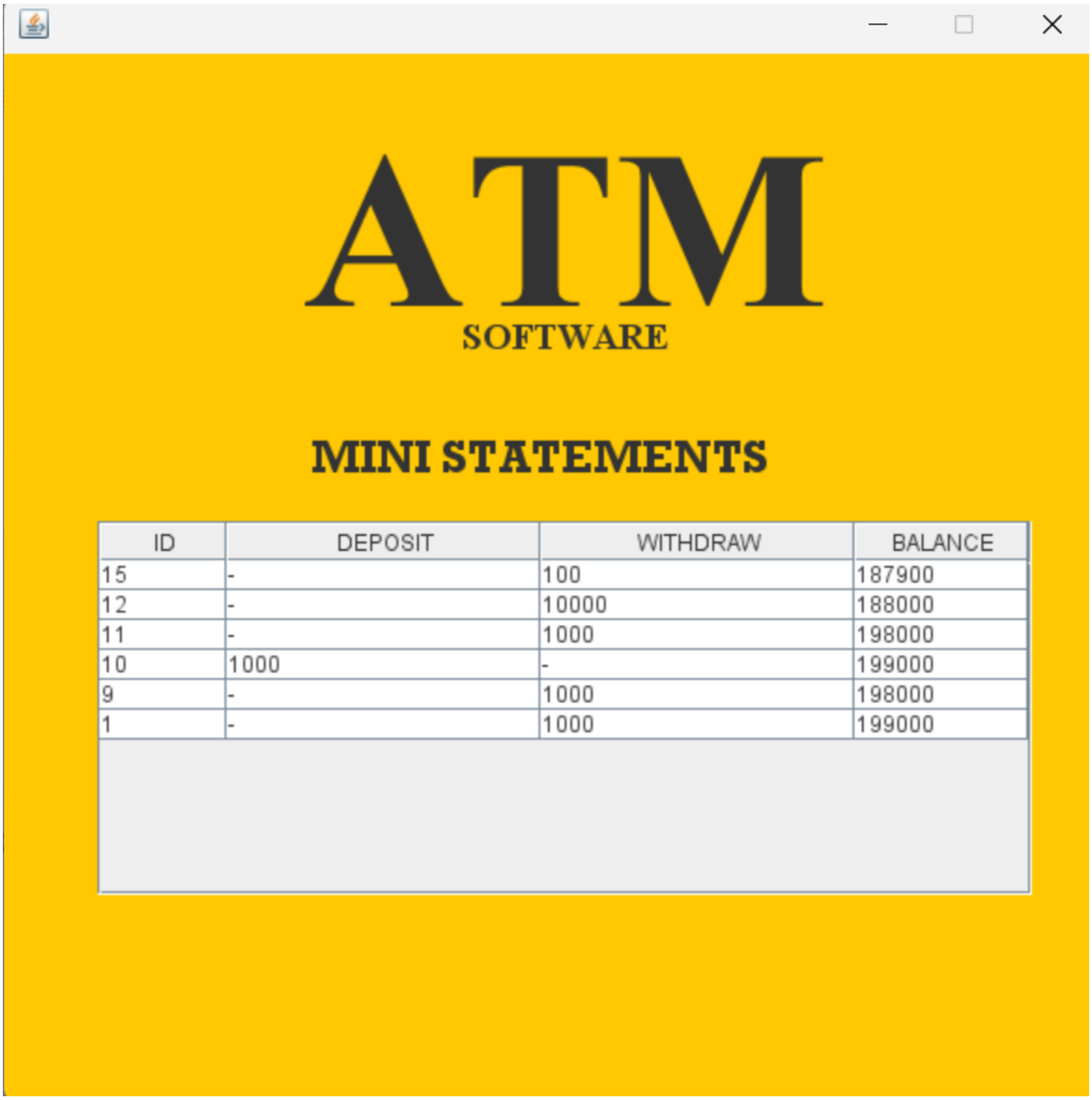
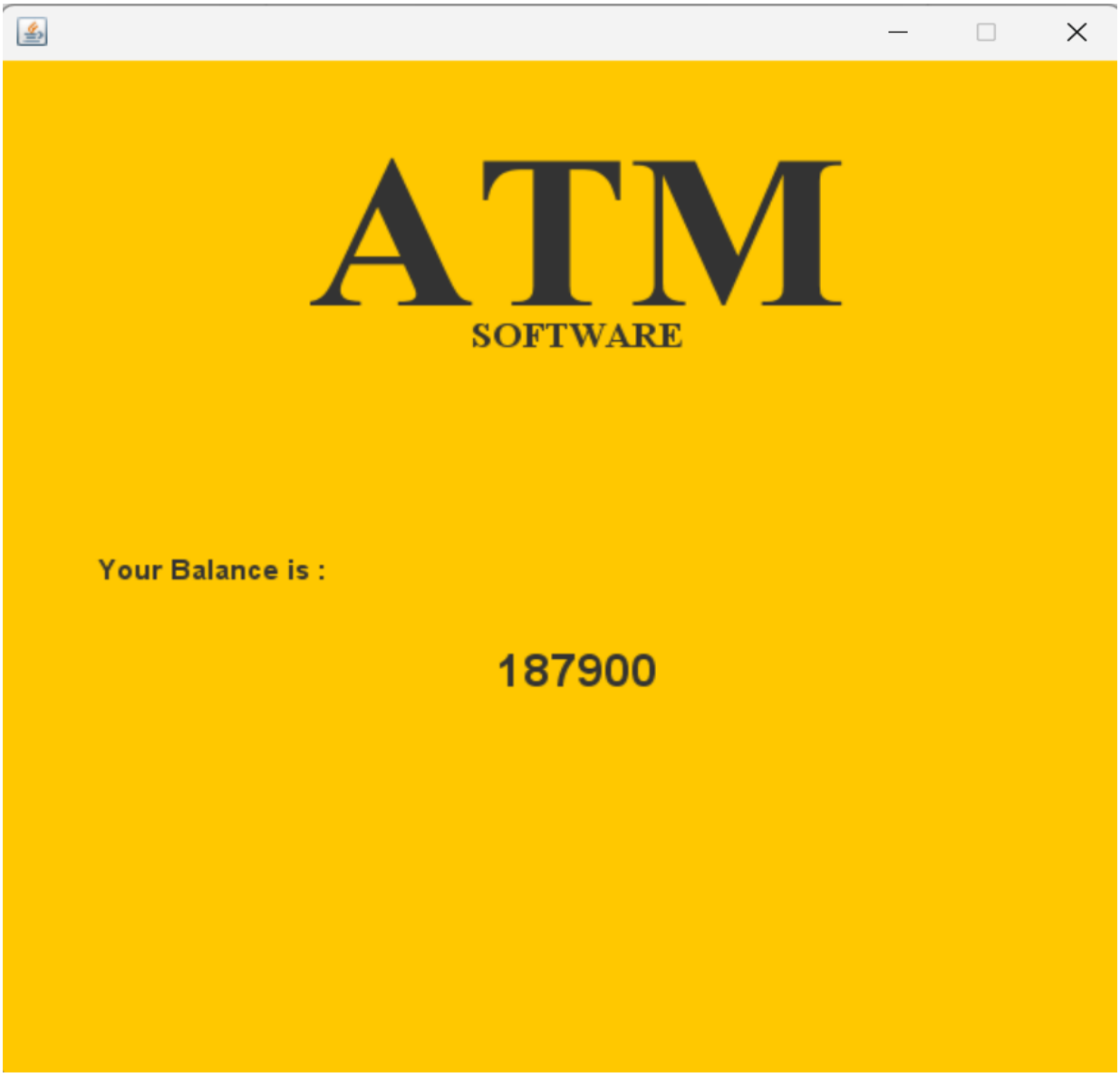


Fig: MiniStatement

Balance Enquiry:

Here User can see there balance of there account.



Chapter 5: CODE

Main.java:

```
package atmMain;

/**
 *
 * @author Vivek
 */

 import java.sql.SQLException;

public class Main1 {
                public static void main(String[] args) throws InterruptedException,
SQLException {



        Login login = new Login();
                login.loginView();


                }
}
```

Login.java

```
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JTextField;

public class Login extends Commons{
	public void loginView() {
		Commons common = new Commons();
		JFrame frame = (JFrame)common.Frame();
		Font txt = new Font("", Font.BOLD, 15);
		Pin pin = new Pin();

		//---------------CARDNUMBER----------------
		JLabel card = new JLabel("ENTER YOUR CARD NUMBER");
		card.setBounds(50, 270, 250, 20);
		card.setFont(txt);
		JTextField cardNumber = new JTextField();
		cardNumber.setBounds(50, 300, 500, 35);
		cardNumber.setFont(txt);
		frame.add(cardNumber);
		frame.add(card);
		//---------------------------------------

		//----------------ADMIN--------------------
		JLabel admin = new JLabel("ADMIN LOGIN >");
		admin.setBounds(0, 400, 570, 30);
		admin.setHorizontalAlignment(JLabel.RIGHT);
		admin.setFont(txt);
		frame.add(admin);
		admin.addMouseListener(new MouseAdapter() {
			public void mousePressed(MouseEvent e) {
				pin.pinView("admin");
				frame.dispose();
			}
		});
```

```
		//-----------------BUTTON-----------------
		JButton conts = new JButton("CONTINUE");
		conts.setBounds(50, 350, 200, 30);
		conts.setFont(new Font("Rockwell", Font.BOLD, 25));
		frame.add(conts);
		conts.addActionListener(new ActionListener() {
			@Override
			public void actionPerformed(ActionEvent e) {
				if(cardNumber.getText().length() == 16) {
					pin.pinView(cardNumber.getText());

				}
				else {
					Fail fail = new Fail();
					fail.failView("WRONG CARD NUMBER!!!");
					frame.dispose();
				}
			}

		});

		frame.setVisible(true);
	}
}
```

Admin.java :

```
import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.SQLException;
import java.util.logging.Level;
import java.util.logging.Logger;

import javax.swing.JButton;
import javax.swing.JFrame;

public class Admin {

	public void adminView() {
		Commons commons = new Commons();
```

```
        JFrame frame = new JFrame();
        frame.setLocationRelativeTo(null);
        frame.getContentPane().setBackground(Color.orange);
        frame.setLayout(null);
        frame.setSize(600, 600);

        //-------------ADDUSERS---------------------
        JButton add = new JButton("ADD USERS");
        add.setBounds(150, 250, 300, 50);
        add.setFont(new Font("Rockwell", Font.BOLD, 25));
        frame.add(add);
        add.addActionListener(new ActionListener() {
                @Override
                public void actionPerformed(ActionEvent e) {
                        AddUser user = new AddUser();
                        try {
                                user.addView();
                        } catch (SQLException e1) {
                                e1.printStackTrace();
                        } catch (ClassNotFoundException ex) {
                    Logger.getLogger(Admin.class.getName()).log(Level.SEVERE, null, ex);
                }
                        frame.dispose();
                }
        });
        //-----------------------------------------

        //--------------EXIT--------------------------
        JButton exit = new JButton("EXIT");
        exit.setBounds(150, 400, 300, 50);
        exit.setFont(new Font("Rockwell", Font.BOLD, 25));
        frame.add(exit);
        exit.addActionListener(new ActionListener() {
                @Override
                public void actionPerformed(ActionEvent e) {
                        System.exit(0);
                }
        });
        //-----------------------------------------
        frame.setVisible(true);


    }

}
```

Pin.java :

```
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.logging.Level;
import java.util.logging.Logger;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPasswordField;

public class Pin {
	public void pinView(String cardNum) {
		Commons common = new Commons();
		JFrame frame = (JFrame)common.Frame();
		Font txt = new Font("", Font.BOLD, 15);
		Home home = new Home();
		Admin admin = new Admin();

		//---------------PASSWORD----------------
		JLabel pswd = new JLabel("ENTER YOUR PIN");
		pswd.setBounds(50, 270, 250, 20);
		pswd.setFont(txt);
		JPasswordField pswdField = new JPasswordField();
		pswdField.setBounds(50, 300, 500, 35);
		pswdField.setFont(txt);
		frame.add(pswdField);
		frame.add(pswd);
		//---------------------------------------

		//-----------------BUTTON-----------------
		JButton conts = new JButton("COUNTINUE");
		conts.setBounds(200, 400, 200, 50);
		conts.setFont(new Font("Rockwell", Font.BOLD, 25));
		frame.add(conts);
		conts.addActionListener(new ActionListener() {
			@Override
public void actionPerformed(ActionEvent e) {
try {
SQLManage man = new SQLManage();
ResultSet rst = man.check(cardNum, pswdField.getText());
if(rst.next()) {
if(rst.getString("card").equals("admin")) {
```

```
admin.adminView();
frame.dispose();
}
else {
home.homeView(rst.getInt("id"));
frame.dispose();
}
}
else {
Fail fail = new Fail();
fail.failView("WRONG PIN!!!");
frame.dispose();
}
} catch (SQLException e1) {
e1.printStackTrace();
} catch (ClassNotFoundException ex) {
Logger.getLogger(Pin.class.getName()).log(Level.SEVERE, null, ex);
}
}

});
                //---------------------------------------
                frame.setVisible(true);
        }
}
```

AddUser.java :

```
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.SQLException;
import java.util.Random;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JTextField;

public class AddUser {

        JTextField pinField, atmField;
        Random random = new Random();

        public void addView() throws SQLException, ClassNotFoundException {
```

```
Commons commons = new Commons();
JFrame frame = (JFrame) commons.Frame();
Font txt = new Font("", Font.BOLD, 20);
SQLManage manage = new SQLManage();
Success success = new Success();

//-------------NAME--------------------
JLabel name = new JLabel("Name : ");
name.setBounds(50, 200, 100, 25);
name.setFont(txt);
JTextField nmField = new JTextField();
nmField.setBounds(50, 230, 500, 30);
frame.add(nmField);
frame.add(name);
//-------------------------------------

//-------------ATMNUMBER-----------------
JLabel atmno = new JLabel("ATM Card Number : ");
atmno.setBounds(50, 300, 500, 25);
atmno.setFont(txt);
atmField = new JTextField();
atmField.setBounds(50, 330, 500, 30);
atmField.setEditable(true);
frame.add(atmField);
frame.add(atmno);
//-------------------------------------

//-------------ATMPIN-----------------
JLabel atmpin = new JLabel("ATM Card PIN : ");
atmpin.setBounds(50, 400, 500, 25);
atmpin.setFont(txt);
pinField = new JTextField();
pinField.setBounds(50, 430, 200, 30);
pinField.setEditable(true);
frame.add(pinField);
frame.add(atmpin);
//-------------------------------------

//-------------BALANCE-----------------
JLabel bal = new JLabel("BALANCE : ");
bal.setBounds(350, 400, 500, 25);
bal.setFont(txt);
JTextField balField = new JTextField();
balField.setBounds(350, 430, 200, 30);
frame.add(balField);
frame.add(bal);
//-------------------------------------
```

```
		//--------------AUTOGENERATION----------------
		auto();
		//----------------------------------------

		//---------------SUBMIT-------------------
		JButton sbmt = new JButton("SUBMIT");
		sbmt.setBounds(200, 500, 200, 50);
		frame.add(sbmt);
		sbmt.addActionListener(new ActionListener() {
			@Override
			public void actionPerformed(ActionEvent e) {
				if(!nmField.getText().equals("")) {
					if(balField.getText().equals(""))
						balField.setText("0");
					try {
						manage.adding(atmField.getText(),
pinField.getText(), nmField.getText(), balField.getText());
					} catch (SQLException e1) {
						e1.printStackTrace();
					}
					success.detailView(atmField.getText(), pinField.getText());
					balField.setText("");
					nmField.setText("");
					auto();
				}
			}
		});
		//----------------------------------------

		frame.setVisible(true);
	}

	public void auto() {
		String str = "";
		for(int i=0; i<16; i++) {
			str += random.nextInt(9 - 0 + 1) + 0;
		}
		atmField.setText(str);
		str = "";
		for(int i=0; i<4; i++) {
			str += random.nextInt(9 - 0 + 1) + 0;
		}
		pinField.setText(str);
	}

}
```

Commons.java :

```
import java.awt.Color;
import java.awt.Component;
import java.awt.Font;

import javax.swing.JFrame;
import javax.swing.JLabel;

public class Commons {
	public Component Frame() {
		JFrame frame = new JFrame();
		frame.setSize(600, 600);
		frame.setLocationRelativeTo(null);
		frame.setLayout(null);
		frame.setResizable(false);
		frame.getContentPane().setBackground(Color.orange);

		//------------------LOGO---------------------------------
		JLabel atm = new JLabel("ATM");
		atm.setBounds(0, 30, 600, 120);
		atm.setHorizontalAlignment(JLabel.CENTER);
		atm.setFont(new Font("Serif", Font.BOLD, 120));
		JLabel man = new JLabel("SOFTWARE");
		man.setBounds(0, 140, 600, 20);
		man.setHorizontalAlignment(JLabel.CENTER);
		man.setFont(new Font("Serif", Font.BOLD, 20));
		frame.add(man);
		frame.add(atm);
		//-----------------------------------------------
		return frame;
	}

}
```

Fail.java

```
import java.awt.Font;

import javax.swing.JFrame;
import javax.swing.JLabel;

public class Fail {
	public void failView(String str) {
		Commons commons = new Commons();
		JFrame frame =(JFrame) commons.Frame();

		//-----------------FAIL------------------
		JLabel fail = new JLabel("YOUR TRANSACTIONS FAILED!!!");
		fail.setBounds(0, 280, 600, 50);
		fail.setHorizontalAlignment(JLabel.CENTER);
		fail.setFont(new Font("Rockwell", Font.BOLD, 25));
		JLabel st = new JLabel(str);
		st.setBounds(0, 320, 600, 50);
		st.setHorizontalAlignment(JLabel.CENTER);
		st.setFont(new Font("Rockwell", Font.BOLD, 25));
		frame.add(st);
		frame.add(fail);
		//----------------------------------------

		frame.setVisible(true);
	}
}
```

Home.java :

```
import java.awt.Font;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;
import java.sql.SQLException;
import java.util.logging.Level;
import java.util.logging.Logger;

import javax.swing.JFrame;
import javax.swing.JLabel;

public class Home {
	public void homeView(int id) throws SQLException, ClassNotFoundException {
		Operations operations = new Operations();
		Font txt = new Font("", Font.BOLD, 25);
		Commons commons = new Commons();
		JFrame frame = (JFrame)commons.Frame();
		JLabel quick = new JLabel("< Quick Cash");
		quick.setBounds(30, 250, 200, 30);
		quick.setFont(txt);
		JLabel withdraw = new JLabel("Withdraw >");
		withdraw.setBounds(350, 250, 200, 30);
		withdraw.setHorizontalAlignment(JLabel.RIGHT);
		withdraw.setFont(txt);
		JLabel deposit = new JLabel("< Deposit");
		deposit.setBounds(30, 350, 200, 30);
		deposit.setFont(txt);
		JLabel sts = new JLabel("Mini Statement >");
		sts.setBounds(350, 350, 200, 30);
		sts.setHorizontalAlignment(JLabel.RIGHT);
		sts.setFont(txt);
		JLabel bal = new JLabel("< Balance Enquiry");
		bal.setBounds(30, 450, 250, 30);
		bal.setFont(txt);
		JLabel pinchange = new JLabel("Change Pin >");
		pinchange.setBounds(350, 450, 200, 30);
		pinchange.setHorizontalAlignment(JLabel.RIGHT);
		pinchange.setFont(txt);
		frame.add(quick);
		frame.add(withdraw);
		frame.add(deposit);
		frame.add(sts);
		frame.add(bal);
```

```
frame.add(pinchange);
frame.setVisible(true);

quick.addMouseListener(new MouseAdapter() {
        public void mousePressed(MouseEvent e) {
                Quick qk = new Quick();
                try {
                        qk.quickView(id);
                } catch (SQLException e1) {
                        e1.printStackTrace();
                } catch (ClassNotFoundException ex) {
            Logger.getLogger(Home.class.getName()).log(Level.SEVERE, null, ex);
        }
                frame.dispose();
        }
});
withdraw.addMouseListener(new MouseAdapter() {
        public void mousePressed(MouseEvent e) {
                try {
                        operations.opView("Withdraw Amount", id);
                } catch (SQLException e1) {
                        e1.printStackTrace();
                }
                frame.dispose();
        }
});
deposit.addMouseListener(new MouseAdapter() {
        public void mousePressed(MouseEvent e) {
                try {
                        operations.opView("Deposit Amount", id);
                } catch (SQLException e1) {
                        e1.printStackTrace();
                }
                frame.dispose();
        }
});
sts.addMouseListener(new MouseAdapter() {
        public void mousePressed(MouseEvent e) {
                Statements state = new Statements();
                try {
                        state.stateView(id);
                } catch (SQLException e1) {
                        e1.printStackTrace();
                } catch (ClassNotFoundException ex) {
            Logger.getLogger(Home.class.getName()).log(Level.SEVERE, null, ex);
        }
                frame.dispose();
```

```
                }
            });
            bal.addMouseListener(new MouseAdapter() {
                public void mousePressed(MouseEvent e) {
                    try {
                        operations.opView("Balance", id);
                    } catch (SQLException e1) {
                        e1.printStackTrace();
                    }
                    frame.dispose();
                }
            });
            pinchange.addMouseListener(new MouseAdapter() {
                public void mousePressed(MouseEvent e) {
                    try {
                        operations.opView("New PIN", id);
                    } catch (SQLException e1) {
                        e1.printStackTrace();
                    }
                    frame.dispose();
                }
            });
        }


}
```

Operations.java:

```
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.SQLException;
import java.util.logging.Level;
import java.util.logging.Logger;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JTextField;

public class Operations {

    SQLManage manage;
```

```
	Fail fail;
	Success success;

	Operations() throws SQLException, ClassNotFoundException {
		manage = new SQLManage();
		fail = new Fail();
		success = new Success();
	}

	public void opView(String str, int id) throws SQLException {
		Commons commons = new Commons();
		JFrame frame = (JFrame)commons.Frame();
		Font txt = new Font("", Font.BOLD, 15);

		//-----------------AMOUNT/PIN------------------
		JLabel label = new JLabel("Enter the " + str);
		label.setBounds(50, 270, 250, 20);
		label.setFont(txt);
		JTextField amt = new JTextField();
		amt.setBounds(50, 300, 500, 35);
		amt.setFont(txt);
		frame.add(label);
		frame.add(amt);
		//-----------------------------------------

		//-----------------SUBMIT-----------------------
		JButton sbt = new JButton("SUBMIT");
		sbt.setBounds(200, 400, 200, 50);
		sbt.setFont(new Font("Rockwell", Font.BOLD, 25));
		frame.add(sbt);
		sbt.addActionListener(new ActionListener() {
			@Override
			public void actionPerformed(ActionEvent e) {
				if(str.equals("Withdraw Amount")) {
			   try {
			      withdrawal(Integer.parseInt(amt.getText()), id);
			   } catch (ClassNotFoundException ex) {
			      Logger.getLogger(Operations.class.getName()).log(Level.SEVERE,
null, ex);
			   }
					frame.dispose();
				}
				else if(str.equals("Deposit Amount")) {
					try {
						manage.deposit(Integer.parseInt(amt.getText()), id);
						success.successView(id);
						frame.dispose();
```

```
                                        }
                                        catch (SQLException e1) {
                                                e1.printStackTrace();
                                        } catch (ClassNotFoundException ex) {
                                    Logger.getLogger(Operations.class.getName()).log(Level.SEVERE,
null, ex);
                                }
                                    }
                                    else if(str.equals("New PIN")){
                                        try {
                                                manage.pinchange(amt.getText(), id);
                                                success.successView(id);
                                                frame.dispose();
                                        }
                                        catch (SQLException e1) {
                                                e1.printStackTrace();
                                        } catch (ClassNotFoundException ex) {
                                    Logger.getLogger(Operations.class.getName()).log(Level.SEVERE,
null, ex);
                                }
                                    }
                                }

                        });
                        //-----------------------------------------------

                        if (str.equals("Balance")){
                                amt.setVisible(false);
                                sbt.setVisible(false);
                                label.setText("Your Balance is : ");
                                JLabel bal;
                                try {
                                        bal = new JLabel(manage.balCheck(id)+"");
                                        bal.setBounds(0, 325, 600, 20);
                                        bal.setHorizontalAlignment(JLabel.CENTER);
                                        bal.setFont(new Font("", Font.BOLD, 25));
                                        frame.add(bal);
                                } catch (SQLException e1) {
                                        e1.printStackTrace();
                                }
                        }

                        frame.setVisible(true);
                }

                public void withdrawal(int amount, int id) throws ClassNotFoundException {
                        try {
```

```
                int check = manage.withdraw(amount, id);
                if(check==1) {
                        success.successView(id);
                }
                else {
                        fail.failView("INSUFFICIENT BALANCE!!!");
                }
            }
            catch (SQLException e1) {
                e1.printStackTrace();
            }
        }


}
```

Quick.java

```
import java.awt.Font;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;
import java.sql.SQLException;
import java.util.logging.Level;
import java.util.logging.Logger;

import javax.swing.JFrame;
import javax.swing.JLabel;

public class Quick {
        public void quickView(int id) throws SQLException, ClassNotFoundException {

                Operations oper = new Operations();

                Font txt = new Font("", Font.BOLD, 25);
                Commons commons = new Commons();
                JFrame frame = (JFrame)commons.Frame();
                JLabel two = new JLabel("< 200");
                two.setBounds(30, 250, 200, 30);
                two.setFont(txt);
                JLabel five = new JLabel("500 >");
                five.setBounds(350, 250, 200, 30);
                five.setHorizontalAlignment(JLabel.RIGHT);
```

```
five.setFont(txt);
JLabel ten = new JLabel("< 1000");
ten.setBounds(30, 350, 200, 30);
ten.setFont(txt);
JLabel twenty = new JLabel("2000 >");
twenty.setBounds(350, 350, 200, 30);
twenty.setHorizontalAlignment(JLabel.RIGHT);
twenty.setFont(txt);
JLabel fifty = new JLabel("< 5000");
fifty.setBounds(30, 450, 250, 30);
fifty.setFont(txt);
JLabel hundred = new JLabel("10000 >");
hundred.setBounds(350, 450, 200, 30);
hundred.setHorizontalAlignment(JLabel.RIGHT);
hundred.setFont(txt);
frame.add(two);
frame.add(five);
frame.add(ten);
frame.add(twenty);
frame.add(fifty);
frame.add(hundred);
frame.setVisible(true);

two.addMouseListener(new MouseAdapter() {
        public void mousePressed(MouseEvent e) {
    try {
       oper.withdrawal(200, id);
    } catch (ClassNotFoundException ex) {
       Logger.getLogger(Quick.class.getName()).log(Level.SEVERE, null, ex);
    }
                frame.dispose();
       }
});
five.addMouseListener(new MouseAdapter() {
        public void mousePressed(MouseEvent e) {
    try {
       oper.withdrawal(500, id);
       frame.dispose();
    } catch (ClassNotFoundException ex) {
       Logger.getLogger(Quick.class.getName()).log(Level.SEVERE, null, ex);
    }
       }
});
ten.addMouseListener(new MouseAdapter() {
        public void mousePressed(MouseEvent e) {
    try {
       oper.withdrawal(1000, id);
```

```
                frame.dispose();
            } catch (ClassNotFoundException ex) {
                Logger.getLogger(Quick.class.getName()).log(Level.SEVERE, null, ex);
            }
                }
        });
        twenty.addMouseListener(new MouseAdapter() {
                public void mousePressed(MouseEvent e) {
            try {
                oper.withdrawal(2000, id);
                frame.dispose();
            } catch (ClassNotFoundException ex) {
                Logger.getLogger(Quick.class.getName()).log(Level.SEVERE, null, ex);
            }
                }
        });
        fifty.addMouseListener(new MouseAdapter() {
                public void mousePressed(MouseEvent e) {
            try {
                oper.withdrawal(5000, id);
                frame.dispose();
            } catch (ClassNotFoundException ex) {
                Logger.getLogger(Quick.class.getName()).log(Level.SEVERE, null, ex);
            }
                }
        });
        hundred.addMouseListener(new MouseAdapter() {
                public void mousePressed(MouseEvent e) {
            try {
                oper.withdrawal(10000, id);
                frame.dispose();
            } catch (ClassNotFoundException ex) {
                Logger.getLogger(Quick.class.getName()).log(Level.SEVERE, null, ex);
            }
                }
        });
    }


}
```

SQLmanage.java

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class SQLManage {
	Connection con;

	SQLManage() throws SQLException, ClassNotFoundException {
	 Class.forName("com.mysql.jdbc.Driver");
		String usr = "root";
		String pass = "root";
		String url = "jdbc:mysql://localhost:3306/atm";
		con = DriverManager.getConnection(url, usr, pass);
	}

	public ResultSet check(String usr, String pass) throws SQLException {
		String str = "SELECT * FROM users WHERE card = '"+ usr +"' or pin = '"+ pass+
"'";
		Statement stm = con.createStatement();
		ResultSet rst = stm.executeQuery(str);
		return rst;
	}

	public void deposit(int amt, int id) throws SQLException {
		String str = "UPDATE users SET bal = bal + "+amt+" WHERE id = "+id;
		Statement stm = con.createStatement();
		stm.executeUpdate(str);
		int bal = balCheck(id);
		str = "INSERT INTO transactions (id, amount, stat, bal) VALUES("+id+", "+amt+",
'dep', "+bal+")";
		Statement stm2 = con.createStatement();
		stm2.executeUpdate(str);
	}

	public int withdraw(int amt, int id) throws SQLException {
		int bal = balCheck(id);
		if(bal >= amt) {
			String str = "UPDATE users SET bal = bal - "+amt+" WHERE id = "+id;
			Statement stm = con.createStatement();
			stm.executeUpdate(str);
```

```
                        bal -= amt;
                        str = "INSERT INTO transactions (id, amount, stat, bal) VALUES("+id+",
"+amt+", 'wit', "+bal+")";
                        Statement stm2 = con.createStatement();
                        stm2.executeUpdate(str);
                        return 1;
                }
                return 0;
        }

        public void pinchange(String pin, int id) throws SQLException {
                String str = "UPDATE users SET pin = '"+pin+"' WHERE id = " + id;
                Statement stm = con.createStatement();
                stm.executeUpdate(str);
        }

        public int balCheck(int id) throws SQLException {
                String str = "SELECT bal FROM users WHERE id = " + id;
                Statement stm = con.createStatement();
                ResultSet rst = stm.executeQuery(str);
                rst.next();
                return rst.getInt("bal");
        }

        public ResultSet stmt(int id) throws SQLException {
                String str = "SELECT * FROM transactions WHERE id = " + id + " order by
transid desc";
                Statement stm = con.createStatement();
                ResultSet rst = stm.executeQuery(str);
                return rst;
        }

        public void adding(String card, String pin, String name, String bal) throws SQLException
{
                String str = "INSERT INTO users (card, pin, uname, bal) values ('" +card+ "',
'"+pin+"', '"+name+"', "+bal+")";
                Statement stm = con.createStatement();
                stm.executeUpdate(str);
        }
}
```

Statements.java:

```
import java.awt.Font;
import java.sql.ResultSet;
import java.sql.SQLException;

import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JScrollPane;
import javax.swing.JTable;
import javax.swing.table.DefaultTableModel;


public class Statements {
	public void stateView(int id) throws SQLException, ClassNotFoundException {
		DefaultTableModel model = new DefaultTableModel();
		Commons commons = new Commons();
		JFrame frame = (JFrame)commons.Frame();
		SQLManage manage = new SQLManage();

		//----------------LABEL-----------------------
		JLabel label = new JLabel("MINI STATEMENTS");
		label.setBounds(0, 200, 575, 30);
		label.setHorizontalAlignment(JLabel.CENTER);
		label.setFont(new Font("Rockwell", Font.BOLD, 25));
		frame.add(label);
		//----------------------------------------

		//---------------TABLE--------------------
		JTable table=new JTable(){
			public boolean isCellEditable(int row,int column){
					return false;
			}
		};
		model = (DefaultTableModel)table.getModel();
		model.addColumn("ID");
		model.addColumn("DEPOSIT");
		model.addColumn("WITHDRAW");
		model.addColumn("BALANCE");
		table.getColumnModel().getColumn(0).setPreferredWidth(50);
		table.getColumnModel().getColumn(1).setPreferredWidth(150);
		table.getColumnModel().getColumn(2).setPreferredWidth(150);
		table.getColumnModel().getColumn(2).setPreferredWidth(150);
		JScrollPane sc = new JScrollPane(table);
```

```
		sc.setBounds(50, 250, 500, 200);
		frame.add(sc);
		//-----------------------------------------------

		//--------------------TABLEDATA-----------------------
		ResultSet rst = manage.stmt(id);
		int i=0;
		while(rst.next()) {
			model.addRow(new Object[0]);
			model.setValueAt(rst.getInt("transid"), i, 0);
			if(rst.getString("stat").equals("dep")) {
				model.setValueAt(rst.getString("amount"), i, 1);
				model.setValueAt("-", i, 2);
			}
			else {
				model.setValueAt("-", i, 1);
				model.setValueAt(rst.getString("amount"), i, 2);
			}
		model.setValueAt(rst.getInt("bal"), i, 3);
		i++;
		  }
		  //-------------------------------------------------

		  frame.setVisible(true);
	}
}
```

Success.java

```
import java.awt.Font;
import java.sql.SQLException;

import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JTextField;

public class Success {
	public void successView(int id) throws SQLException, ClassNotFoundException {
		Home home = new Home();
		Commons commons = new Commons();
		JFrame frame =(JFrame) commons.Frame();
```

```
            //-----------------SUCCESS------------------
            JLabel sucss = new JLabel("TRANSACTION SUCCESS.");
            sucss.setBounds(0, 280, 600, 50);
            sucss.setHorizontalAlignment(JLabel.CENTER);
            sucss.setFont(new Font("Rockwell", Font.BOLD, 25));
            frame.add(sucss);
            //----------------------------------------
            home.homeView(id);
            frame.setVisible(true);
        }

        public void detailView(String num, String pin) {
            Commons commons = new Commons();
            JFrame frame =(JFrame) commons.Frame();

            //-----------------DETAILS------------------
            JLabel sucss = new JLabel("REMEMBER THE DETAILS!!!");
            sucss.setBounds(0, 200, 600, 50);
            sucss.setHorizontalAlignment(JLabel.CENTER);
            sucss.setFont(new Font("Rockwell", Font.BOLD, 25));
            frame.add(sucss);
            JTextField number = new JTextField("CARD NUMBER : " + num);
            number.setBounds(0, 300, 600, 50);
            number.setEditable(false);
            number.setHorizontalAlignment(JLabel.CENTER);
            number.setFont(new Font("Rockwell", Font.BOLD, 20));
            frame.add(number);
            JTextField pinno = new JTextField("DEFAULT PIN : " + pin);
            pinno.setBounds(0, 400, 600, 50);
            pinno.setHorizontalAlignment(JLabel.CENTER);
            pinno.setEditable(false);
            pinno.setFont(new Font("Rockwell", Font.BOLD, 20));
            frame.add(pinno);
            //----------------------------------------
            frame.setVisible(true);
        }
}
```

Chapter 6: FUTURE SCOPE

The Simple ATM software can be further enhanced by adding more advanced features & functions, such as e-withdraw etc. The application can also be made more user-friendly by adding error handling for invalid input. It can also be developed for Other banking machines platforms such as deposit and checker machines and by giving to the additional things likes face detection money withdraw and more features.

Chapter 7: LIMITATIONS

The Simple ATM Software is designed to perform Banking operations, such as, it is not intended to replace more advanced API based delveloped apps. The developed application is limited to performing just deposite,withdraw and some other applications and does not support the online trasaction methods. Additionally, the application does provide the history or memory functions, for users to keep track of their transactions.

Chapter 8: CONCLUSION

So in this project, we have learned to design a simple ATM Management System Project in Java with a database using MySQL. We can deposit, withdraw, change pin, and view statements and balances in this system. Adding user access is given to the admin.