A Project Report on

"BANK MANAGEMENT SYSTEM "

Submitted in partial fulfillment of the requirement for
Degree in Bachelor of Engineering (Information Technology)

By

Shivam Bhosale (12)
Parth Dali (19)
Pranav Dalvi (20)

Guided by:

Dr. Vaishali.P. Jadhav

Department of Information Technology
St. Francis Institute of Technology
Mount Poinsur, S.V.P. Road, Borivali (West), Mumbai 400 103
University of Mumbai
2019-2020

CERTIFICATE

This is to certify that the project entitled

“BANK MANAGEMENT SYSTEM”

Submitted By

Shivam Bhosale
Parth Dali
Pranav Dalvi

In partial fulfillment of degree of **B.E**. in **Information Technology** for term work of the project is approved.

External Examiner **Internal Examiner**

_____ _____

External Guide **Internal Guide**

_____ _____

Head of the Department **Principal**

_____ _____

Date: - **College Seal**

Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Signature)

(Shivam Bhosale, 12)

(Parth Dali, 19)

(Pranav Dalvi, 20)

Date:

ABSTRACT

The Bank Account Management System is an application for maintaining a person's account in a bank. In this project we tried to show the working of a banking account system and cover the basic functionality of a Bank Account Management System. To develop a project for solving financial applications of a customer in banking environment in order to nurture the needs of an end banking user by providing various ways to perform banking tasks. Also, to enable the user's work space to have additional functionalities which are not provided under a conventional banking project. The Bank Account Management System undertaken as a project is based on relevant technologies. The main aim of this project is to develop software for Bank Account Management System. This project has been developed to carry out the processes easily and quickly, which is not possible with the manuals systems, which are overcome by this software. This project is developed using JAVA language and MYSQL use for database connection. The system design is then implemented with MYSQL, JAVA forms, and JDBC. The system is designed as an interactive and content management system. The content management system deals with data entry, validation confirm and updating whiles the interactive system deals with system interaction with the administration and users. Thus, above features of this project will save transaction time and therefore increase the efficiency of the system.

INDEX

LIST OF FIGURES

Chapter 1

Introduction

- Bank is the place where customers feel the sense of safety for their property. In the bank, customers deposit and withdraw their money.
- Transaction of money also is a part where customer takes shelter of the bank. Now to keep the belief and trust of customers, there is the positive need for management of the bank, which can handle all this with comfort and ease. Smooth and efficient management affects the satisfaction of the customers and staff members, indirectly.
- And of course, it encourages management committee in taking some needed decision for future enhancement of the bank. Now a days, managing a bank is tedious job upto certain limit. So a bank management system that reduces the work is essential. Also today's world is a genuine computer world and is getting faster and faster day-by-day.
- Thus, considering above necessities, the system for bank management has became necessary which would be useful in managing the bank more efficiently.

Chapter 2

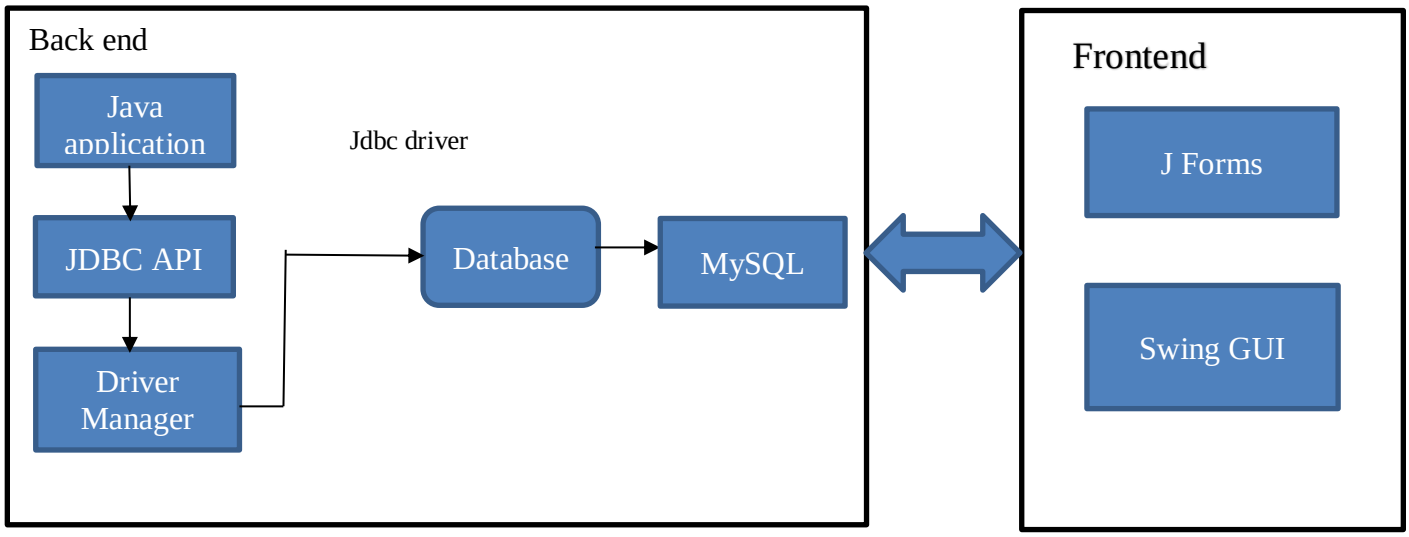
Problem Statement

- The Bank Management system consists of the bank administrator and the customer. The administrator will need to create a new account for the customer by logging into their existing account.
- For creating a new account on the system the administrator will need to enter details of customer like first name, last name, address, contact no and will add the branch where the account has been created.
- The system executes operations like creating new account, checking the balance of the account, withdrawing, depositing, money transfers between two accounts and viewing transaction history.

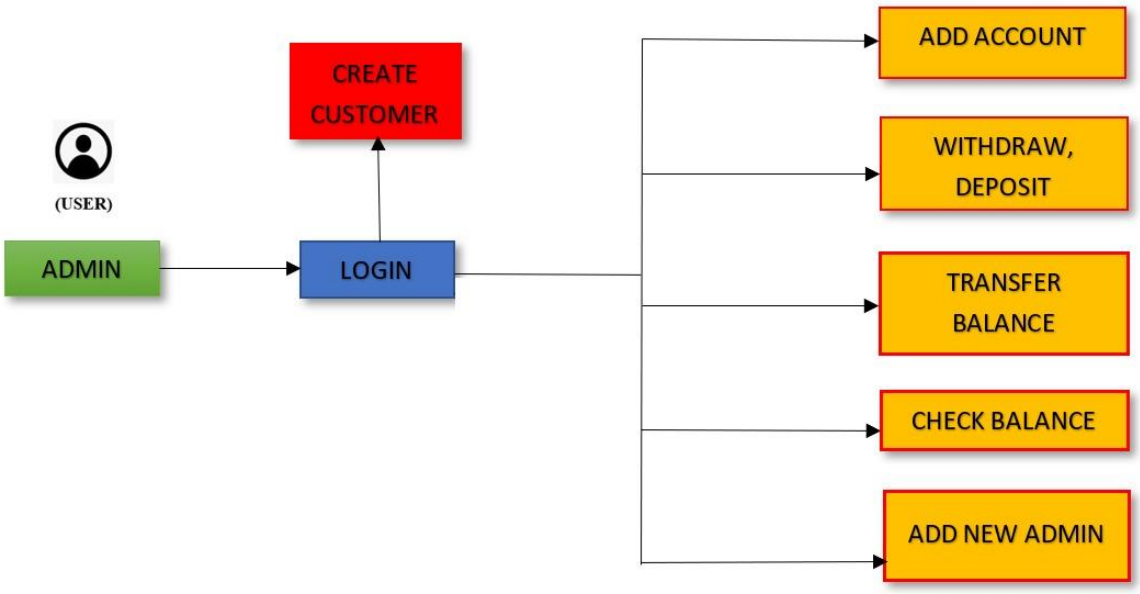
Chapter 3

System Design

3.1 Architectural Diagram/ block diagram



3.2 Flow chart



System Requirements:

Developer Requirements:

- Interpreters: Java and Java Database Connectivity(JDBC).
- Software: Netbeans, Xampp.
- Databases: MySQL.
- Hardware: 10GB storage, 4gb ram 4.1.2

Chapter 4

Source code

Main menu:

```
package bank;

import java.awt.Desktop;
import java.awt.PopupMenu;
import static java.awt.SystemColor.desktop;
import javax.swing.JDesktopPane;
import javax.swing.JFrame;
import javax.swing.JInternalFrame;
/**
 *
 * @author parth
 */
public class mainmenu extends javax.swing.JFrame {

    private PopupMenu internalFrame;


    public mainmenu() {
        initComponents();
    }

    /**
     * This method is called from within the constructor to initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jMenuItem1 = new javax.swing.JMenuItem();
        jDesktopPane1 = new javax.swing.JDesktopPane();
        jMenuBar1 = new javax.swing.JMenuBar();
        jMenu1 = new javax.swing.JMenu();
        jMenuItem2 = new javax.swing.JMenuItem();
        jMenuItem3 = new javax.swing.JMenuItem();
        jMenu2 = new javax.swing.JMenu();
```

```
        jMenuItem4 = new javax.swing.JMenuItem();
        jMenuItem5 = new javax.swing.JMenuItem();
        jMenu3 = new javax.swing.JMenu();
        jMenuItem6 = new javax.swing.JMenuItem();
        jMenu4 = new javax.swing.JMenu();
        jMenuItem7 = new javax.swing.JMenuItem();
        jMenu5 = new javax.swing.JMenu();
        jMenuItem8 = new javax.swing.JMenuItem();
        jMenu6 = new javax.swing.JMenu();
        jMenuItem9 = new javax.swing.JMenuItem();

        jMenuItem1.setText("jMenuItem1");

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        setPreferredSize(new java.awt.Dimension(1366, 768));

        javax.swing.GroupLayout jDesktopPane1Layout = new javax.swing.GroupLayout(jDesktopPane1);
        jDesktopPane1.setLayout(jDesktopPane1Layout);
        jDesktopPane1Layout.setHorizontalGroup(
            jDesktopPane1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGap(0, 823, Short.MAX_VALUE)
        );
        jDesktopPane1Layout.setVerticalGroup(
            jDesktopPane1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGap(0, 764, Short.MAX_VALUE)
        );

        jMenu1.setText("File");


jMenuItem2.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK_BACK_
SPACE, 0));
        jMenuItem2.setText("Customer");
        jMenuItem2.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jMenuItem2ActionPerformed(evt);
            }
        });
        jMenu1.add(jMenuItem2);

        jMenuItem3.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK_B,
java.awt.event.InputEvent.SHIFT_DOWN_MASK));
        jMenuItem3.setText("Account");
        jMenuItem3.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jMenuItem3ActionPerformed(evt);
            }
        });
        jMenu1.add(jMenuItem3);
```

```
jMenuBar1.add(jMenu1);

jMenu2.setText("Transaction");

jMenuItem4.setText("Withdraw");
jMenuItem4.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem4ActionPerformed(evt);
    }
});
jMenu2.add(jMenuItem4);

jMenuItem5.setText("Deposit");
jMenuItem5.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem5ActionPerformed(evt);
    }
});
jMenu2.add(jMenuItem5);

jMenuBar1.add(jMenu2);

jMenu3.setText("Transfer");
jMenu3.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenu3ActionPerformed(evt);
    }
});

jMenuItem6.setText("AccountToAccount");
jMenuItem6.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem6ActionPerformed(evt);
    }
});
jMenu3.add(jMenuItem6);

jMenuBar1.add(jMenu3);

jMenu4.setText("Report");

jMenuItem7.setText("Customer Report");
jMenuItem7.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jMenuItem7ActionPerformed(evt);
    }
});
jMenu4.add(jMenuItem7);

jMenuBar1.add(jMenu4);
```

```
    jMenu5.setText("Balance");

    jMenuItem8.setText("Balance Check");
    jMenuItem8.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jMenuItem8ActionPerformed(evt);
        }
    });
    jMenu5.add(jMenuItem8);

    jMenuBar1.add(jMenu5);

    jMenu6.setText("Admin");

    jMenuItem9.setText("Account Creation");
    jMenuItem9.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            jMenuItem9ActionPerformed(evt);
        }
    });
    jMenu6.add(jMenuItem9);

    jMenuBar1.add(jMenu6);

    setJMenuBar(jMenuBar1);

    javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jDesktopPane1)
            .addGap(69, 69, 69))
    );
    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jDesktopPane1)
            .addContainerGap())
    );

    pack();
}// </editor-fold>

private void jMenuItem2ActionPerformed(java.awt.event.ActionEvent evt) {

  customer cus = new customer();
```

```
  jDesktopPane1.add(cus);
  cus.setVisible(true);

}

private void jMenuItem3ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:


    account cus = new account();
  jDesktopPane1.add(cus);
  cus.setVisible(true);


}

private void jMenuItem4ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:

    withdraw cus = new withdraw();
  jDesktopPane1.add(cus);
  cus.setVisible(true);

}

private void jMenuItem5ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:

     deposit cus = new deposit();
  jDesktopPane1.add(cus);
  cus.setVisible(true);

}

private void jMenu3ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:




}

private void jMenuItem6ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
     transfer cus1 = new transfer();
         jDesktopPane1.add(cus1);
         cus1.setVisible(true);

}
```

```
    private void jMenuItem7ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        cusreport cus1 = new cusreport();
            jDesktopPane1.add(cus1);
            cus1.setVisible(true);



    }

    private void jMenuItem8ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:



         balance cus1 = new balance();
            jDesktopPane1.add(cus1);
            cus1.setVisible(true);

    }

    private void jMenuItem9ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:


             user1 cus1 = new user1();
            jDesktopPane1.add(cus1);
            cus1.setVisible(true);




    }

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
         * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {
                    javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
```

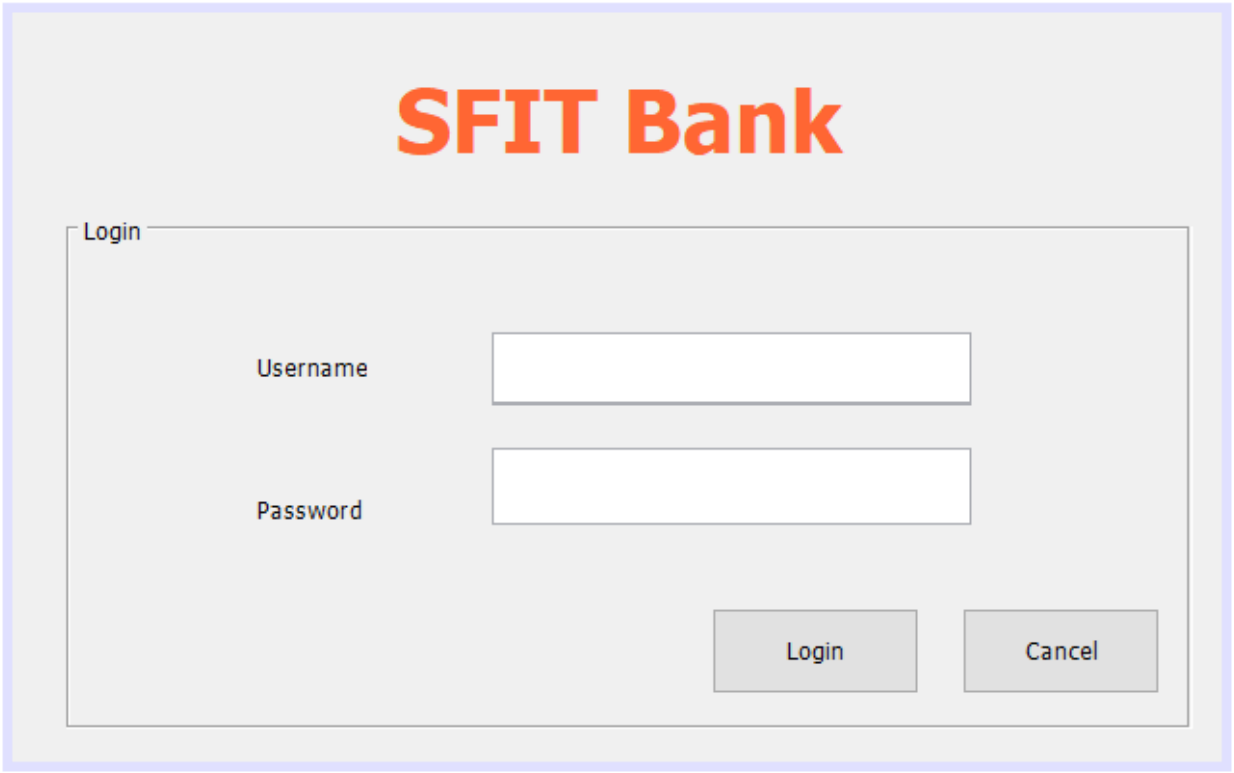
```
            }
        }
    } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(mainmenu.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
    } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(mainmenu.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
    } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(mainmenu.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(mainmenu.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
    }
    //</editor-fold>

    /* Create and display the form */
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new mainmenu().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JDesktopPane jDesktopPane1;
private javax.swing.JMenu jMenu1;
private javax.swing.JMenu jMenu2;
private javax.swing.JMenu jMenu3;
private javax.swing.JMenu jMenu4;
private javax.swing.JMenu jMenu5;
private javax.swing.JMenu jMenu6;
private javax.swing.JMenuBar jMenuBar1;
private javax.swing.JMenuItem jMenuItem1;
private javax.swing.JMenuItem jMenuItem2;
private javax.swing.JMenuItem jMenuItem3;
private javax.swing.JMenuItem jMenuItem4;
private javax.swing.JMenuItem jMenuItem5;
private javax.swing.JMenuItem jMenuItem6;
private javax.swing.JMenuItem jMenuItem7;
private javax.swing.JMenuItem jMenuItem8;
private javax.swing.JMenuItem jMenuItem9;
// End of variables declaration
```

Chapter 5

Experimental Results

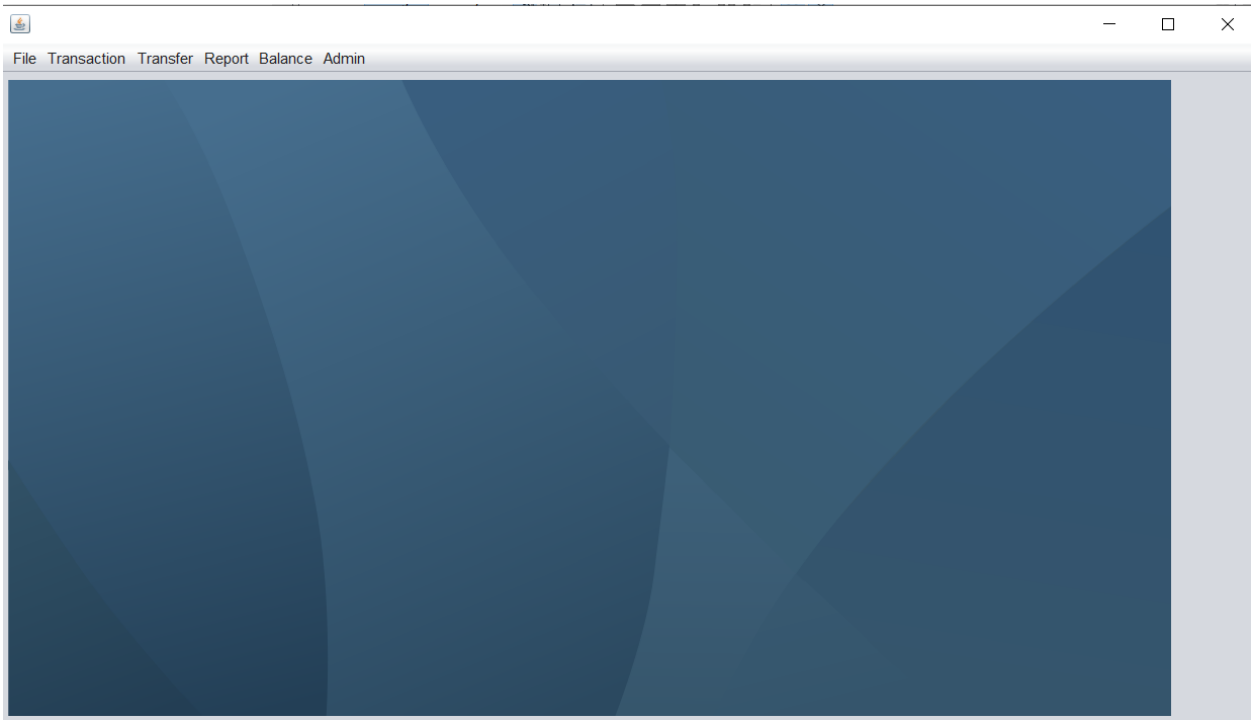
5.1 GUI

Login Page: Administrator(user) have to insert his credentials to access the system. All the data of the administrator is already stored in the database. If the credentials does not match, the user does not have access to the system.



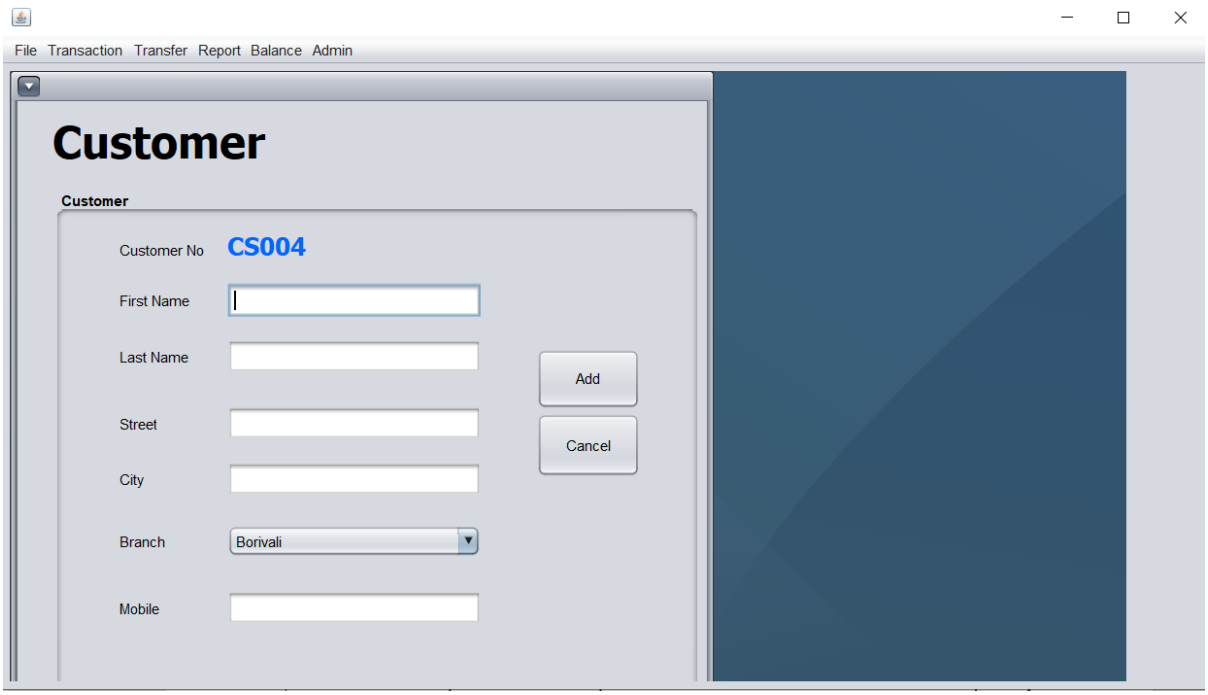
Login Page

Main Menu: Through this page the user can add a customer, create account, deposit, withdraw, transfer, report, balance check and can add new admin.

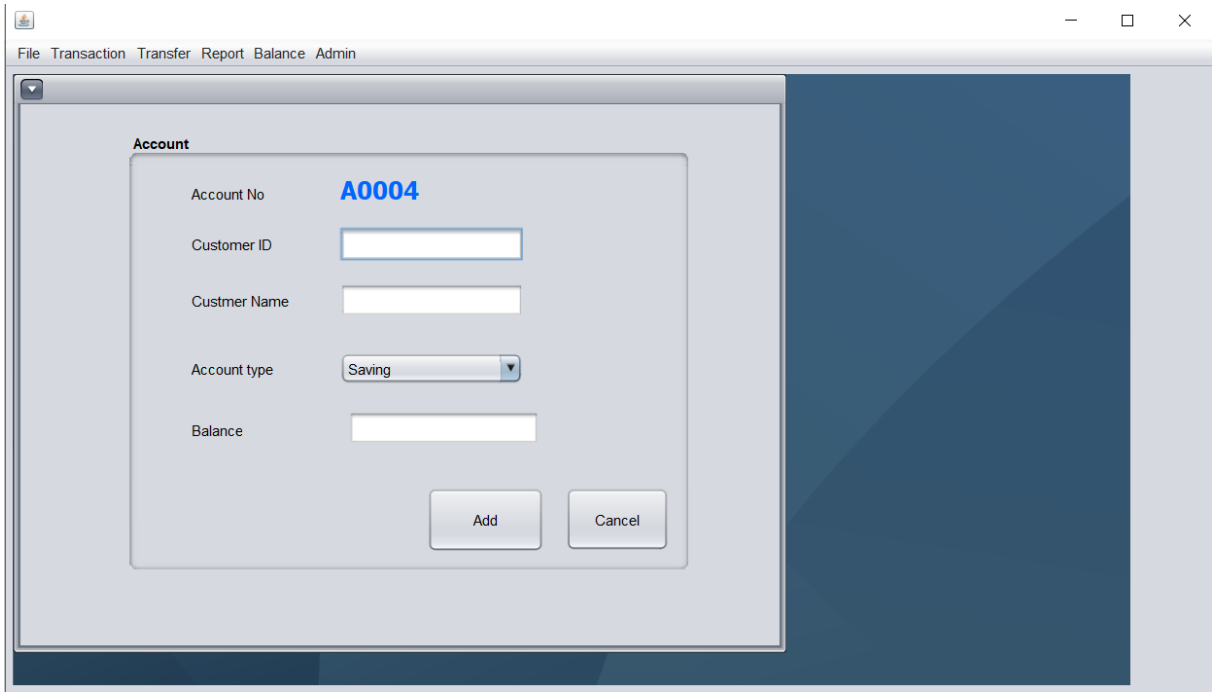


Main Menu

Addition of new customer and then creating account for the customer.

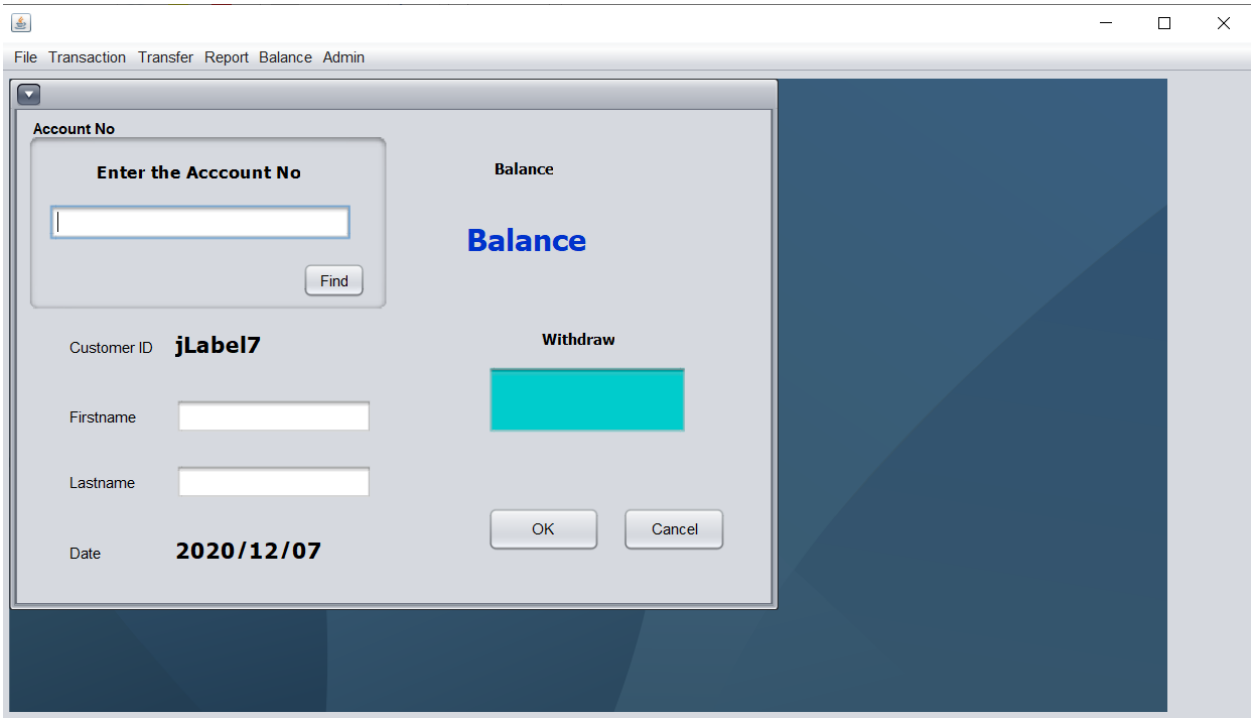


ADD Customer

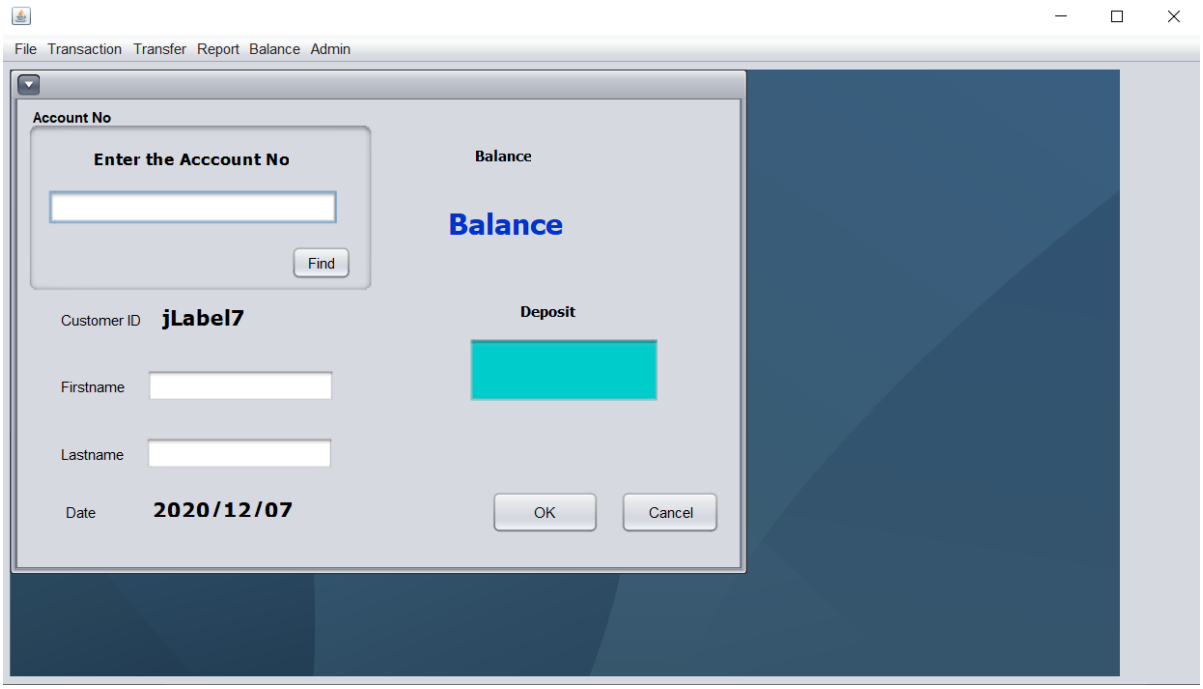


ADD account

Transaction: The user can perform two actions i.e. withdrawing and depositing on his account.
On loading of withdraw/deposit page, the user has to enter the account no. If the account no details is stored in the database, It will be reflected on this page.

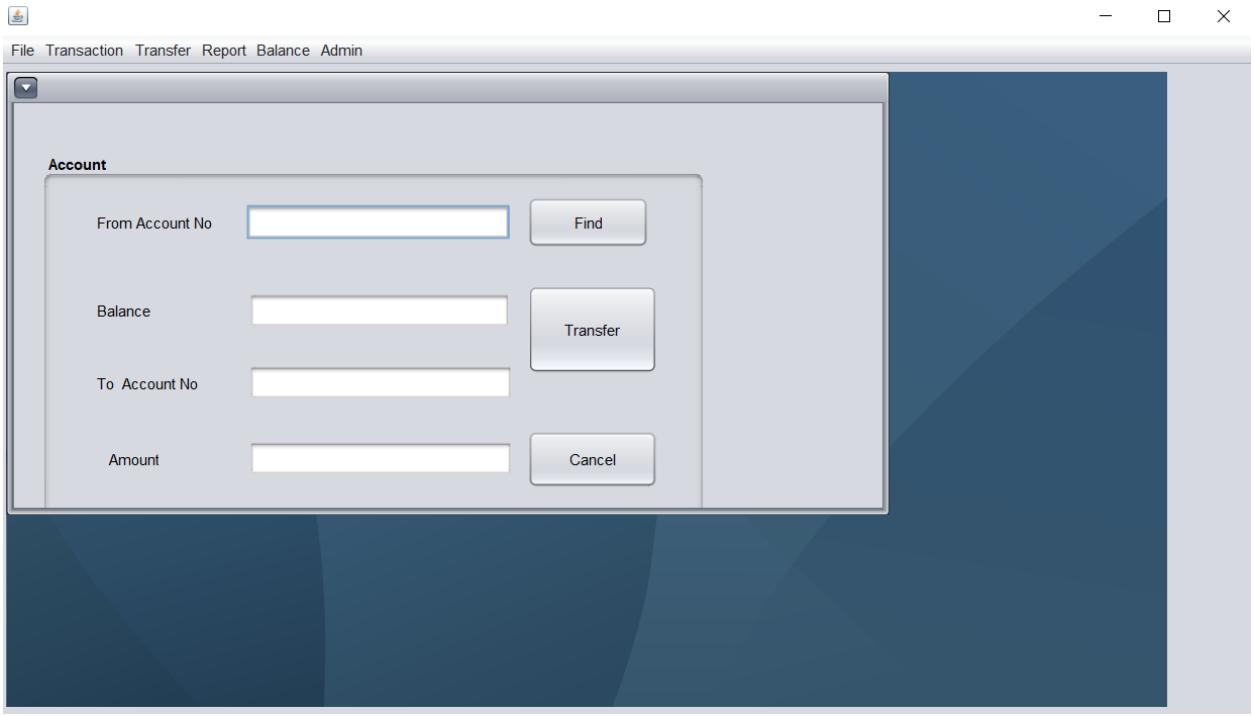


Withdraw page



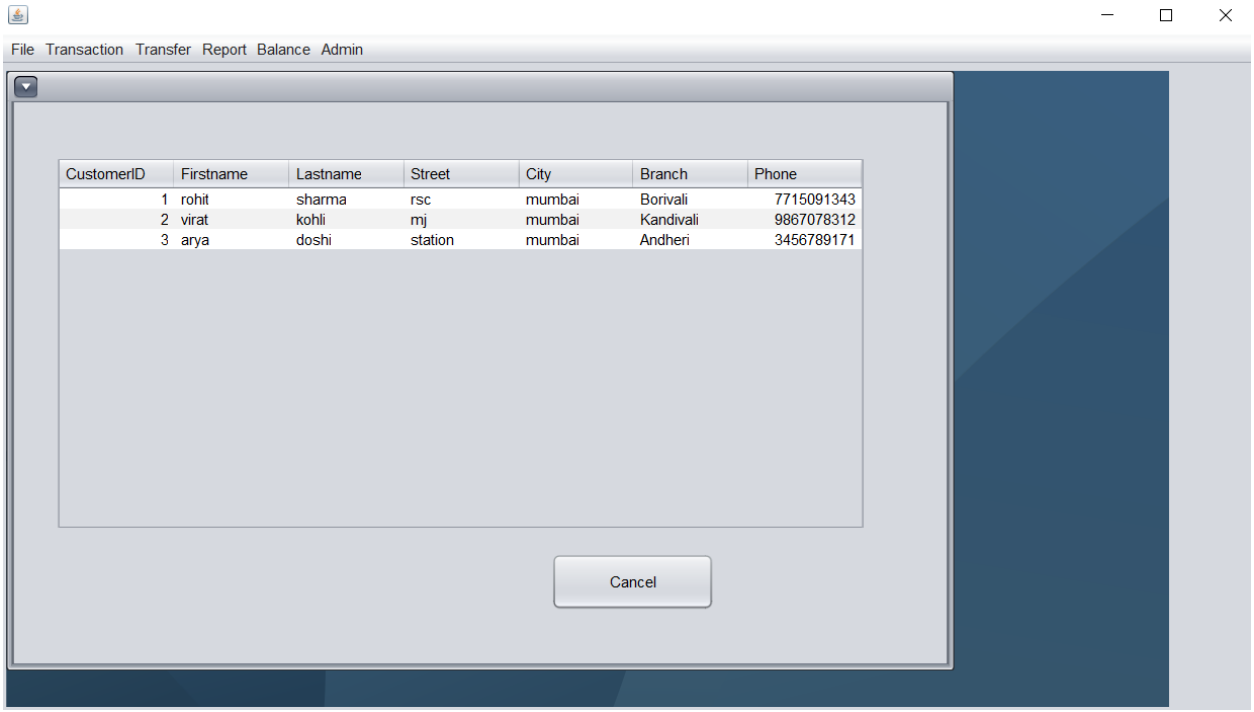
Deposit Page

Transfer: On this page, the user can transfer money from one account to another account. On loading, the user has to enter the account no's from which he wishes to transfer money.



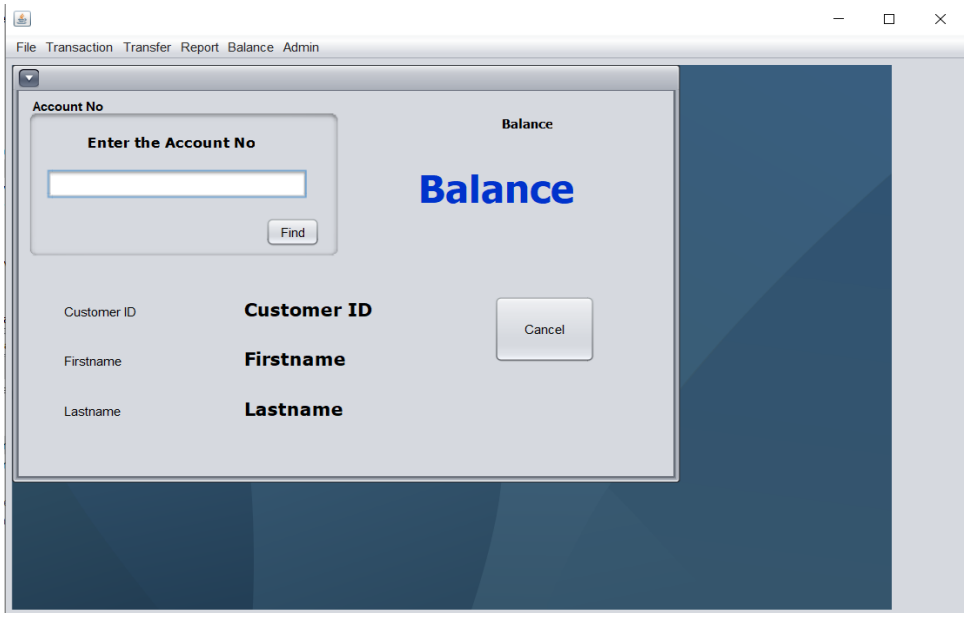
Transfer Page

Report Page: It displays all the details of the customer to the administrator.



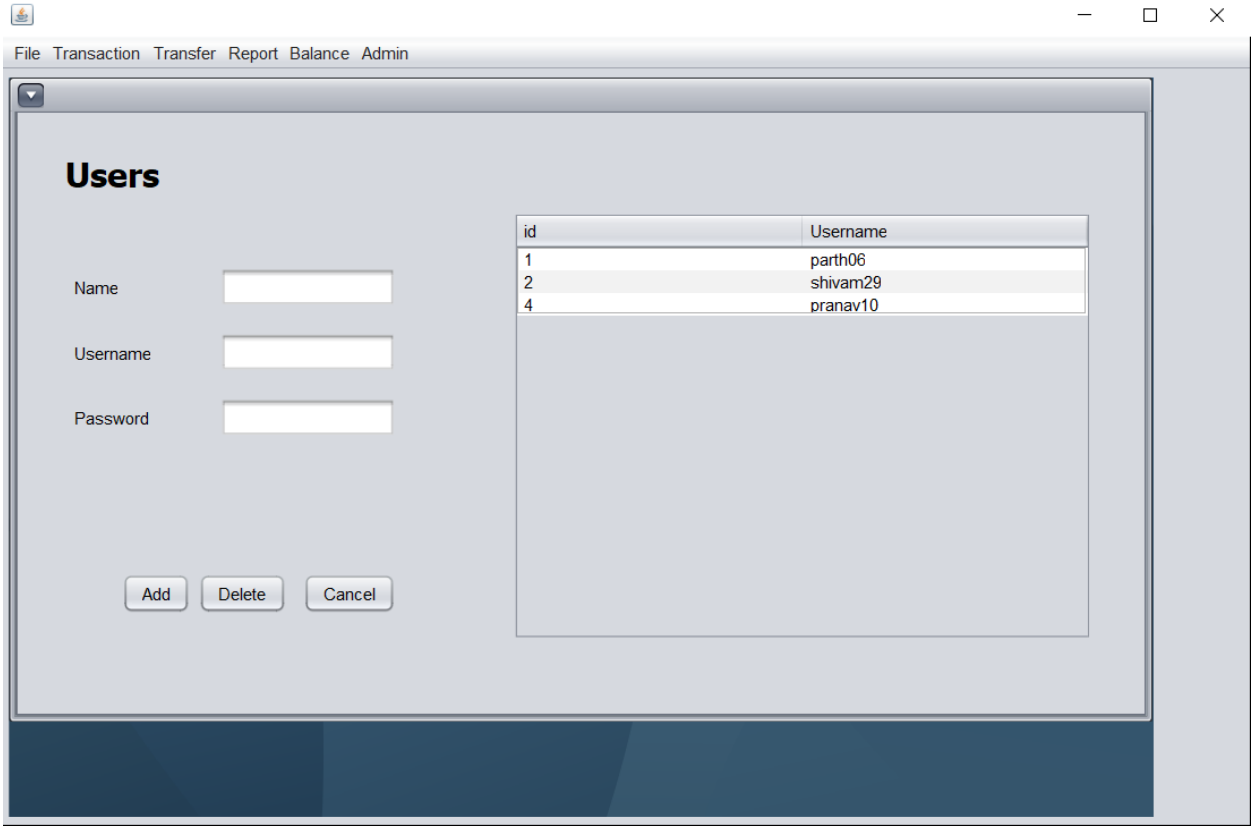
Report Page

Balance: It displays the account balance of the selected customer.



Balance page

Admin: It allows the existing administrator to create new administrators and delete existing administrator.



Admin Page

Chapter 6

Conclusion and Future Scope

6.1 Conclusion

Bank management system is a virtualization of transactions in banking system. The banking system are used manual working but when we used bank management system it is totally virtualization process which avoid manual process and converts it in automatic process. Bank management system is saving the time with accuracy than bank manual system.

6.2 Future Scope

- This project aspires to be a simulation of the Banking system for banks and also reduces the human errors caused by employees or the customer itself.
- If coupled with appropriate hardware this system can be turned into an ATM software.

References

- https://netbeans.org/kb/index.html
- https://www.javatpoint.com/online-banking-project
- https://projectsgeek.com/2016/02/complete-banking-system-java-project.html#:~:text=Proposed%20System,valid%20user%20id%20and%20password.
- https://www.apachefriends.org/index.html