**ONLINE BANKING APPLICATION**

****

***Submitted by***

| **Name of the Student** | **University Roll No.** |
| --- | --- |
| NILANJIT MAJUMDER | 12022002016004 |
| BISHRUTA ADAK | 12022002016056 |

Under the supervision of

Prof. Deepsubhra Guha Roy

***Academic Year: 2024-25***

**REPORT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING (ARTIFICIAL INTELLIGENCE MACHINE LEARNING) OF**

**MAULANA ABUL KALAM AZAD UNIVERSITY OF TECHNOLOGY**

****

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**(ARTIFICIAL INTELLIGENCE MACHINE LEARNING)**

**INSTITUTE OF ENGINEERING AND MANAGEMENT**

**KOLKATA**



*CERTIFICATE OF RECOMMENDATION*

We hereby recommend that the thesis prepared under our supervision by **Nilanjit Majumder and Bishruta Adak**

entitled **Online Banking Application**

be accepted in partial fulfillment of the requirements for the degree of **BACHELOR OF TECHNOLOGY IN “COMPUTER SCIENCE AND ENGINEERING (ARTIFICIAL INTELLIGENCE MACHINE LEARNING)”.**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

***Head, CSE(AIML)Department Project Guide***

***Institute of Engineering and Management, Kolkata***

* **Introduction**

The “Bank Account Management System” project is a model Internet Banking Site. This site enables the customers to perform the basic banking transactions by sitting at their office or at homes through PC or laptop. The system provides the access to the customer to create an account, deposit/withdraw the cash from his account, also to view reports of all accounts present. The customers can access the banks website for viewing their Account details and perform the transactions on account as per their requirements. With Internet Banking, the brick and mortar structure of the traditional banking gets converted into a click and portal model, thereby giving a concept of virtual banking a real shape. Thus, today's banking is no longer confined to branches. E-banking facilitates banking transactions by customers round the clock globally.

The primary aim of this “Bank Account Management System” is to provide an improved design methodology, which envisages the future expansion, and modification, which is necessary for a core sector like banking. This necessitates the design to be expandable and modifiable and so a modular approach is used in developing the application software.

Anybody who is an Account holder in this bank can become a member of Bank Account Management System. He has to fill a form with his personal details and Account Number. 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 day’s, managing a bank is tedious job up to certain limit. So software 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 software for bank management has become necessary which would be useful in managing the bank more efficiently. All transactions are carried out online by transferring from accounts in the same Bank or international bank. The software is meant to overcome the drawbacks of the manual system.

* **Abstract**

The Bank Account Management System is an application for maintaining a person's account in a bank. In this project I 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. Creating and managing requirements is a challenge of IT, systems and product development projects or indeed for any activity where you have to manage a contractual relationship. Organization need to effectively define and manage requirements to ensure they are meeting needs of the customer, while proving compliance and staying on the schedule and within budget.

The impact of a poorly expressed requirement can bring a business out of compliance or even cause injury or death. Requirements definition and management is an activity that can deliver a high, fast return on investment. The project analyzes the system requirements and then comes up with the requirements specifications. It studies other related systems and then come up with system specifications. The system is then designed in accordance with specifications to satisfy the requirements. The system design is then implemented with Java. 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

* **AIM of this project**

The main aim of designing and developing this Internet banking System Java primarily based Engineering project is to provide secure and efficient net banking facilities to the banking customers over the internet. Apache Server Pages, MYSQL database used to develop this bank application where all banking customers can login through the secured web page by their account login id and password. Users will have all options and features in that application like get money from western union, money transfer to others, and send cash or money to inter banking as well as other banking customers by simply adding them as payees.

The Traditional way of maintaining details of a user in a bank was to enter the details and record them. Every time the user needs to perform some transactions he has to go to bank and perform the necessary actions, which may not be so feasible all the time. It may be a hard-hitting task for the users and the bankers too. The project gives real life understanding of Online Banking System and activities performed by various roles in the supply chain. Here, we provide automation for banking system through Internet. Online Banking System project captures activities performed by different roles in real life banking which provides enhanced techniques for maintaining the required information up-to-date, which results in efficiency. The project gives real life understanding of Online Banking System and activities performed by various roles in the supply chain

**Main Goal**

**1. Motto-** Our motto is to develop a software program for managing the entire bank process related to Administration accounts customer accounts and to keep each every track about their property and their various transaction processes efficiently.

Hereby, our main objective is the customer’s satisfaction considering today’s faster in the world.

**2. Customer Satisfaction**: Client can do his operations comfortably without any risk or losing of his privacy. Our software will perform and fulfill all the tasks that any customer would desire.

**3. Saving Customer Time**: Client doesn't need to go to the bank to do small operation.

**4. Protecting the Customer:** It helps the customer to be satisfied and comfortable in his choices, this protection contains customer’s account, money and his privacy.

**5. Transferring Money:** Help client transferring money to/or another bank or country.

**Methods**

* We need to be able to generate an account number
* Account types: Savings or Current Account
* Maintain/update Balance
* Open/Close Account
* Withdraw/Deposit

1. **Administrative Modules**

Here in my project there are two types of modules. This module is the main module which performs all the main operations in the system. The major operations in the system are:

* 1. **Admin Module**

Admin can access this project there is an authorization process. If you login as an Admin then you will be redirected to the Admin Home Page and if you are a simple user you will be redirected to your Account Home Page. This performs the following functions: Create Individual Accounts, manage existing accounts, View all transactions, Balance enquiry, Delete/close account etc.

* Admin login
* Add/delete/update account
* Withdrawal/deposit/statements transaction
* Account Information
* User details list
* Active/Inactive account
* View transaction histories
  1. **User Module**

A simple user can access their account and can deposit/withdraw money from their account. User can also transfer money from their account to any other bank account. User can see their transaction report and balance enquiry too.

* User login, use PIN system
* Creating/open new account registration
* Funds transfer (local/international/domestic)
* View statements transaction
* User account details
* Change Password and Pin
  1. **Banks terms**
* All requests received from customers are logged for backend fulfillment and are effective from the time they are recorded at the branch.
* Rules and regulations applicable to normal banking transactions in India will be applicable mutatis mutandis for the transactions executed through this site.
* The BAMS Bank service cannot be claimed as a right. The bank may also convert this into a discretionary service anytime.
* Dispute between the customer and the Bank in this service is subject to the jurisdiction of the courts in the Republic of India and governed by the laws prevailing in India.
* The Bank reserves the right to modify the services offered or the Terms of service of the Bank. The changes will be notified to the customers through a notification on the Site.
  1. **Customer’s obligations**
* The customer has an obligation to maintain secrecy in regard to Username & Password registered with the Bank. The bank presupposes that login using valid Username and Password is a valid session initiated by none other than the customer.
* Transaction executed through a valid session will be construed by RR to have emanated from the registered customer and will be binding on him/her.
* The customer will not attempt or permit others to attempt accessing the BAMS Bank through any unlawful means.

1. **COMPONENTS OF BANKING APPLICATION**

Our online banking application consist of following classes :

* LOGIN
* SIGNUP
* DEPOSITE
* WITHDRAW
* TRANSACTION
* PIN CHANGE
* MINI STATEMENT
* BALANCE ENQUIRY
* FAST CASH
* CONNECTION
  1. **LOGIN**

This is a process where customers input their credentials, usually in the form of a username and password, to gain entrance into their online banking accounts. It authenticates the user's identity for secure access to the account details.

* **SOURCE CODE**

package bank.management.system;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.sql.\*;

public class Login extends JFrame implements ActionListener {

JButton login, clear, signup;

JTextField cardTextField;

JPasswordField pinTextField;

Login(){

setTitle("Bank Management System");

setLayout(null);

ImageIcon i1= new ImageIcon(ClassLoader.getSystemResource("icons/logo.jpg"));

Image i2 = i1.getImage().getScaledInstance(100, 100, Image.SCALE\_DEFAULT);

ImageIcon i3 = new ImageIcon(i2);

JLabel label = new JLabel(i3);

label.setBounds(70, 10, 100, 100);

add(label);

JLabel text = new JLabel("Welcome to Bank");

text.setFont(new Font("Oswward", Font.BOLD,38));

text.setBounds(200, 40, 400, 40);

add(text);

JLabel cardno = new JLabel("Card Number :");

cardno.setFont(new Font("Raleway", Font.BOLD,20));

cardno.setBounds(120, 150, 150, 30);

add(cardno);

cardTextField = new JTextField();

cardTextField.setBounds(300, 150, 250, 30);

cardTextField.setFont(new Font("Arial",Font.BOLD,14));

add(cardTextField);

JLabel pin = new JLabel("Pin :");

pin.setFont(new Font("Raleway", Font.BOLD,20));

pin.setBounds(120, 220, 230, 30);

add(pin);

pinTextField = new JPasswordField();

pinTextField.setBounds(300, 220, 230, 30);

pinTextField.setFont(new Font("Arial",Font.BOLD,14));

add(pinTextField);

login = new JButton("SIGN IN");

login.setBounds(300, 300, 100, 30);

login.setBackground(Color.BLACK);

login.setForeground(Color.WHITE);

login.addActionListener(this);

add(login);

clear = new JButton("CLEAR");

clear.setBounds(430, 300, 100, 30);

clear.setBackground(Color.BLACK);

clear.setForeground(Color.WHITE);

clear.addActionListener(this);

add(clear);

signup = new JButton("SIGN UP");

signup.setBounds(300, 350, 230, 30);

signup.setBackground(Color.BLACK);

signup.setForeground(Color.WHITE);

signup.addActionListener(this);

add(signup);

getContentPane().setBackground(Color.WHITE);

setSize(800,480);

setVisible(true);

setLocation(350,200);

}

public void actionPerformed(ActionEvent ae){

if (ae.getSource() == clear){

cardTextField.setText("");

pinTextField.setText("");

} else if(ae.getSource()== signup){

setVisible(false);

new SignupOne().setVisible(true);

}else if (ae.getSource()== login){

Conn conn = new Conn();

String cardnumber = cardTextField.getText();

String pinnumber = pinTextField.getText();

String query = "Select \* from log where cardno = '"+cardnumber+"' and pin = '"+pinnumber+"'";

try {

ResultSet rs = conn.s.executeQuery(query);

if (rs.next()){

setVisible(false);

new Transaction(pinnumber).setVisible(true);

}else {

JOptionPane.showMessageDialog(null,"Incorrect card number or pin");

}

}catch(Exception e){

System.out.println(e);

}

}

}

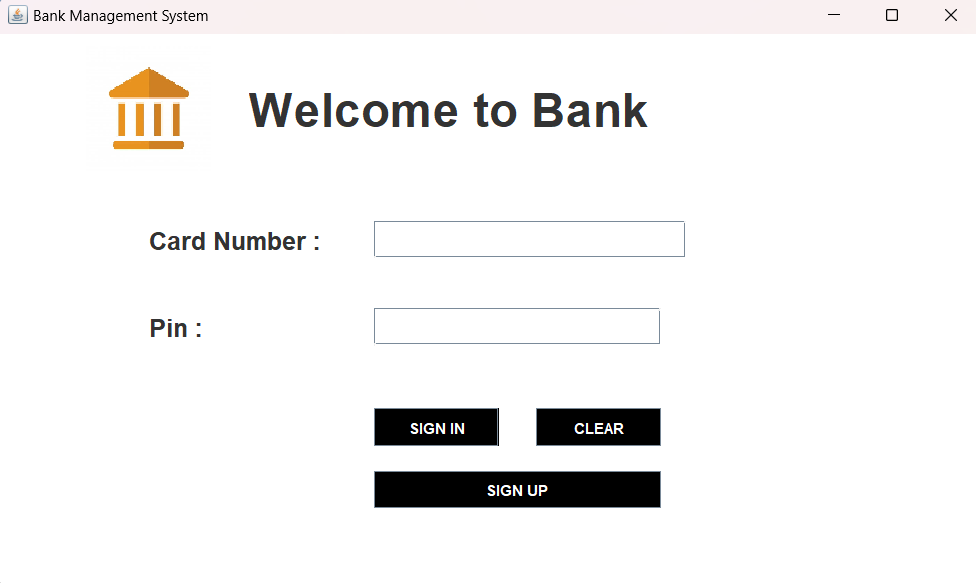
public static void main(String args[]) {

new Login();

}

}

* **SCREEN SHOT**



* 1. **SIGNUP**

SIGNUP is the first stage of registration for new users. This involves creating an account by providing personal information, using a username, password, and generally additional security features such as two-factor authentication

* **SOURCE CODE**
  + **SIGNUP-ONE**

package bank.management.system;

import java.awt.\*;

import javax.swing.\*;

import java.util.\*;

import com.toedter.calendar.JDateChooser;

import java.awt.event.\*;

public class SignupOne extends JFrame implements ActionListener{

long random;

JTextField nameTextField, fnameTextField, emailTextField, addressTextField, cityTextField, stateTextField, pincodeTextField;

JButton next;

JRadioButton male,female,other,married,unmarried;

JDateChooser dateChooser;

SignupOne(){

setLayout(null);

Random ran = new Random();

random = Math.abs((ran.nextLong()%9000L)+1000L);

JLabel formno = new JLabel("APPLICATION FORM NO :"+ random);

formno.setFont(new Font("Raleway",Font.BOLD,38));

formno.setBounds(140,20,600,40);

add(formno);

JLabel personDetails = new JLabel("Page 1: Personal Details:");

personDetails.setFont(new Font("Raleway",Font.BOLD,22));

personDetails.setBounds(290,80,400,30);

add(personDetails);

JLabel name = new JLabel("NAME:");

name.setFont(new Font("Raleway",Font.BOLD,20));

name.setBounds(100,140,100,30);

add(name);

nameTextField = new JTextField();

nameTextField.setFont(new Font("Raleway",Font.BOLD,14));

nameTextField.setBounds(300, 140, 400, 30);

add(nameTextField);

JLabel fname = new JLabel("FATHER'S NAME:");

fname.setFont(new Font("Raleway",Font.BOLD,20));

fname.setBounds(100,190,200,30);

add(fname);

fnameTextField = new JTextField();

fnameTextField.setFont(new Font("Raleway",Font.BOLD,14));

fnameTextField.setBounds(300, 190, 400, 30);

add(fnameTextField);

JLabel dob = new JLabel("DATE OF BIRTH:");

dob.setFont(new Font("Raleway",Font.BOLD,20));

dob.setBounds(100,240,200,30);

add(dob);

dateChooser = new JDateChooser();

dateChooser.setBounds(300,240,400,30);

dateChooser.setForeground(new Color(105, 105, 105));

add(dateChooser);

JLabel gender = new JLabel("GENDER:");

gender.setFont(new Font("Raleway",Font.BOLD,20));

gender.setBounds(100,290,200,30);

add(gender);

male = new JRadioButton("Male");

male.setBounds(300,290,60,30);

male.setBackground(Color.WHITE);

add(male);

female = new JRadioButton("Female");

female.setBounds(450,290,120,30);

female.setBackground(Color.WHITE);

add(female);

ButtonGroup gendergroup = new ButtonGroup();

gendergroup.add(male);

gendergroup.add(female);

JLabel email = new JLabel("EMAIL ADD:");

email.setFont(new Font("Raleway",Font.BOLD,20));

email.setBounds(100,340,200,30);

add(email);

emailTextField = new JTextField();

emailTextField.setFont(new Font("Raleway",Font.BOLD,14));

emailTextField.setBounds(300, 340, 400, 30);

add(emailTextField);

JLabel marital = new JLabel("MARITAL STATUS:");

marital.setFont(new Font("Raleway",Font.BOLD,20));

marital.setBounds(100,390,200,30);

add(marital);

married = new JRadioButton("Married");

married.setBounds(300,390,100,30);

married.setBackground(Color.WHITE);

add(married);

unmarried = new JRadioButton("Unmarried");

unmarried.setBounds(450,390,100,30);

unmarried.setBackground(Color.WHITE);

add(unmarried);

other = new JRadioButton("Other");

other.setBounds(630,390,100,30);

other.setBackground(Color.WHITE);

add(other);

ButtonGroup maritalgroup = new ButtonGroup();

maritalgroup.add(married);

maritalgroup.add(unmarried);

maritalgroup.add(other);

JLabel address = new JLabel("ADDRESS:");

address.setFont(new Font("Raleway",Font.BOLD,20));

address.setBounds(100,440,200,30);

add(address);

addressTextField = new JTextField();

addressTextField.setFont(new Font("Raleway",Font.BOLD,14));

addressTextField.setBounds(300, 440, 400, 30);

add(addressTextField);

JLabel city = new JLabel("CITY:");

city.setFont(new Font("Raleway",Font.BOLD,20));

city.setBounds(100,490,200,30);

add(city);

cityTextField = new JTextField();

cityTextField.setFont(new Font("Raleway",Font.BOLD,14));

cityTextField.setBounds(300, 490, 400, 30);

add(cityTextField);

JLabel state = new JLabel("STATE:");

state.setFont(new Font("Raleway",Font.BOLD,20));

state.setBounds(100,540,200,30);

add(state);

stateTextField = new JTextField();

stateTextField.setFont(new Font("Raleway",Font.BOLD,14));

stateTextField.setBounds(300, 540, 400, 30);

add(stateTextField);

JLabel pincode = new JLabel("PINCODE:");

pincode.setFont(new Font("Raleway",Font.BOLD,20));

pincode.setBounds(100,590,200,30);

add(pincode);

pincodeTextField = new JTextField();

pincodeTextField.setFont(new Font("Raleway",Font.BOLD,14));

pincodeTextField.setBounds(300, 590, 400, 30);

add(pincodeTextField);

next = new JButton("Next");

next.setBackground(Color.BLACK);

next.setForeground(Color.WHITE);

next.setFont(new Font("Raleway",Font.BOLD,14));

next.setBounds(620,660,80,30);

next.addActionListener(this);

add(next);

getContentPane().setBackground(Color.WHITE);

setSize(850, 800);

setLocation(350, 10);

setVisible(true);

}

public void actionPerformed(ActionEvent ae){

String formno = "" + random;

String name = nameTextField.getText();

String fname = fnameTextField.getText();

String dob = ((JTextField)dateChooser.getDateEditor().getUiComponent()).getText();

String gender = null;

if(male.isSelected()){

gender = "Male";

}

else if (female.isSelected()){

gender = "female";

}

String email = emailTextField.getText();

String marital = null;

if(married.isSelected()){

marital = "Married";

}

else if(unmarried.isSelected()){

marital= "Unmarried";

}

else if(other.isSelected()){

marital = "Other";

}

String address = addressTextField.getText();

String city = cityTextField.getText();

String pincode = pincodeTextField.getText();

String state = stateTextField.getText();

try{

if(name.equals("")){

JOptionPane.showMessageDialog(null, "Name is required");

}

else if(fname.equals("")){

JOptionPane.showMessageDialog(null, "Father name is required");

}

else if(dob.equals("")){

JOptionPane.showMessageDialog(null, "Date of birth is required");

}

else if(gender.equals("")){

JOptionPane.showMessageDialog(null, "gender of birth is required");

}

else if(email.equals("")){

JOptionPane.showMessageDialog(null, "Email of birth is required");

}

else if(marital.equals("")){

JOptionPane.showMessageDialog(null, "Marital Status is required");

}

else if(address.equals("")){

JOptionPane.showMessageDialog(null, "Address is required");

}

else if(city.equals("")){

JOptionPane.showMessageDialog(null, "City is required");

}

else if(pincode.equals("")){

JOptionPane.showMessageDialog(null, "Pincode is required");

}

else if(state.equals("")){

JOptionPane.showMessageDialog(null, "State is required");

}

else{ Conn c = new Conn();

String query = "insert into signup values('"+formno+"', '"+name+"','"+fname+"','"+dob+"','"+gender+"','"+email+"','"+marital+"','"+address+"','"+city+"','"+pincode+"','"+state+"')";

c.s.executeUpdate(query);

setVisible(false);

new SignupTwo(formno).setVisible(true);

}

}catch(Exception e){

System.out.println(e);

}

}

public static void main(String args[]) {

new SignupOne();

}

}

* **SCREEN SHOT**

****

* + **SIGNUP TWO**

package bank.management.system;

import java.awt.\*;

import javax.swing.\*;

import com.toedter.calendar.JDateChooser;

import java.awt.event.\*;

public class SignupTwo extends JFrame implements ActionListener{

long random;

JTextField pan,aadhar;

JButton next;

JRadioButton syes,sno,eyes,eno;

JDateChooser dateChooser;

JComboBox religion, category, occupation, educational,income;

String formno;

SignupTwo(String formno){

this.formno = formno;

setLayout(null);

setTitle("Page 2: NEW ACCOUNT APPLICATION FORM ");

JLabel additionalDetails = new JLabel("Page 2: Additional Details:");

additionalDetails.setFont(new Font("Raleway",Font.BOLD,22));

additionalDetails.setBounds(290,80,400,30);

add(additionalDetails);

JLabel name = new JLabel("Religion:");

name.setFont(new Font("Raleway",Font.BOLD,20));

name.setBounds(100,140,100,30);

add(name);

String valReligion[] = {"Hindu","Muslim","Sikh","Christian","Other"};

religion = new JComboBox(valReligion);

religion.setBackground(Color.WHITE);

religion.setBounds(300, 140, 400, 30);

add(religion);

JLabel fname = new JLabel("Category:");

fname.setFont(new Font("Raleway",Font.BOLD,20));

fname.setBounds(100,190,200,30);

add(fname);

String varCategory[] = {"General","OBC","SC","ST","Other"};

category = new JComboBox(varCategory);

category.setBounds(300, 190, 400, 30);

category.setBackground(Color.WHITE);

add(category);

JLabel dob = new JLabel("Income:");

dob.setFont(new Font("Raleway",Font.BOLD,20));

dob.setBounds(100,240,200,30);

add(dob);

String varIncome[] = {"NULL","< 1,50,000","< 2,50,000","< 5,00,000","upto 10,00,000"};

income = new JComboBox(varIncome);

income.setBounds(300, 240, 400, 30);

income.setBackground(Color.WHITE);

add(income);

JLabel gender = new JLabel("Educational:");

gender.setFont(new Font("Raleway",Font.BOLD,20));

gender.setBounds(100,290,200,30);

add(gender);

JLabel email = new JLabel("Qualification:");

email.setFont(new Font("Raleway",Font.BOLD,20));

email.setBounds(100,315,200,30);

add(email);

String varEducational[] = {"Non Graduate","Graduate","Post Graduate","Doctrate","Others"};

educational = new JComboBox(varEducational);

educational.setFont(new Font("Raleway",Font.BOLD,14));

educational.setBackground(Color.WHITE);

educational.setBounds(300, 315, 400, 30);

add(educational);

JLabel marital = new JLabel("Occupation:");

marital.setFont(new Font("Raleway",Font.BOLD,20));

marital.setBounds(100,390,200,30);

add(marital);

String varOccupation[] = {"Salaried","Self Employed","Bussiness","Student","Retired","Others"};

occupation = new JComboBox(varOccupation);

occupation.setBounds(300,390,400,30);

occupation.setBackground(Color.WHITE);

add(occupation);

JLabel address = new JLabel("PAN no:");

address.setFont(new Font("Raleway",Font.BOLD,20));

address.setBounds(100,440,200,30);

add(address);

pan = new JTextField();

pan.setFont(new Font("Raleway",Font.BOLD,14));

pan.setBounds(300, 440, 400, 30);

add(pan);

JLabel city = new JLabel("Aadhar no:");

city.setFont(new Font("Raleway",Font.BOLD,20));

city.setBounds(100,490,200,30);

add(city);

aadhar = new JTextField();

aadhar.setFont(new Font("Raleway",Font.BOLD,14));

aadhar.setBounds(300, 490, 400, 30);

add(aadhar);

JLabel state = new JLabel("Senior Citizen:");

state.setFont(new Font("Raleway",Font.BOLD,20));

state.setBounds(100,540,200,30);

add(state);

syes = new JRadioButton("Yes");

syes.setBounds(300,540,100,30);

syes.setBackground(Color.WHITE);

add(syes);

sno = new JRadioButton("No");

sno.setBounds(450,540,100,30);

sno.setBackground(Color.WHITE);

add(sno);

ButtonGroup senior = new ButtonGroup();

senior.add(syes);

senior.add(sno);

JLabel pincode = new JLabel("Existing Account:");

pincode.setFont(new Font("Raleway",Font.BOLD,20));

pincode.setBounds(100,590,200,30);

add(pincode);

eyes = new JRadioButton("Yes");

eyes.setBounds(300,590,100,30);

eyes.setBackground(Color.WHITE);

add(eyes);

eno = new JRadioButton("No");

eno.setBounds(450,590,100,30);

eno.setBackground(Color.WHITE);

add(eno);

ButtonGroup existing = new ButtonGroup();

existing.add(eyes);

existing.add(eno);

next = new JButton("Next");

next.setBackground(Color.BLACK);

next.setForeground(Color.WHITE);

next.setFont(new Font("Raleway",Font.BOLD,14));

next.setBounds(620,660,80,30);

next.addActionListener(this);

add(next);

getContentPane().setBackground(Color.WHITE);

setSize(850, 800);

setLocation(350, 10);

setVisible(true);

}

public void actionPerformed(ActionEvent ae){

formno = "" + random;

String sreligion = (String)religion.getSelectedItem();

String scategory = (String)category.getSelectedItem();

String sincome = (String)income.getSelectedItem();

String seducation = (String)educational.getSelectedItem();

String soccupation = (String)occupation.getSelectedItem();

String seniorcitizen = null;

if(syes.isSelected()){

seniorcitizen = "Yes";

}

else if (sno.isSelected()){

seniorcitizen = "No";

}

String existingaccount = null;

if(eyes.isSelected()){

existingaccount = "Yes";

}

else if(eno.isSelected()){

existingaccount= "No";

}

String span = pan.getText();

String saadhar = aadhar.getText();

try {

Conn c = new Conn();

String query = "insert into signuptwo values('"+formno+"', '"+sreligion+"','"+scategory+"','"+sincome+"','"+seducation+"','"+soccupation+"','"+span+"','"+saadhar+"','"+seniorcitizen+"','"+existingaccount+"')";

c.s.executeUpdate(query);

}

catch(Exception e){

System.out.println(e);

}

}

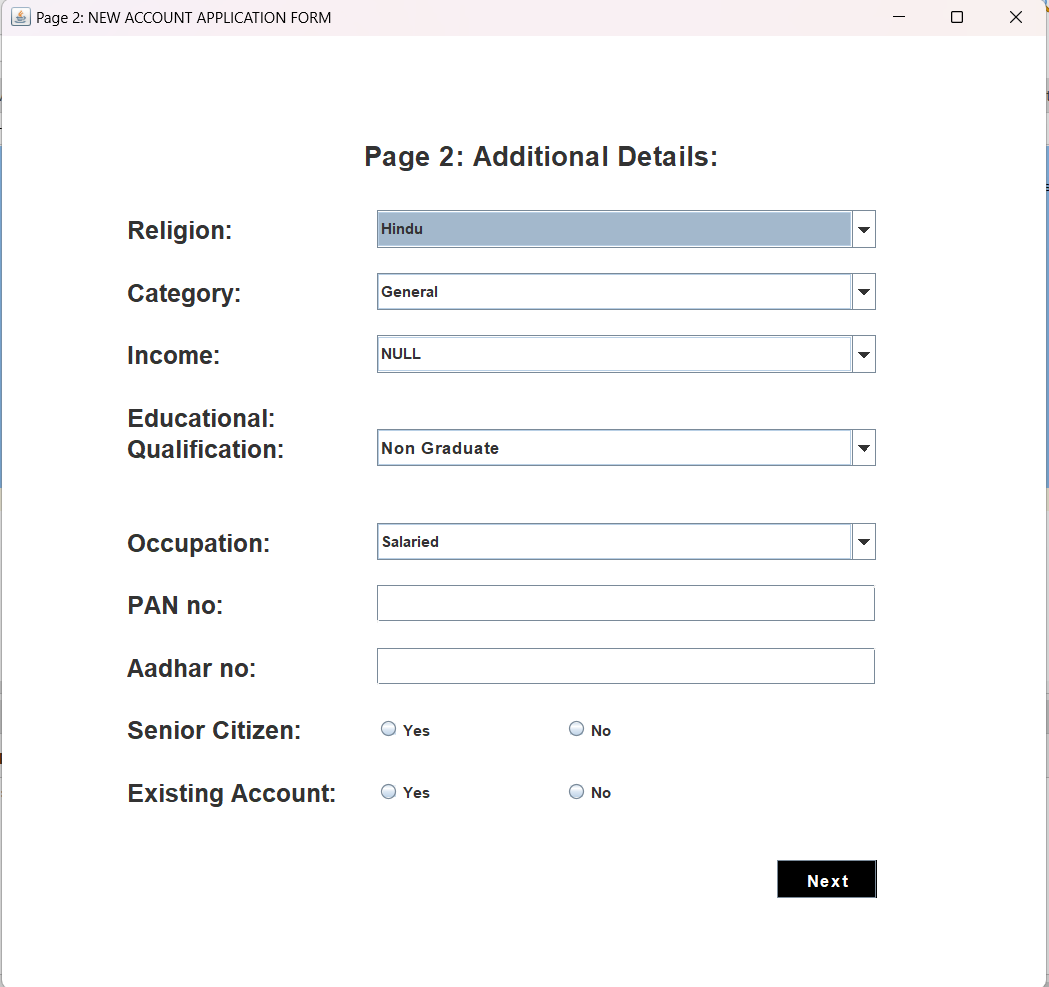
public static void main(String args[]) {

new SignupTwo("");

}

}

* **SCREEN SHOT**

****

* + **SIGNUP THREE**

package bank.management.system;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import java.sql.\*;

import java.util.\*;

public class SignupThree extends JFrame implements ActionListener {

JLabel l1,l2,l3,l4,l5,l6,l7,l8,l9,l10,l11,l12;

JRadioButton r1,r2,r3,r4;

JButton b1,b2;

JCheckBox c1,c2,c3,c4,c5,c6,c7;

String formno;

SignupThree(String formno){

this.formno = formno;

setTitle("NEW ACCOUNT APPLICATION FORM - PAGE 3");

l1 = new JLabel("Page 3: Account Details");

l1.setFont(new Font("Raleway", Font.BOLD, 22));

l2 = new JLabel("Account Type:");

l2.setFont(new Font("Raleway", Font.BOLD, 18));

l3 = new JLabel("Card Number:");

l3.setFont(new Font("Raleway", Font.BOLD, 18));

l4 = new JLabel("XXXX-XXXX-XXXX-4184");

l4.setFont(new Font("Raleway", Font.BOLD, 18));

l5 = new JLabel("(Your 16-digit Card number)");

l5.setFont(new Font("Raleway", Font.BOLD, 12));

l6 = new JLabel("It would appear on ATM Card/Cheque Book and Statements");

l6.setFont(new Font("Raleway", Font.BOLD, 12));

l7 = new JLabel("PIN:");

l7.setFont(new Font("Raleway", Font.BOLD, 18));

l8 = new JLabel("XXXX");

l8.setFont(new Font("Raleway", Font.BOLD, 18));

l9 = new JLabel("(4-digit password)");

l9.setFont(new Font("Raleway", Font.BOLD, 12));

l10 = new JLabel("Services Required:");

l10.setFont(new Font("Raleway", Font.BOLD, 18));

l11 = new JLabel("Form No:");

l11.setFont(new Font("Raleway", Font.BOLD, 14));

l12 = new JLabel(formno);

l12.setFont(new Font("Raleway", Font.BOLD, 14));

b1 = new JButton("Submit");

b1.setFont(new Font("Raleway", Font.BOLD, 14));

b1.setBackground(Color.BLACK);

b1.setForeground(Color.WHITE);

b2 = new JButton("Cancel");

b2.setFont(new Font("Raleway", Font.BOLD, 14));

b2.setBackground(Color.BLACK);

b2.setForeground(Color.WHITE);

c1 = new JCheckBox("ATM CARD");

c1.setBackground(Color.WHITE);

c1.setFont(new Font("Raleway", Font.BOLD, 16));

c2 = new JCheckBox("Internet Banking");

c2.setBackground(Color.WHITE);

c2.setFont(new Font("Raleway", Font.BOLD, 16));

c3 = new JCheckBox("Mobile Banking");

c3.setBackground(Color.WHITE);

c3.setFont(new Font("Raleway", Font.BOLD, 16));

c4 = new JCheckBox("EMAIL Alerts");

c4.setBackground(Color.WHITE);

c4.setFont(new Font("Raleway", Font.BOLD, 16));

c5 = new JCheckBox("Cheque Book");

c5.setBackground(Color.WHITE);

c5.setFont(new Font("Raleway", Font.BOLD, 16));

c6 = new JCheckBox("E-Statement");

c6.setBackground(Color.WHITE);

c6.setFont(new Font("Raleway", Font.BOLD, 16));

c7 = new JCheckBox("I hereby declares that the above entered details correct to th best of my knowledge.",true);

c7.setBackground(Color.WHITE);

c7.setFont(new Font("Raleway", Font.BOLD, 12));

r1 = new JRadioButton("Saving Account");

r1.setFont(new Font("Raleway", Font.BOLD, 16));

r1.setBackground(Color.WHITE);

r2 = new JRadioButton("Fixed Deposit Account");

r2.setFont(new Font("Raleway", Font.BOLD, 16));

r2.setBackground(Color.WHITE);

r3 = new JRadioButton("Current Account");

r3.setFont(new Font("Raleway", Font.BOLD, 16));

r3.setBackground(Color.WHITE);

r4 = new JRadioButton("Recurring Deposit Account");

r4.setFont(new Font("Raleway", Font.BOLD, 16));

r4.setBackground(Color.WHITE);

ButtonGroup groupgender = new ButtonGroup();

groupgender.add(r1);

groupgender.add(r2);

groupgender.add(r3);

groupgender.add(r4);

setLayout(null);

l11.setBounds(700,10,70,30);

add(l11);

l12.setBounds(770,10,40,30);

add(l12);

l1.setBounds(280,40,400,40);

add(l1);

l2.setBounds(100,140,200,30);

add(l2);

r1.setBounds(100,180,150,30);

add(r1);

r2.setBounds(350,180,300,30);

add(r2);

r3.setBounds(100,220,250,30);

add(r3);

r4.setBounds(350,220,250,30);

add(r4);

l3.setBounds(100,300,200,30);

add(l3);

l4.setBounds(330,300,250,30);

add(l4);

l5.setBounds(100,330,200,20);

add(l5);

l6.setBounds(330,330,500,20);

add(l6);

l7.setBounds(100,370,200,30);

add(l7);

l8.setBounds(330,370,200,30);

add(l8);

l9.setBounds(100,400,200,20);

add(l9);

l10.setBounds(100,450,200,30);

add(l10);

c1.setBounds(100,500,200,30);

add(c1);

c2.setBounds(350,500,200,30);

add(c2);

c3.setBounds(100,550,200,30);

add(c3);

c4.setBounds(350,550,200,30);

add(c4);

c5.setBounds(100,600,200,30);

add(c5);

c6.setBounds(350,600,200,30);

add(c6);

c7.setBounds(100,680,600,20);

add(c7);

b1.setBounds(250,720,100,30);

add(b1);

b2.setBounds(420,720,100,30);

add(b2);

getContentPane().setBackground(Color.WHITE);

setSize(850,850);

setLocation(500,120);

setVisible(true);

b1.addActionListener(this);

b2.addActionListener(this);

}

public void actionPerformed(ActionEvent ae){

String atype = null;

if(r1.isSelected()){

atype = "Saving Account";

}

else if(r2.isSelected()){

atype = "Fixed Deposit Account";

}

else if(r3.isSelected()){

atype = "Current Account";

}else if(r4.isSelected()){

atype = "Recurring Deposit Account";

}

Random ran = new Random();

long first7 = (ran.nextLong() % 90000000L) + 5040936000000000L;

String cardno = "" + Math.abs(first7);

long first3 = (ran.nextLong() % 9000L) + 1000L;

String pin = "" + Math.abs(first3);

String facility = "";

if(c1.isSelected()){

facility = facility + " ATM Card";

}

if(c2.isSelected()){

facility = facility + " Internet Banking";

}

if(c3.isSelected()){

facility = facility + " Mobile Banking";

}

if(c4.isSelected()){

facility = facility + " EMAIL Alerts";

}

if(c5.isSelected()){

facility = facility + " Cheque Book";

}

if(c6.isSelected()){

facility = facility + " E-Statement";

}

try{

if(ae.getSource()==b1){

if(atype.equals("")){

JOptionPane.showMessageDialog(null, "Fill all the required fields");

}else{

Conn c1 = new Conn();

String q1 = "insert into signup3 values('"+formno+"','"+atype+"','"+cardno+"','"+pin+"','"+facility+"')";

String q2 = "insert into log values('"+formno+"','"+cardno+"','"+pin+"')";

c1.s.executeUpdate(q1);

c1.s.executeUpdate(q2);

JOptionPane.showMessageDialog(null, "Card Number: " + cardno + "\n Pin:"+ pin);

setVisible(false);

new Deposite(pin).setVisible(true);

}

}else if(ae.getSource()==b2){

setVisible(false);

new Login().setVisible(true);

}

}catch(Exception ex){

ex.printStackTrace();

}

}

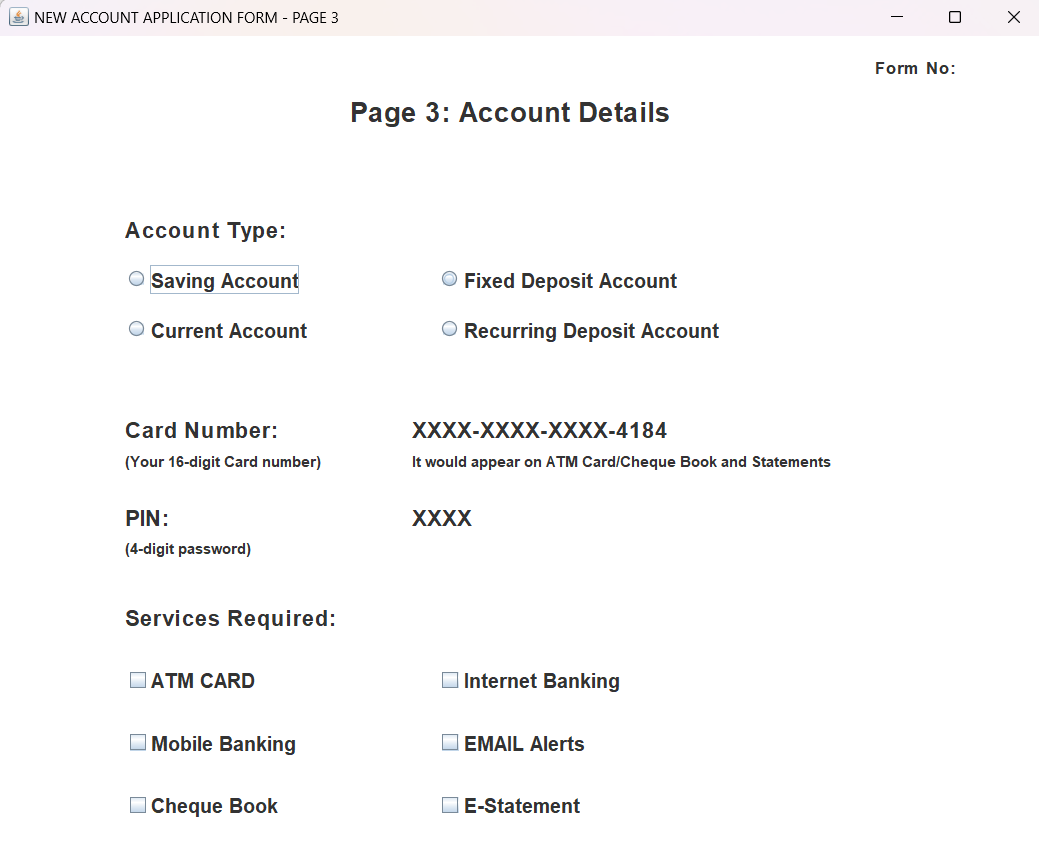
public static void main(String args[]) {

new SignupThree("").setVisible(true);

}

}

* **SCREEN SHOT**

****

* 1. **DEPOSIT**

This refers to adding money to a bank account. With online banking, deposits can be made by transferring money from another account, depositing a check on a mobile device, or by direct deposit.

package bank.management.system;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import java.util.\*;

public class Deposite extends JFrame implements ActionListener{

JTextField t1,t2;

JButton b1,b2,b3;

JLabel l1,l2,l3;

String pin;

Deposite(String pin){

this.pin = pin;

ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icons/atm.jpg"));

Image i2 = i1.getImage().getScaledInstance(1000, 1180, Image.SCALE\_DEFAULT);

ImageIcon i3 = new ImageIcon(i2);

JLabel l3 = new JLabel(i3);

l3.setBounds(0, 0, 960, 1080);

add(l3);

l1 = new JLabel("ENTER AMOUNT YOU WANT TO DEPOSIT");

l1.setForeground(Color.WHITE);

l1.setFont(new Font("System", Font.BOLD, 16));

t1 = new JTextField();

t1.setFont(new Font("Raleway", Font.BOLD, 22));

b1 = new JButton("DEPOSIT");

b2 = new JButton("BACK");

setLayout(null);

l1.setBounds(190,350,400,35);

l3.add(l1);

t1.setBounds(190,420,320,25);

l3.add(t1);

b1.setBounds(390,588,150,35);

l3.add(b1);

b2.setBounds(390,633,150,35);

l3.add(b2);

b1.addActionListener(this);

b2.addActionListener(this);

setSize(960,1080);

setLocation(500,0);

setVisible(true);

}

public void actionPerformed(ActionEvent ae){

try{

String amount = t1.getText();

Date date = new Date();

if(ae.getSource()==b1){

if(t1.getText().equals("")){

JOptionPane.showMessageDialog(null, "Please enter the Amount to you want to Deposit");

}

else{

Conn c1 = new Conn();

String query = "insert into bank values('"+pin+"', '"+date+"', 'Deposit', '"+amount+"')";

c1.s.executeUpdate(query);

JOptionPane.showMessageDialog(null, "Rs. "+amount+" Deposited Successfully");

setVisible(false);

new Transaction(pin).setVisible(true);

}

}else if(ae.getSource()==b2){

setVisible(false);

new Transaction(pin).setVisible(true);

}

}catch(Exception e){

e.printStackTrace();

}

}

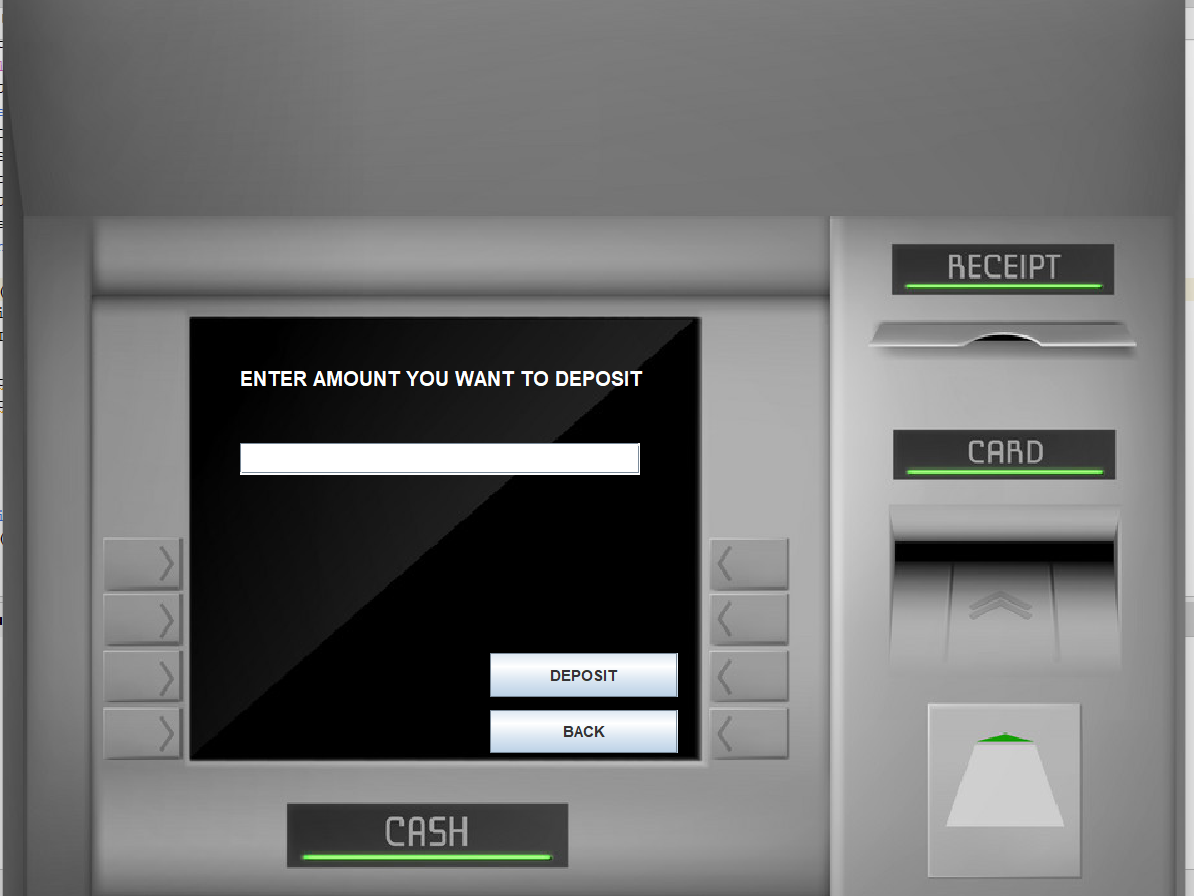
public static void main(String[] args){

new Deposite("").setVisible(true);

}

}

* **SCREEN SHOT**

****

* 1. **WITHDRAW**

This is the process of withdrawing money from a bank account. Online banking allows users to transfer funds to other accounts or request withdrawals from ATMs using a linked debit card.

* **SOURCE CODE**

package bank.management.system;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import java.util.Date;

import java.sql.\*;

public class Withdrawl extends JFrame implements ActionListener{

JTextField t1,t2;

JButton b1,b2,b3;

JLabel l1,l2,l3,l4;

String pin;

Withdrawl(String pin){

this.pin = pin;

ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icons/atm.jpg"));

Image i2 = i1.getImage().getScaledInstance(1000, 1180, Image.SCALE\_DEFAULT);

ImageIcon i3 = new ImageIcon(i2);

JLabel l3 = new JLabel(i3);

l3.setBounds(0, 0, 960, 1080);

add(l3);

l1 = new JLabel("MAXIMUM WITHDRAWAL IS RS.10,000");

l1.setForeground(Color.WHITE);

l1.setFont(new Font("System", Font.BOLD, 16));

l2 = new JLabel("PLEASE ENTER YOUR AMOUNT");

l2.setForeground(Color.WHITE);

l2.setFont(new Font("System", Font.BOLD, 16));

t1 = new JTextField();

t1.setFont(new Font("Raleway", Font.BOLD, 25));

b1 = new JButton("WITHDRAW");

b2 = new JButton("BACK");

setLayout(null);

l1.setBounds(190,350,400,20);

l3.add(l1);

l2.setBounds(190,400,400,20);

l3.add(l2);

t1.setBounds(190,450,330,30);

l3.add(t1);

b1.setBounds(390,588,150,35);

l3.add(b1);

b2.setBounds(390,633,150,35);

l3.add(b2);

b1.addActionListener(this);

b2.addActionListener(this);

setSize(960,1080);

setLocation(500,0);

setVisible(true);

}

public void actionPerformed(ActionEvent ae){

try{

String amount = t1.getText();

Date date = new Date();

if(ae.getSource()==b1){

if(t1.getText().equals("")){

JOptionPane.showMessageDialog(null, "Please enter the Amount to you want to Withdraw");

}else{

Conn c1 = new Conn();

ResultSet rs = c1.s.executeQuery("select \* from bank where pin = '"+pin+"'");

int balance = 0;

while(rs.next()){

if(rs.getString("type").equals("Deposit")){

balance += Integer.parseInt(rs.getString("amount"));

}else{

balance -= Integer.parseInt(rs.getString("amount"));

}

}

if(balance < Integer.parseInt(amount)){

JOptionPane.showMessageDialog(null, "Insuffient Balance");

return;

}

c1.s.executeUpdate("insert into bank values('"+pin+"', '"+date+"', 'Withdrawl', '"+amount+"')");

JOptionPane.showMessageDialog(null, "Rs. "+amount+" Debited Successfully");

setVisible(false);

new Transaction(pin).setVisible(true);

}

}else if(ae.getSource()==b2){

setVisible(false);

new Transaction(pin).setVisible(true);

}

}catch(Exception e){

e.printStackTrace();

System.out.println("error: "+e);

}

}

public static void main(String[] args){

new Withdrawl("").setVisible(true);

}

}

* **SCREEN SHOT**



* 1. **TRANSACTION**

Any action that transfers funds to or from an account, such as a payment, deposit, withdrawal, or transfer, is a transaction. Users can view detailed records of each case to follow up.

* **SOURCE CODE**

package bank.management.system;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import java.sql.\*;

public class Transaction extends JFrame implements ActionListener{

JLabel l1;

JButton b1,b2,b3,b4,b5,b6,b7;

String pin;

Transaction(String pin){

this.pin = pin;

ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icons/atm.jpg"));

Image i2 = i1.getImage().getScaledInstance(1000, 1180, Image.SCALE\_DEFAULT);

ImageIcon i3 = new ImageIcon(i2);

JLabel l2 = new JLabel(i3);

l2.setBounds(0, 0, 960, 1080);

add(l2);

l1 = new JLabel("Please Select Your Transaction");

l1.setForeground(Color.WHITE);

l1.setFont(new Font("System", Font.BOLD, 16));

b1 = new JButton("DEPOSIT");

b2 = new JButton("CASH WITHDRAWL");

b3 = new JButton("FAST CASH");

b4 = new JButton("MINI STATEMENT");

b5 = new JButton("PIN CHANGE");

b6 = new JButton("BALANCE ENQUIRY");

b7 = new JButton("EXIT");

setLayout(null);

l1.setBounds(235,400,700,35);

l2.add(l1);

b1.setBounds(170,499,150,35);

l2.add(b1);

b2.setBounds(390,499,150,35);

l2.add(b2);

b3.setBounds(170,543,150,35);

l2.add(b3);

b4.setBounds(390,543,150,35);

l2.add(b4);

b5.setBounds(170,588,150,35);

l2.add(b5);

b6.setBounds(390,588,150,35);

l2.add(b6);

b7.setBounds(390,633,150,35);

l2.add(b7);

b1.addActionListener(this);

b2.addActionListener(this);

b3.addActionListener(this);

b4.addActionListener(this);

b5.addActionListener(this);

b6.addActionListener(this);

b7.addActionListener(this);

setSize(960,1080);

setLocation(500,0);

setVisible(true);

}

public void actionPerformed(ActionEvent ae){

if(ae.getSource()==b7){

System.exit(0);

}else if(ae.getSource()==b1){

setVisible(false);

new Deposite(pin).setVisible(true);

}else if(ae.getSource()==b2){

setVisible(false);

new Withdrawl(pin).setVisible(true);

}else if(ae.getSource()==b3){

setVisible(false);

new Fastcash(pin).setVisible(true);

}else if(ae.getSource()==b5){

setVisible(false);

new PinChange(pin).setVisible(true);

}else if(ae.getSource()==b6){

this.setVisible(false);

new BalanceEnquiry(pin).setVisible(true);

}else if(ae.getSource()==b4){

new MiniStatement(pin).setVisible(true);

}

}

public static void main(String[] args){

new Transaction("").setVisible(true);

}

}

* **SCREEN SHOT**

****

* 1. **PIN CHANGE**

Online banking allows users to change their PIN (Personal Identification Number) from time to time for security purposes. This is usually done in the app system or through an ATM.

* **SOURCE CODE**

package bank.management.system;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import java.sql.\*;

public class PinChange extends JFrame implements ActionListener{

JPasswordField t1,t2;

JButton b1,b2;

JLabel l1,l2,l3;

String pin;

PinChange(String pin){

this.pin = pin;

ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icons/atm.jpg"));

Image i2 = i1.getImage().getScaledInstance(1000, 1180, Image.SCALE\_DEFAULT);

ImageIcon i3 = new ImageIcon(i2);

JLabel l4 = new JLabel(i3);

l4.setBounds(0, 0, 960, 1080);

add(l4);

l1 = new JLabel("CHANGE YOUR PIN");

l1.setFont(new Font("System", Font.BOLD, 16));

l1.setForeground(Color.WHITE);

l2 = new JLabel("New PIN:");

l2.setFont(new Font("System", Font.BOLD, 16));

l2.setForeground(Color.WHITE);

l3 = new JLabel("Re-Enter New PIN:");

l3.setFont(new Font("System", Font.BOLD, 16));

l3.setForeground(Color.WHITE);

t1 = new JPasswordField();

t1.setFont(new Font("Raleway", Font.BOLD, 25));

t2 = new JPasswordField();

t2.setFont(new Font("Raleway", Font.BOLD, 25));

b1 = new JButton("CHANGE");

b2 = new JButton("BACK");

b1.addActionListener(this);

b2.addActionListener(this);

setLayout(null);

l1.setBounds(280,330,800,35);

l4.add(l1);

l2.setBounds(180,390,150,35);

l4.add(l2);

l3.setBounds(180,440,200,35);

l4.add(l3);

t1.setBounds(350,390,180,25);

l4.add(t1);

t2.setBounds(350,440,180,25);

l4.add(t2);

b1.setBounds(390,588,150,35);

l4.add(b1);

b2.setBounds(390,633,150,35);

l4.add(b2);

setSize(960,1080);

setLocation(500,0);

setUndecorated(true);

setVisible(true);

}

public void actionPerformed(ActionEvent ae){

try{

String npin = t1.getText();

String rpin = t2.getText();

if(!npin.equals(rpin)){

JOptionPane.showMessageDialog(null, "Entered PIN does not match");

return;

}

if(ae.getSource()==b1){

if (t1.getText().equals("")){

JOptionPane.showMessageDialog(null, "Enter New PIN");

}

if (t2.getText().equals("")){

JOptionPane.showMessageDialog(null, "Re-Enter new PIN");

}

Conn c1 = new Conn();

String q1 = "update bank set pin = '"+rpin+"' where pin = '"+pin+"' ";

String q2 = "update log set pin = '"+rpin+"' where pin = '"+pin+"' ";

String q3 = "update signup3 set pin = '"+rpin+"' where pin = '"+pin+"' ";

c1.s.executeUpdate(q1);

c1.s.executeUpdate(q2);

c1.s.executeUpdate(q3);

JOptionPane.showMessageDialog(null, "PIN changed successfully");

setVisible(false);

new Transaction(rpin).setVisible(true);

}else if(ae.getSource()==b2){

new Transaction(pin).setVisible(true);

setVisible(false);

}

}catch(Exception e){

e.printStackTrace();

}

}

public static void main(String[] args){

new PinChange("").setVisible(true);

}

}

* **SCREEN SHOT**

****

* 1. **MINI STATEMENT**

A quick overview of recent transactions in the account. It summarizes the last few transactions, and allows users to view their recent banking activities.

* **SOURCE CODE**

package bank.management.system;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import java.sql.\*;

public class MiniStatement extends JFrame implements ActionListener{

JButton b1, b2;

JLabel l1;

MiniStatement(String pin){

super("Mini Statement");

getContentPane().setBackground(Color.WHITE);

setSize(400,600);

setLocation(20,20);

l1 = new JLabel();

add(l1);

JLabel l2 = new JLabel("NB Bank");

l2.setBounds(150, 20, 100, 20);

add(l2);

JLabel l3 = new JLabel();

l3.setBounds(20, 80, 300, 20);

add(l3);

JLabel l4 = new JLabel();

l4.setBounds(20, 400, 300, 20);

add(l4);

try{

Conn c = new Conn();

ResultSet rs = c.s.executeQuery("select \* from log where pin = '"+pin+"'");

while(rs.next()){

l3.setText("Card Number: " + rs.getString("cardno").substring(0, 4) + "XXXXXXXX" + rs.getString("cardno").substring(12));

}

}catch(Exception e){}

try{

int balance = 0;

Conn c1 = new Conn();

ResultSet rs = c1.s.executeQuery("SELECT \* FROM bank where pin = '"+pin+"'");

while(rs.next()){

l1.setText(l1.getText() + "<html>"+rs.getString("date")+ "&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;" + rs.getString("mode") + "&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;" + rs.getString("amount") + "<br><br><html>");

if(rs.getString("mode").equals("Deposit")){

balance += Integer.parseInt(rs.getString("amount"));

}else{

balance -= Integer.parseInt(rs.getString("amount"));

}

}

l4.setText("Your total Balance is Rs "+balance);

}catch(Exception e){

e.printStackTrace();

}

setLayout(null);

b1 = new JButton("Exit");

add(b1);

b1.addActionListener(this);

l1.setBounds(20, 140, 400, 200);

b1.setBounds(20, 500, 100, 25);

}

public void actionPerformed(ActionEvent ae){

this.setVisible(false);

}

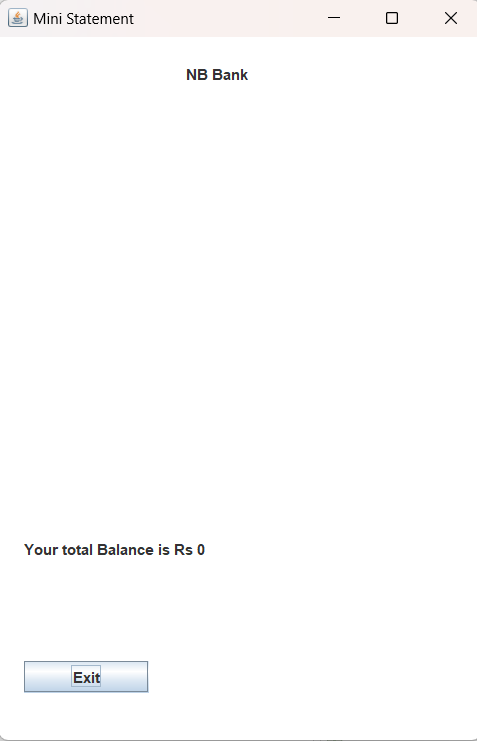
public static void main(String[] args){

new MiniStatement("").setVisible(true);

}

}

* **SCREEN SHOT**

****

* 1. **BALANCE ENQUIRY**

This feature allows users to check the current balance in their bank account. It’s an easy way to see how much money is in their account at any given time.

* **SOURCE CODE**

package bank.management.system;

import java.awt.\*;

import java.awt.event.\*;

import java.sql.ResultSet;

import javax.swing.\*;

import java.util.\*;

class BalanceEnquiry extends JFrame implements ActionListener {

JTextField t1, t2;

JButton b1, b2, b3;

JLabel l1, l2, l3;

String pin;

BalanceEnquiry(String pin) {

this.pin = pin;

ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icons/atm.jpg"));

Image i2 = i1.getImage().getScaledInstance(1000, 1180, Image.SCALE\_DEFAULT);

ImageIcon i3 = new ImageIcon(i2);

JLabel l3 = new JLabel(i3);

l3.setBounds(0, 0, 960, 1080);

add(l3);

l1 = new JLabel();

l1.setForeground(Color.WHITE);

l1.setFont(new Font("System", Font.BOLD, 16));

b1 = new JButton("BACK");

setLayout(null);

l1.setBounds(190, 350, 400, 35);

l3.add(l1);

b1.setBounds(390, 633, 150, 35);

l3.add(b1);

int balance = 0;

try{

Conn c1 = new Conn();

ResultSet rs = c1.s.executeQuery("select \* from bank where pin = '"+pin+"'");

while (rs.next()) {

if (rs.getString("mode").equals("Deposit")) {

balance += Integer.parseInt(rs.getString("amount"));

} else {

balance -= Integer.parseInt(rs.getString("amount"));

}

}

}catch(Exception e){}

l1.setText("Your Current Account Balance is Rs "+balance);

b1.addActionListener(this);

setSize(960, 1080);

setUndecorated(true);

setLocation(500, 0);

setVisible(true);

}

public void actionPerformed(ActionEvent ae) {

setVisible(false);

new Transaction(pin).setVisible(true);

}

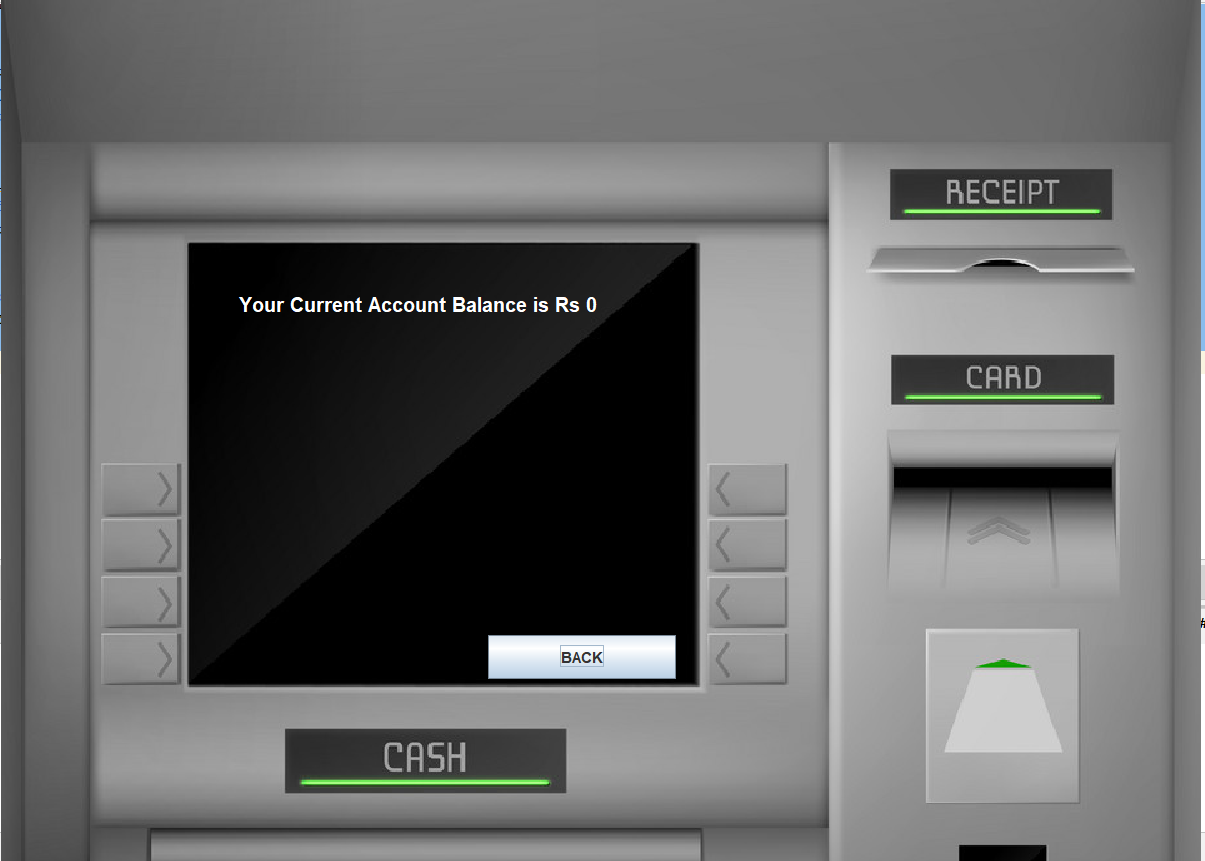
public static void main(String[] args) {

new BalanceEnquiry("").setVisible(true);

}

}

* **SCREEN SHOT**

****

* 1. **FAST CASH**

This feature allows users to quickly withdraw a predetermined amount of cash within the system to allow quick access to an ATM without multiple steps

* **SOURCE CODE**

package bank.management.system;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import java.sql.\*;

import java.util.Date;

public class Fastcash extends JFrame implements ActionListener {

JLabel l1, l2;

JButton b1, b2, b3, b4, b5, b6, b7, b8;

JTextField t1;

String pin;

Fastcash(String pin) {

this.pin = pin;

ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icons/atm.jpg"));

Image i2 = i1.getImage().getScaledInstance(1000, 1180, Image.SCALE\_DEFAULT);

ImageIcon i3 = new ImageIcon(i2);

JLabel l3 = new JLabel(i3);

l3.setBounds(0, 0, 960, 1080);

add(l3);

l1 = new JLabel("SELECT WITHDRAWL AMOUNT");

l1.setForeground(Color.WHITE);

l1.setFont(new Font("System", Font.BOLD, 16));

b1 = new JButton("Rs 100");

b2 = new JButton("Rs 500");

b3 = new JButton("Rs 1000");

b4 = new JButton("Rs 2000");

b5 = new JButton("Rs 5000");

b6 = new JButton("Rs 10000");

b7 = new JButton("BACK");

setLayout(null);

l1.setBounds(235, 400, 700, 35);

l3.add(l1);

b1.setBounds(170, 499, 150, 35);

l3.add(b1);

b2.setBounds(390, 499, 150, 35);

l3.add(b2);

b3.setBounds(170, 543, 150, 35);

l3.add(b3);

b4.setBounds(390, 543, 150, 35);

l3.add(b4);

b5.setBounds(170, 588, 150, 35);

l3.add(b5);

b6.setBounds(390, 588, 150, 35);

l3.add(b6);

b7.setBounds(390, 633, 150, 35);

l3.add(b7);

b1.addActionListener(this);

b2.addActionListener(this);

b3.addActionListener(this);

b4.addActionListener(this);

b5.addActionListener(this);

b6.addActionListener(this);

b7.addActionListener(this);

setSize(960, 1080);

setLocation(500, 0);

setVisible(true);

}

public void actionPerformed(ActionEvent ae) {

try {

if(ae.getSource() == b7) {

setVisible(false);

new Transaction(pin).setVisible(true);

}else{

String amount = ((JButton)ae.getSource()).getText().substring(3); //k

Conn c = new Conn();

ResultSet rs = c.s.executeQuery("select \* from bank where pin = '"+pin+"'");

int balance = 0;

while (rs.next()) {

if (rs.getString("type").equals("Deposit")) {

balance += Integer.parseInt(rs.getString("amount"));

} else {

balance -= Integer.parseInt(rs.getString("amount"));

}

} String num = "17";

if (ae.getSource() != b7 && balance < Integer.parseInt(amount)) {

JOptionPane.showMessageDialog(null, "Insuffient Balance");

return;

}

else if (ae.getSource() == b7) {

setVisible(false);

new Transaction(pin).setVisible(true);

}else{

Date date = new Date();

c.s.executeUpdate("insert into bank values('"+pin+"', '"+date+"', 'Withdrawl', '"+amount+"')");

JOptionPane.showMessageDialog(null, "Rs. "+amount+" Debited Successfully");

setVisible(false);

new Transaction(pin).setVisible(true);

}

}

} catch (Exception e) {

e.printStackTrace();

}

}

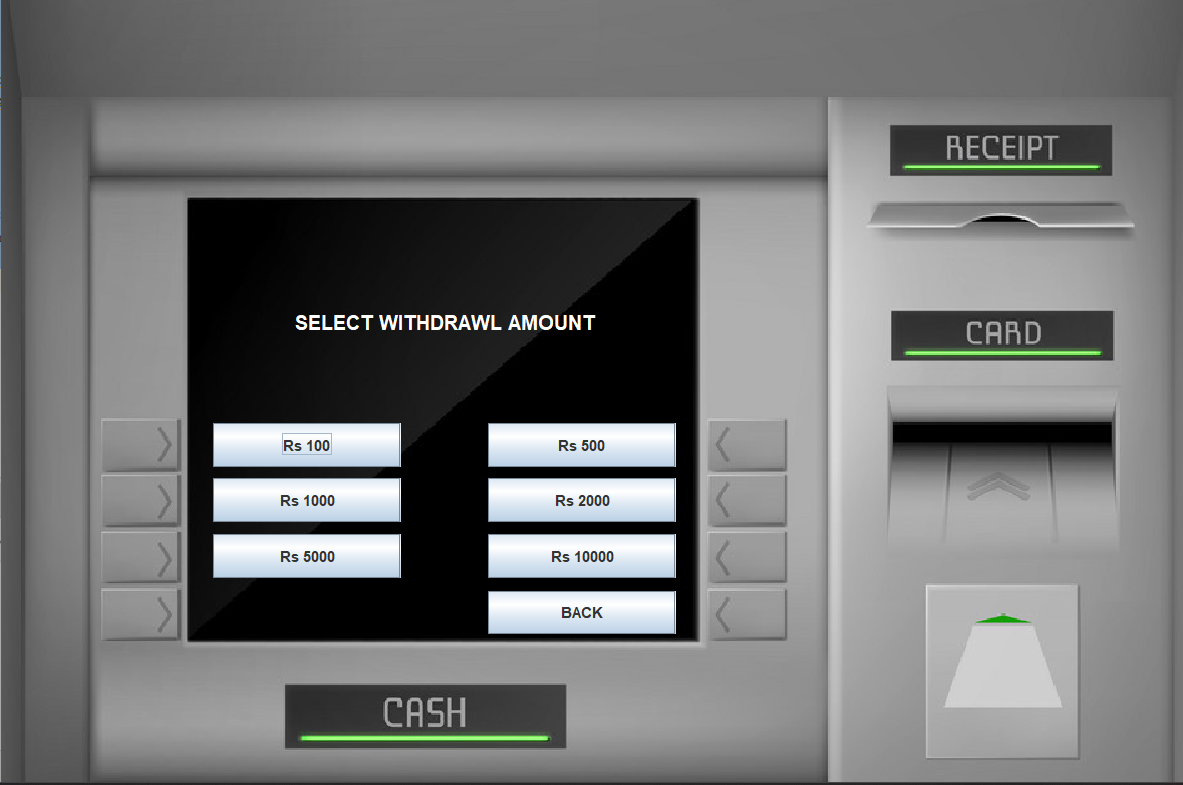
public static void main(String[] args) {

new Fastcash("").setVisible(true);

}

}

* **SCREEN SHOT**

****

* 1. **CONNECTION**
* **SOURCE CODE**

package bank.management.system;

import java.sql.\*;

public class Conn {

Connection c;

Statement s;

public Conn(){

try{

c = DriverManager.getConnection("jdbc:mysql:///bankmanagementsystem","root","Nil.2003");

s = c.createStatement();

}catch(Exception e){

System.out.println(e);

}

}

}

1. **Database:**

In this online banking we use MySQL to handle and store Information organizing it into a well-structured relational Data base called BankManagementSystem. this database comprises five name tables that get different exploiter and dealing inside information to enable prompt bill direction and banking trading operations. Here a breakdown of each table:

In this online banking application, we use **MySQL** to handle and store data, organizing it into a well-structured relational database called **BankManagementSystem**. This database comprises five key tables that capture various user and transaction details to enable efficient account management and banking operations. Here’s a breakdown of each table:

### **Signup Table**

* **Structure**: This table records personal information for each user signing up for an account.
* **Fields**:
  + formno: Unique identifier for the signup form.
  + name, father\_name: Personal details like user’s and father's name.
  + dob: Date of birth.
  + gender: User's gender.
  + email: Contact email address.
  + marital\_status: Marital status of the user.
  + address, city, pincode, state: Detailed address information.
* **Purpose**: Captures fundamental personal details required for identification and communication with the user.

### **SignupTwo Table**

* **Structure**: Stores additional information gathered during the second stage of the signup process.
* **Fields**:
  + formno: Unique signup identifier linking this table to the main signup table.
  + religion, category, income, education, occupation: Demographic and socio-economic data.
  + pan: PAN (Permanent Account Number) for financial identification.
  + aadhar: Aadhar number for identity verification.
  + seniorsitizen, existingaccount: Flags to indicate if the user is a senior citizen or holds an existing account.
* **Purpose**: Collects additional background information to comply with regulatory and identity verification standards.

### **Signup3 Table**

* **Structure**: This table captures account-specific setup details as part of the final signup step.
* **Fields**:
  + formno: Unique identifier linked to the previous signup stages.
  + atype: Type of account (e.g., Savings, Current).
  + cardno: Assigned card number for banking access.
  + pin: Personal Identification Number for secure transactions.
  + facility: Lists any extra facilities or features enabled for the user (e.g., Internet Banking, Mobile Banking).
* **Purpose**: Manages account settings and enables linkage with the user’s debit/credit card and PIN.

### **Log Table**

* **Structure**: Stores secure login details required for user authentication.
* **Fields**:
  + formno: Unique form number linking it to other tables.
  + cardno: Card number used for login.
  + pin: PIN for authenticating access to the account.
* **Purpose**: Facilitates secure login by verifying the card number and PIN.

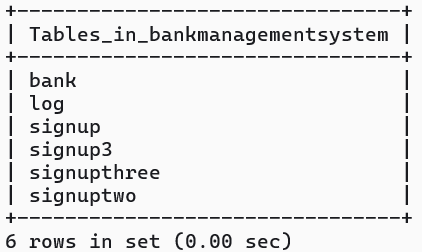
### **Bank Table**

* **Structure**: Maintains transaction details for each user, supporting a transaction history.
* **Fields**:
  + pin: User's PIN, linking this table to other tables.
  + date: Date of the transaction.
  + type: Type of transaction (e.g., Deposit, Withdrawal).
  + amount: Amount involved in the transaction.
* **Purpose**: Records all banking transactions to provide users with a history of their deposits, withdrawals, and transfers.

### **Summary**

The **BankManagementSystem** database is structured to handle user information across different signup stages, manage authentication through a secure log table, and store detailed transaction records in the bank table. MySQL's relational model helps link data across these tables efficiently, ensuring quick retrieval, integrity, and security of user and account information.

* **Screen shot**

****

1. **Benefits of online banking**

Many of us lead busy lives. Some of us are up before the crack of dawn, getting ourselves prepared so we can in turn get our families ready for the day. We rush to work, rush to get the kids to school, and at the end of the day we rush home only to brace ourselves for the next day. After a hectic day, the last thing you want to do is spend time waiting in line at the bank, or even the post office. That's where Online Banking comes in. Many of the benefits of doing our banking online are obvious:

* You don't have to wait in line.
* You don't have to plan your day around the bank's hours.
* You can look at your balance whenever you want, not just when you get a statement.

There are some hidden benefits too. As a young bank customer, you're just learning how to manage your money and observe your spending patterns. Online banking allows you to watch your money on a daily basis if you want to. By keeping close tabs on your funds, you'll always be aware of what's happening in your bank account. For those experienced spenders, this option is far more appealing than the sudden discovery that you're broke!

It's also helpful to watch how much interest you're gathering on investments and savings or what service charges you have incurred.

**Most available benefits**

1. Online banking with key bank is fast, secure, convenient and free.

2. Quick, simple, authenticated access to accounts via the web application.

3. Simply scalable to grow with changing system requirement.

4. Global enterprise wide access to information.

5. Improved data security, restricting unauthorized access.

6. Minimize Storage Space.

1. **Future Look**

The “Banking Online System is a big and ambitious project. I am thankful for being provided this great opportunity to work on it. As already mentioned, this project has gone through extensive research work. On the basis of the research work, we have successfully designed and implemented banking online System. To know what the future of online banking looks like, it’s probably worth looking at the present – online banking isn’t new. When you think of online banking, you probably think about a computer (either a desktop or laptop), a three or four step security process and then an interface that lets you view the balance of your various bank accounts and credit cards, whilst permitting you to transfer money and pay bills. And you’re not wrong either. The most valuable future looks are following below:

* More branches of the bank, maybe it will be international, that means more ATM machines outside.
* Customer issues development based on their needs, so the help desk will be aware of their needs and easy to use.
* Developing a mobile App for banking system that help users to do the obtained his operations without go to the bank only he needs to sign in using his A/C NO. And password and then use your own PIN. Finally the system will update automatically.

**Conclusion**

This project is developed to nurture the needs of a user in a banking sector by embedding all the tasks of transactions taking place in a bank. Future version of this project will still be much enhanced than the current version. Writing and depositing checks are perhaps the most fundamental ways to move money in and out of a checking account, but advancements in technology have added ATM and debit card transactions. All banks have rules about how long it takes to access your deposits, how many debit card transactions you're allowed in a day, and how much cash you can withdraw from an ATM. Access to the balance in your checking account can also be limited by businesses that place holds on your funds.

Banks are providing internet banking services also so that the customers can be attracted. By asking the bank employs we came to know that maximum numbers of internet bank account holders are youth and business man. Online banking is an innovative tool that is fast becoming a necessity. It is a successful strategic weapon for banks to remain profitable in a volatile and competitive marketplace of today. If proper training should be given to customer by the bank employs to open an account will be beneficial secondly the website should be made friendlier from where the customers can directly make and access their accounts.

Thus, the Bank Management System it is developed and executed successfully.

**References**

1. Code for Interview YouTube Channel (https://www.youtube.com/playlist?list=PL5BFcXE899zxVrWaO3Ul6ly2SVJMnJFOr)

2. Online Bank Account Management System

Website: http://www.slideshare.net (Collect some info for report documents)

3. Learning MYSQL, JavaScript, jQuery, PHP, HTML, CSS3,

Website: http://www.w3schools.com

4. PHP and MySQL video tutorials

Website: http://www.freehinditutorial.com, http://www.youtube.com

5. Veneeva, V. (2006), “E-Banking (Online Banking) and Its Role in Today's Society”,

Ezine articles

6. JavaScript validation for empty input field

Website:http://stackoverflow.com/questions/3937513/javascript-validation-for-empty-

input-field ,

7. JavaScript form validation: Validate Password, Validate Email, Validate Phone

Number, http://webcheatsheet.com/javascript/form\_validation.php