PANJAB UNIVERSITY ★ पंजाब यूनिवर्सिटी ★ ਪੰਜਾਬ ਯੂਨੀਵਰਸਿਟੀ ★ 1947 ★

तमसो मा ज्योतिर्गमय

Project Report on

"BANK MANAGEMENT SYSTEM"

Relational Database Management System

B.E *3rd year*

(Branch – IT)

Submitted To:

Mr.Gurpinder singh

Submitted By:

Sumit Thakur

ACKNOWLEDGEMENT

This Project report was completed as a result of support from many people, although not all of them can be mentioned.

We wish to express our sincere gratitude to God for his protection, providence, guidance and above all, for sustaining us.

We are greatly indebted to our good supervisor **Mr.Gurpinder Singh** for his useful and necessary observation, suggestions, contribution and corrections. We would not have been able to achieve anything in this research without your supervision. May God enrich you greatly in every area of life.

Finally, we wish to express our appreciation to our parents for their love and support.

Sumit Thakur

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.

Main Purpose

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

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:

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.

1- Admin login
2- Add/delete/update account
3- Withdrawal/deposit/statements transaction
4- Account Information
5- User details list
6- Active/Inactive account
7- View transaction histories

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.

1- User login, use PIN system
2- Creating/open new account registration
3- Funds transfer (local/international/domestic)
4- View statements transaction
5- User account details
6- Change Password and Pin

Banks terms:

1. All requests received from customers are logged for backend fulfillment and are effective from the time they are recorded at the branch.
2. Rules and regulations applicable to normal banking transactions in India will be applicable mutatis mutandis for the transactions executed through this site.
3. The BAMS Bank service cannot be claimed as a right. The bank may also convert this into a discretionary service anytime.
4. 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.
5. The Bank reserves the right to modify the services offered or the Terms of service of BAMS Bank. The changes will be notified to the customers through a notification on the Site.

Customer's obligations

1. 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.
2. 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.
3. The customer will not attempt or permit others to attempt accessing the BAMS Bank through any unlawful means.

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:

1- You don't have to wait in line.
2- You don't have to plan your day around the bank's hours.
3- 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.

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:

1- More branches of the bank, maybe it will be international, that means more ATM machines outside.
2- Customer issues development based on their needs, so the help desk will be aware of their needs and easy to use.
3- 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.

Reference

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

Source Code

Deposit.java

```
 package bank.management.system;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.util.Date;

public class Deposit extends JFrame implements ActionListener {
    JTextField amount;
    JButton deposit, back;
    String pinnumber;

    Deposit(String pinnumber) {

        this.pinnumber = pinnumber;

        setLayout(null);

        ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("icons/atm.jpg"));
        Image i2 = i1.getImage().getScaledInstance(900, 900, Image.SCALE_DEFAULT);
        ImageIcon i3 = new ImageIcon(i2);
        JLabel image = new JLabel(i3);
        image.setBounds(0, 0, 900, 900);
        add(image);

        JLabel text = new JLabel("Enter the amount you want to deposit");
        text.setForeground(Color.white);
        text.setFont(new Font("System", Font.BOLD, 16));
        text.setBounds(170, 300, 400, 20);
        image.add(text);

        amount = new JTextField();
        amount.setFont(new Font("Raleway", Font.BOLD, 22));
        amount.setBounds(170, 350, 320, 25);
        image.add(amount);

        deposit = new JButton("Deposit");
        deposit.setBounds(355, 485, 150, 30);
        deposit.addActionListener(this);
        image.add(deposit);

        back = new JButton("Back");
        back.setBounds(355, 520, 150, 30);
        back.addActionListener(this);
        image.add(back);

        setSize(900, 900);
        setLocation(300, 0);
        setVisible(true);
    }

    public static void main(String[] args) {
        new Deposit("");
    }

    @Override
    public void actionPerformed(ActionEvent ae) {
        if (ae.getSource() == deposit) {
            String number = amount.getText();
            Date date = new Date();
            if(number.isEmpty()){ //if(number.equals(""))
                JOptionPane.showMessageDialog(null,"Please enter the amount you want to
deposit");
            }else {
                try {
                    Conn conn = new Conn();
                    String query = "insert into bank values('"+pinnumber+"','"+date+"', 'Deposit',
'"+number+"')";
                    conn.s.executeUpdate(query);
                    JOptionPane.showMessageDialog(null,"Rs "+number+" Deposited
Successfully");
                    setVisible(false);
                    new Transactions(pinnumber).setVisible(true);
                } catch (Exception e) {
                    System.out.println(e);
                }
            }
        } else if (ae.getSource() == back) {
            setVisible(false);
           new Transactions(pinnumber).setVisible(true);
        }
    }
}
```

Signup.java

```
 package bank.management.system;

import com.toedter.calendar.JDateChooser;
```

```java
import javax.swing.*;
import java.awt.*;
import java.util.Random;
import java.awt.event.*;

public class SignupOne extends JFrame implements ActionListener {

    long random;
    JTextField nameTextField, fNameTextField, dobTextField, 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 personalDetails = new JLabel("page 1:personal Details");
        personalDetails.setFont((new Font("Raleway",Font.BOLD,22)));
        personalDetails.setBounds(290,80,400,30);
        add(personalDetails);

         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);
```

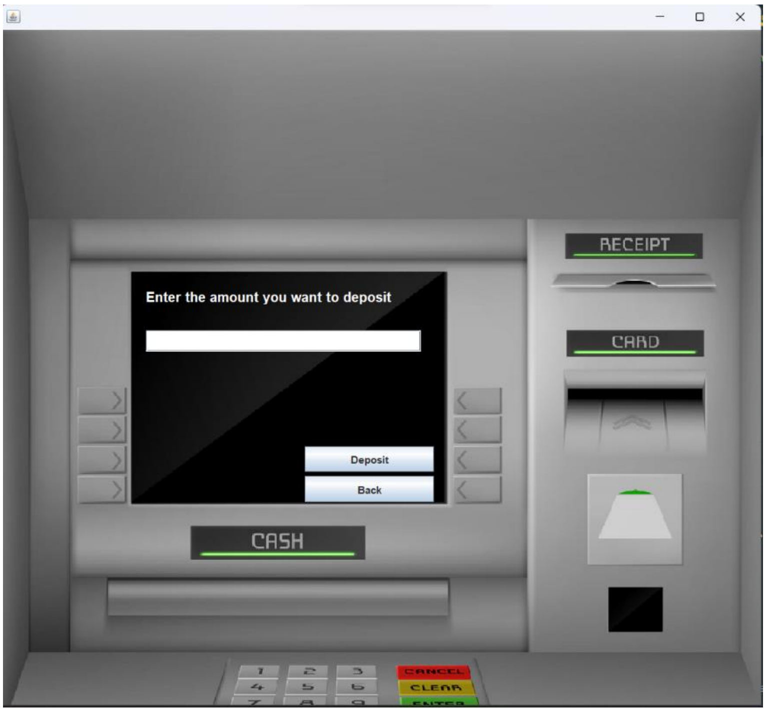
```
JLabel dob = new JLabel("Date of Birth:");
dob.setFont((new Font("Raleway",Font.BOLD,20)));
dob.setBounds(100,240,200,30);
add(dob);

 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:");
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);
```

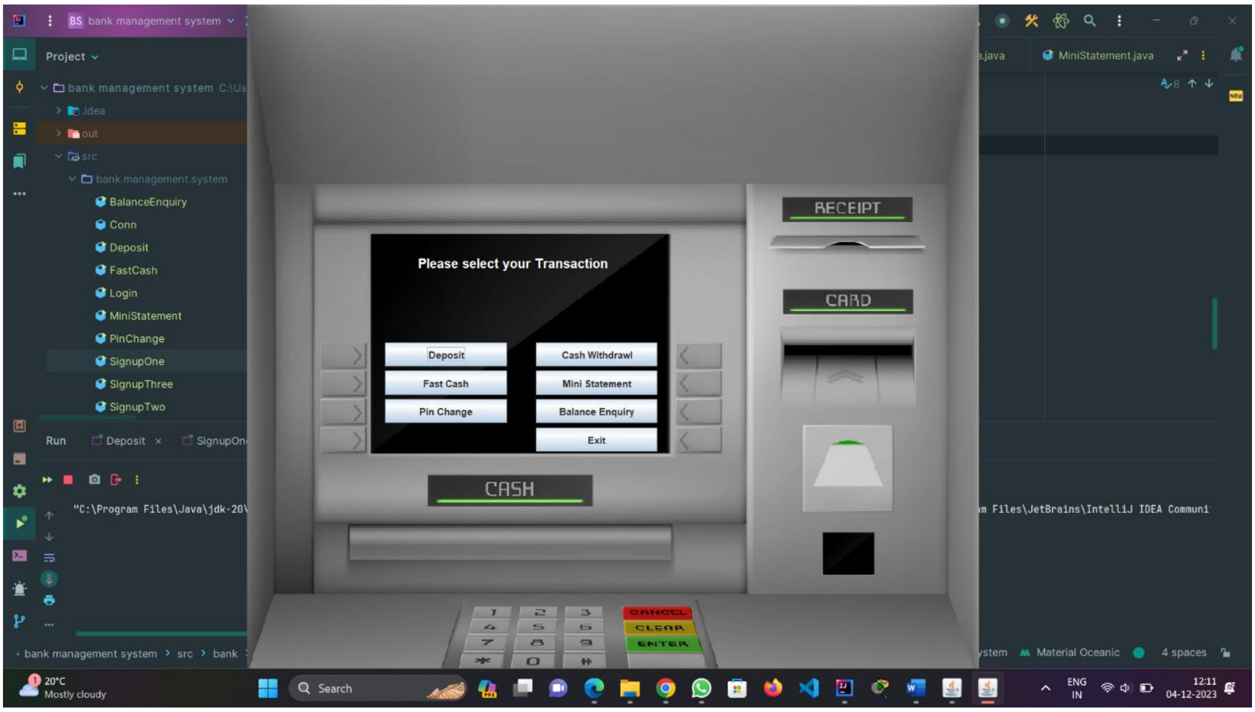
```
unmarried = new JRadioButton("Unmarried");
unmarried.setBounds(450,390,100,30);
unmarried.setBackground(Color.white);
add(unmarried);

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.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 static void main(String[] args) {
    new SignupOne();

}

@Override
public void actionPerformed(ActionEvent ae) {
    String formNo = "" + random; // long
    String name = nameTextField.getText(); // setText
    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 = "others";
    }

    String address = addressTextField.getText();
    String city = cityTextField.getText();
    String state = stateTextField.getText();
    String pinCode = pinCodeTextField.getText();
```

```
        try {
            if (name.equals("")) {
                JOptionPane.showMessageDialog(null, "Name 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("").setVisible(true);
            }
        } catch (Exception e) {
            System.out.println(e);
        }
      }
    }
```

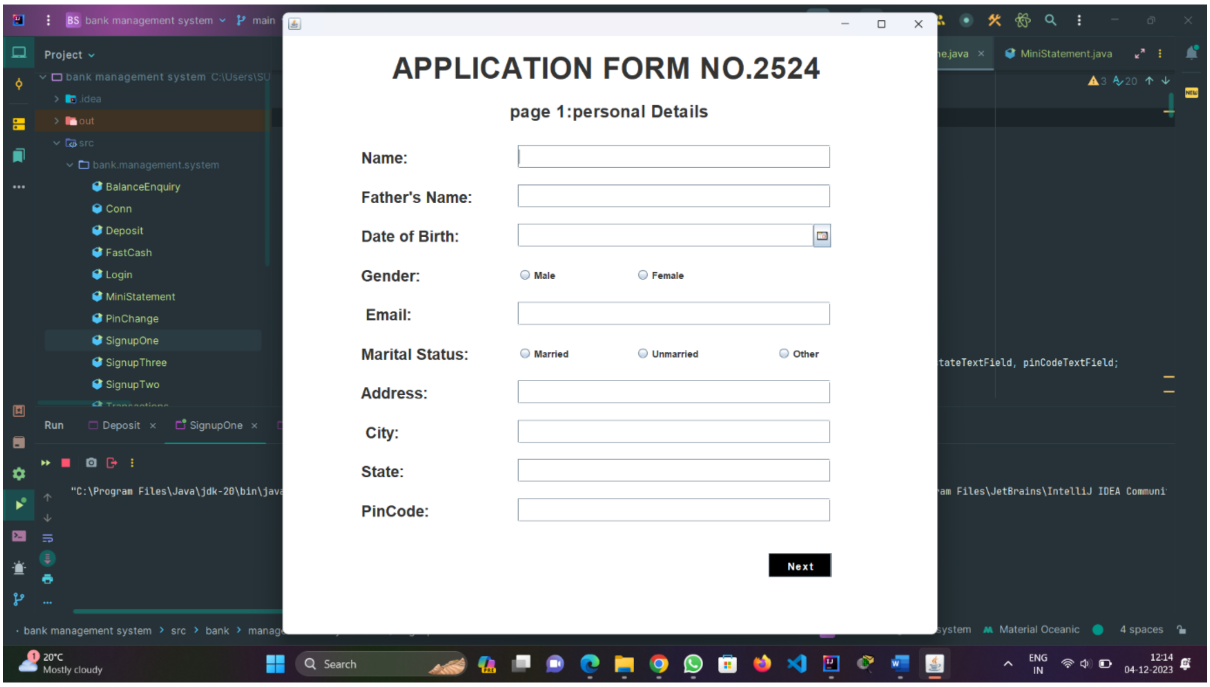
Screenshots





```
"C:\Program Files\Java\jdk-20\
```

```
n Files\JetBrains\IntelliJ IDEA Communi
```



```
tateTextField, pinCodeTextField;
```

```
"C:\Program Files\Java\jdk-20\bin\java
```

```
am Files\JetBrains\IntelliJ IDEA Communi
```

APPLICATION FORM NO.2524

page 1:personal Details

Name:	
Father's Name:	
Date of Birth:	
Gender:	○ Male ○ Female
Email:	
Marital Status:	○ Married ○ Unmarried ○ Other
Address:	
City:	
State:	
PinCode:	

Next