

GOVERNMENT POLYTECHNIC COLLEGE
MUTTOM- THODUPUZHA



Report on

BANK MANAGEMENT SYSTEM

Submitted By

ALOK KUMAR DEBNATH

FOURTH SEMESTER COMPUTER ENGINEERING

Register No. 19138128

DEPARTMENT OF COMPUTER ENGINEERING

2020-2021

GOVERNMENT POLYTECHNIC COLLEGE
MUTTOM- THODUPUZHA



DEPARTMENT OF COMPUTER ENGINEERING

CERTIFICATE

This is to certify that the project entitled “BANK MANAGEMENT SYSTEM” is a Bona fide record of work done by ALOK KUMAR DEBNATH, Reg. No: 19138128 of 4th semester Computer Engineering for the partial fulfilment of requirements for the award of “DIPLOMA IN COMPUTER ENGINEERING” under the Department of Technical Education, Government of Kerala during the academic year 2020-2021.

HEAD OF THE DEPARTMENT

GUIDE IN CHARGE

EXTERNAL EXAMINER

INTERNAL EXAMINER

ACKNOWLEDGEMENT

I wish to extend heartfelt thanks to everyone who helped in providing basic knowledge for completing this project. I would like to give the first honour to MODERN INTERNET which gave the wisdom and knowledge to complete the mini project.

I wish to extend my sincere gratitude to Mrs. GEETHA DEVI, Principal and Mrs. SAJINA K PAULOSE, Head of the Department of Computer Engineering for giving me the opportunity to complete this seminar successfully.

I wish to extend my sincere thanks to my seminar guide Mr. JOSE JAMES for his valuable support and guidance. I thank him for his timely suggestion and constant encouragement that boosted up my morale and led to the accomplishment of this seminar.

ALOK KUMAR DEBNATH

INDEX

Content	Page No:
INTRODUCTION	4
ABSTRACT	5
SYSTEM ANALYSIS AND DESIGN	6

FUNCTION OVERVIEW	7
DATA DICTIONARY	8
SOURCE CODE	9
SCREENSHOTS	28
CONCLUSION	33
REFERENCE	33

INTRODUCTION

This project BANK MANAGEMENT SYSTEM is aimed to develop an app aimed for bank employees for the ease of managing Account holder details and doing online transactions on behalf of Account holder. The UI and ease of usage was the top priority, even if that was the top priority the functionality was not ignored and there are all the features I can currently think of. The database connectivity is planned using the SQL connection methodology. This system is developed on Java platform and supported by a MySQL database to store bank and user specific details.

ABSTRACT

This project BANK MANAGEMENT SYSTEM is aimed to develop an app aimed for bank employees for the ease of managing Account holder details and doing online transactions on behalf of Account holder. The UI and ease of usage was the top priority, even if that was the top priority the functionality was not ignored and there are all the features I can currently think of. The database connectivity is planned using the SQL connection methodology. The application takes care of different modules and their associated reports, which are produced as per the applicable strategies and standards that are put forward by the administrative staff. The entire project has been developed keeping in view of the distributed client server computing technology.

SYSTEM ANALYSIS AND DESIGN

Requirement analysis is the process of identifying the system requirements through the observations of the existing system, discussion with the potential user's procedures, task analysis etc. Requirement's analysis results in the specifications of software operational characteristics indicates software's interface with other elements and establishes constraints that software must need. A project requirements analysis removes all ambiguities and inconsistencies from the initial user perception of the problem.

HARDWARE AND SOFTWARE REQUIREMENTS

SOFTWARE REQUIREMENTS

OPERATING SYSTEM: WINDOWS/UBUNTU

LANGUAGE : JAVA

FONTEND : JAVA

BACKEND : MYSQL

DEVELOPMENT TOOL: Net Beans ide 8.2 HARDWARE

REQUIREMENTS

PROCESSOR: INTEL PENTIUM CORE 8

MAIN MEMORY: 1GB RAM

SECONDARY MEMORY 500 GB HARDISK

VIRTUAL DISPLAY UNIT RESOLUTION OF 1024X76

APPLICATION & USER

There is only one type of user for this application, who has all privileges to make changes to the settings of this application and on the data.

FUNCTION OVERVIEW

REGISTER / LOGIN

Employees who have permission to manage user data and transactions, have to register by providing their name, email, branch, password. If user is already registered then they can login by providing the email and password. These details will be stored in database.

ACCOUNT DETAILS

Employee or Admin can manage basic account holder details from here. Every single change will be logged in the system for security and safety reasons. Here new account data can be added, edited or even be deleted.

TRANSACTIONS

Transactions such as Withdraw/Deposit or Fund transfer between accounts can be done. The process is as simple as possible so that any new employee can also get accustomed to the whole process very easily and effortlessly.

DATA DICTIONARY

REGISTRATION DETAILS DATABASE

Field name	Data type
name	varchar
email	varchar
branch	varchar
Password	varchar

LOGIN DETAILS

Field name	Data type
email	varchar
password	varchar

CUSTOMER DETAILS

Field name	Data type
Account_no	varchar
name	varchar
balance	varchar

LOG

Field name	Data type
log	varchar

SOURCE CODE

Fetch data from database to table

```
public void fetch()
{
    try{
        Class.forName("com.mysql.jdbc.Driver");

        Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/db_internetbanking","root","");

        String sql="select * from tbl_customer_details";
        System.out.println(sql);
        PreparedStatement pst = con.prepareStatement(sql);
        ResultSet rs = pst.executeQuery();
        DefaultTableModel dl=(DefaultTableModel)jTable_account2.getModel();
        dl.setRowCount(0);

        while(rs.next())
        {
            Vector v2=new Vector();

            v2.add(rs.getString("id"));
            v2.add(rs.getString("account_no"));
            v2.add(rs.getString("name"));
            v2.add(rs.getString("balance"));

            dl.addRow(v2);
        }
    }
    catch(Exception e)
    {
        System.out.println(e);
    }
}
```

Insert new account holder data into database

```
String ac_no=jTextField_acc_no_account2.getText();

System.out.println(ac_no);

String name=jTextField_name_account2.getText();

System.out.println(name);


try{

    Class.forName("com.mysql.jdbc.Driver");

    Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/db_internetbanking","root","");

    String sql = "insert into
tbl_customer_details(account_no,name,balance)values('"+ac_no+"','"+name+"','0')";

    System.out.println(sql);

    Statement stmt=con.createStatement();

    int rs=stmt.executeUpdate(sql);

    System.out.println(rs);

    if(rs==1)
    {

        JOptionPane.showMessageDialog(this, "Added Successfully");


        //log

        String log_statement = "A/c added where A/c number = " + ac_no + " and name = " + name;

        String sql_log = "insert into
tbl_log_account(log,user)values('"+log_statement+"','employee')";

        System.out.println(sql_log);

        Statement stmt_log=con.createStatement();

        int rs_log=stmt_log.executeUpdate(sql_log);

        System.out.println(rs_log);

        //log end


        fetch();

        clear();

    }

}
```

```

    }

}
catch(Exception e)
{
    System.out.println(e);
}

```

Update Account holder data

```

DefaultTableModel dl=(DefaultTableModel)jTable_account2.getModel();

int index=jTable_account2.getSelectedRow();

int id=Integer.parseInt(dl.getValueAt(index,0).toString());


String ac_no=jTextField_acc_no_account2.getText();
System.out.println(ac_no);

String name=jTextField_name_account2.getText();
System.out.println(name);


try{

    Class.forName("com.mysql.jdbc.Driver");

    Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/db_internetbanking","root","");


    //log

    String updatelog = "select * from tbl_customer_details where id='"+id+"'";


    System.out.println(updatelog);

    PreparedStatement pst2 = con.prepareStatement(updatelog);

    ResultSet logg;

    logg = pst2.executeQuery();

```

```

System.out.println(logg);
if(logg.next())
{
    String account = logg.getString("account_no");
    String customername = logg.getString("name");
    System.out.println(account);
    System.out.println(customername);

    System.out.println(!ac_no.equals(account));
    System.out.println(!name.equals(customername));
    if(!ac_no.equals(account))
    {
        String log_statement = "A/c number changed from " + account + " to " + ac_no + " where
id = " + id;

        String sql_log = "insert into
tbl_log_account(log,user)values('"+log_statement+"','employee')";

        System.out.println(sql_log);

        Statement stmt_log=con.createStatement();
        int rs_log=stmt_log.executeUpdate(sql_log);
        System.out.println(rs_log);
    }
    if(!name.equals(customername))
    {
        String log_statement = "Name changed from " + customername + " to " + name + " where
A/c number = " + ac_no+ " and id = " + id;

        String sql_log = "insert into
tbl_log_account(log,user)values('"+log_statement+"','employee')";

        System.out.println(sql_log);

        Statement stmt_log=con.createStatement();
        int rs_log=stmt_log.executeUpdate(sql_log);
        System.out.println(rs_log);
    }
}

```

```

    }

    //log end

    String sql = "update tbl_customer_details set account_no='"+ac_no+"',name='"+name+"'
where id='"+id+"'";

    System.out.println(sql);

    Statement stmt=con.createStatement();

    int rs=stmt.executeUpdate(sql);

    System.out.println(rs);

    if(rs==1)
    {
        JOptionPane.showMessageDialog(this, "Updated Successfully");

        fetch();

        clear();
    }

}

catch(Exception e)

{

    System.out.println(e);

}

```

Delete account holder data

```

DefaultTableModel dl=(DefaultTableModel)jTable_account2.getModel();

int index=jTable_account2.getSelectedRow();

int id=Integer.parseInt(dl.getValueAt(index,0).toString());

//for log

String ac_no=jTextField_acc_no_account2.getText();

System.out.println(ac_no);

```

```

String name=jTextField_name_account2.getText();

System.out.println(name);

//

int dialogResult = JOptionPane.showConfirmDialog(null,"Do you want to
delete","warning",JOptionPane.YES_NO_OPTION);

if(dialogResult==JOptionPane.YES_OPTION)
{
    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/db_internetbanking","root","");

        String sql="delete from tbl_customer_details where id = '"+id+"'";
        System.out.println(sql);

        Statement stmt=con.createStatement();

        int rs=stmt.executeUpdate(sql);

        System.out.println(rs);

        if(rs==1){

            JOptionPane.showMessageDialog(this, "Deleted Successfully");


            //log

            String log_statement = "A/c deleted where A/c number = " + ac_no + " and id = " + id;

            String sql_log = "insert into
tbl_log_account(log,user)values('"+log_statement+"','employee')";

            System.out.println(sql_log);

            Statement stmt_log=con.createStatement();

            int rs_log=stmt_log.executeUpdate(sql_log);

            System.out.println(rs_log);

            //log end


            fetch();

            clear();

```

```

    }
}
catch(Exception e)
{
    System.out.println(e);
}
}

```

Deposit money

```

String account_no=jTextField_accno.getText();

System.out.println(account_no);

String amount=jTextField_amount.getText();

System.out.println(amount);

if(amount.equals(""))
{
    jLabel_error_message.setText("Please enter an amount");
    //show jpanel message
    timerDown2.start();
}

try{

    Class.forName("com.mysql.jdbc.Driver");

    Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/db_internetbanking","root","");

    String sql = "select balance from tbl_customer_details where account_no='"+account_no+"'";

    System.out.println(sql);

    PreparedStatement pst = con.prepareStatement(sql);

    ResultSet rs = pst.executeQuery();

    System.out.println(rs);
}

```

```

if(rs.next())
{
    String am = rs.getString("balance");
    Integer f1=Integer.parseInt(amount);
    Integer f2=Integer.parseInt(am);
    Integer f3=f2+f1;
    String s=String.valueOf(f3);
    System.out.println(s);

    String sq = "update tbl_customer_details set balance='"+s+"' where
account_no='"+account_no+"'";
    System.out.println(sq);
    Statement stmt=con.createStatement();
    int rs2=stmt.executeUpdate(sq);
    System.out.println(rs2);
    if(rs2==1)
    {
        //log
        String log = "Rs. "+amount+" deposited to A/c "+account_no;
        String sql_log = "insert into tbl_log(log)values('"+log+"')";
        System.out.println(sql_log);
        Statement stmt_log=con.createStatement();
        int rs_log=stmt_log.executeUpdate(sql_log);
        System.out.println(rs_log);
        /////

        JOptionPane.showMessageDialog(this, "Deposit Successfully");
        clear2();

        // hide the jpanel message
        timerUp.start();
    }
}

```



```

        // hide the jpanel message
        timerUp2.start();
    }
}
else
{
    //for data validation
    if(account_no.equals(""))
    {
        jLabel_error_message.setText("Please enter an account number");
        //show jpanel message
        timerDown2.start();
    }
    /*else if(amount.equals(""))
    {
        jLabel_error_message.setText("Please enter an amount");
        //show jpanel message
        timerDown2.start();
    }
    */
    else
    {
        jLabel_error_message.setText("Incorrect account number");
        //show jpanel message
        timerDown2.start();
    }
    //////////////////////////////////////
}

}

catch(Exception e)
{
    System.out.println(e);
}

```

Withdraw money

```
String account_no=jTextField_accno.getText();

    System.out.println(account_no);

    String amount=jTextField_amount.getText();

    int a = Integer.parseInt(amount);

    System.out.println(amount);


    if(amount.equals(""))
    {
        jLabel_error_message.setText("Please enter an amount");

        //show jpanel message

        timerDown2.start();
    }


    try{

        Class.forName("com.mysql.jdbc.Driver");

        Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/db_internetbanking","root","");

        String sql = "select balance from tbl_customer_details where account_no='"+account_no+"'";

        System.out.println(sql);

        PreparedStatement pst = con.prepareStatement(sql);

        ResultSet rs = pst.executeQuery();

        System.out.println(rs);

        if(rs.next())
        {

            String am = rs.getString("balance");

            int b = Integer.parseInt(am);
```

```

        if(a < b){
            Integer f1=Integer.parseInt(amount);
            Integer f2=Integer.parseInt(am);
            Integer f3=f2-f1;
            String s=String.valueOf(f3);
            System.out.println(s);

            String sq = "update tbl_customer_details set balance='"+s+"' where
account_no='"+account_no+"'";
            System.out.println(sq);
            Statement stmt=con.createStatement();
            int rs2=stmt.executeUpdate(sq);
            System.out.println(rs2);
            if(rs2==1)
            {
                //log
                String log = "Rs. "+amount+" withdrawn from A/c "+account_no;
                String sql_log = "insert into tbl_log(log)values('"+log+"')";
                System.out.println(sql_log);
                Statement stmt_log=con.createStatement();
                int rs_log=stmt_log.executeUpdate(sql_log);
                System.out.println(rs_log);
                /////

                JOptionPane.showMessageDialog(this, "Withdrawal Successfully");
                clear2();

                // hide the jpanel message
                timerUp.start();
                // hide the jpanel message

```

```

        timerUp2.start();
    }
    }else{
        //JOptionPane.showMessageDialog(this, "Amount exceeds available balance");
        jLabel_error_message.setText("Amount exceeds maximum withdrawal limit for this
account");
        //show jpanel message
        timerDown2.start();
    }
}
else
{
    //for data validation
    if(account_no.equals(""))
    {
        jLabel_error_message.setText("Please enter an account number");
        //show jpanel message
        timerDown2.start();
    }/*else if(amount.equals(""))
    {
        jLabel_error_message.setText("Please enter an amount");
        //show jpanel message
        timerDown2.start();
    }*/else
    {
        jLabel_error_message.setText("Incorrect account number");
        //show jpanel message
        timerDown2.start();
    }
    ///////////////////////////////////
}

```

```

    }

    catch(Exception e)

    {

        System.out.println(e);

    }

```

Fund transfer

```

String from=jTextField_from.getText();

    System.out.println(from);

    String to=jTextField_to.getText();

    System.out.println(to);

    String amount=jTextField_amount2.getText();

    int amou = Integer.parseInt(amount);

    System.out.println(amount);


    try{

        Class.forName("com.mysql.jdbc.Driver");

        Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/db_internetbanking","root","");

        String sql = "select * from tbl_customer_details where account_no='"+from+"'";

        String sql2 = "select * from tbl_customer_details where account_no='"+to+"'";


        System.out.println(sql);

        PreparedStatement pst = con.prepareStatement(sql);

        ResultSet rs = pst.executeQuery();

        System.out.println(rs);


        System.out.println(sql2);

        PreparedStatement pst2 = con.prepareStatement(sql2);

        ResultSet rs2 = pst2.executeQuery();

        System.out.println(rs2);

```

```

if(rs.next() && rs2.next())// && amount.equals("")
{
    String bal1 = rs.getString("balance");
    int bala = Integer.parseInt(bal1);

    if(amou < bala){
        Integer f1=Integer.parseInt(bal1);
        Integer f2=Integer.parseInt(amount);
        Integer f3=f1-f2;
        String s=String.valueOf(f3);
        System.out.println(s);

        String sq = "update tbl_customer_details set balance='"+s+"' where
account_no='"+from+"'";
        System.out.println(sq);
        Statement stmt=con.createStatement();
        int rs3=stmt.executeUpdate(sq);
        System.out.println(rs3);

        //////////////////////////////////////

        String bal2 = rs2.getString("balance");
        Integer g1=Integer.parseInt(bal2);
        Integer g2=Integer.parseInt(amount);
        Integer a=g1+g2;
        String s2=String.valueOf(a);
        System.out.println(s2);

        String sq2 = "update tbl_customer_details set balance='"+s2+"' where
account_no='"+to+"'";
        System.out.println(sq2);

```

```

Statement stmt2=con.createStatement();

int rs4=stmt2.executeUpdate(sq2);

System.out.println(rs4);

////////////////////////////////////
//log

String log = "Rs. "+amount+" transfered from A/c "+from+" to A/c "+to;
String sql_log = "insert into tbl_log(log)values('"+log+"')";
System.out.println(sql_log);
Statement stmt_log=con.createStatement();
int rs_log=stmt_log.executeUpdate(sql_log);
System.out.println(rs_log);

JOptionPane.showMessageDialog(this, "Fund Transfer Successfully");
// hide the jpanel message
timerUp3_transfer.start();
clear3();
}else{
    //JOptionPane.showMessageDialog(this, "Amount exceeds available balance");
    jLabel_error_message2.setText("Insufficient balance");
    //show jpanel message
    timerDown3_transfer.start();
}

}else {
    jLabel_error_message2.setText("Please enter Correct account number");
    //show jpanel message
    timerDown3_transfer.start();
}
}

```

```

catch(Exception e)
{
    System.out.println(e);
}

//for data validation
if(from.equals("") && to.equals(""))
{
    jLabel_error_message2.setText("Please enter account number");
    //show jpanel message
    timerDown3_transfer.start();
}else if(from.equals(""))
{
    jLabel_error_message2.setText("Please enter Sender account number");
    //show jpanel message
    timerDown3_transfer.start();
}else if(to.equals(""))
{
    jLabel_error_message2.setText("Please enter Payee account number");
    //show jpanel message
    timerDown3_transfer.start();
}else if(amount.equals(""))
{
    jLabel_error_message2.setText("Please enter an amount to transfer");
    //show jpanel message
    timerDown3_transfer.start();
}

```

[Login page](#)

// TODO add your handling code here:

```

String email=jTextField_email.getText().trim();

System.out.println(email);

```



```

String password = String.valueOf(jPasswordField1.getPassword()).trim();

System.out.println(password);

if(email.equals(""))
{
    jLabel_error_message.setText("Please enter your email");
}else if(password.equals(""))
{
    jLabel_error_message.setText("Please enter your password");
}else
{
    jLabel_error_message.setText("Incorrect email or password");
}

try {
    Class.forName("com.mysql.jdbc.Driver");

    Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/db_internetbanking","root","");

    String sql="select * from tbl_reg where email = '"+email+"' && password = '"+password+"' ";

    System.out.println(sql);

    Statement stmt=con.createStatement();

    ResultSet rs=stmt.executeQuery(sql);

    //System.out.println(rs);

    if(rs.next()){

        JOptionPane.showMessageDialog(null, "Login Successful");

        this.dispose();

        dashboard d=new dashboard();

```

```

        d.setVisible(true);
    }
} catch (Exception e) {
    System.out.println("Error "+e);
}

```

```

//show jpanel message
timerDown.start();
}

```

Registration page

```

String name=jTextField_name.getText();
    System.out.println(name);

String email=jTextField_email.getText();
    System.out.println(email);

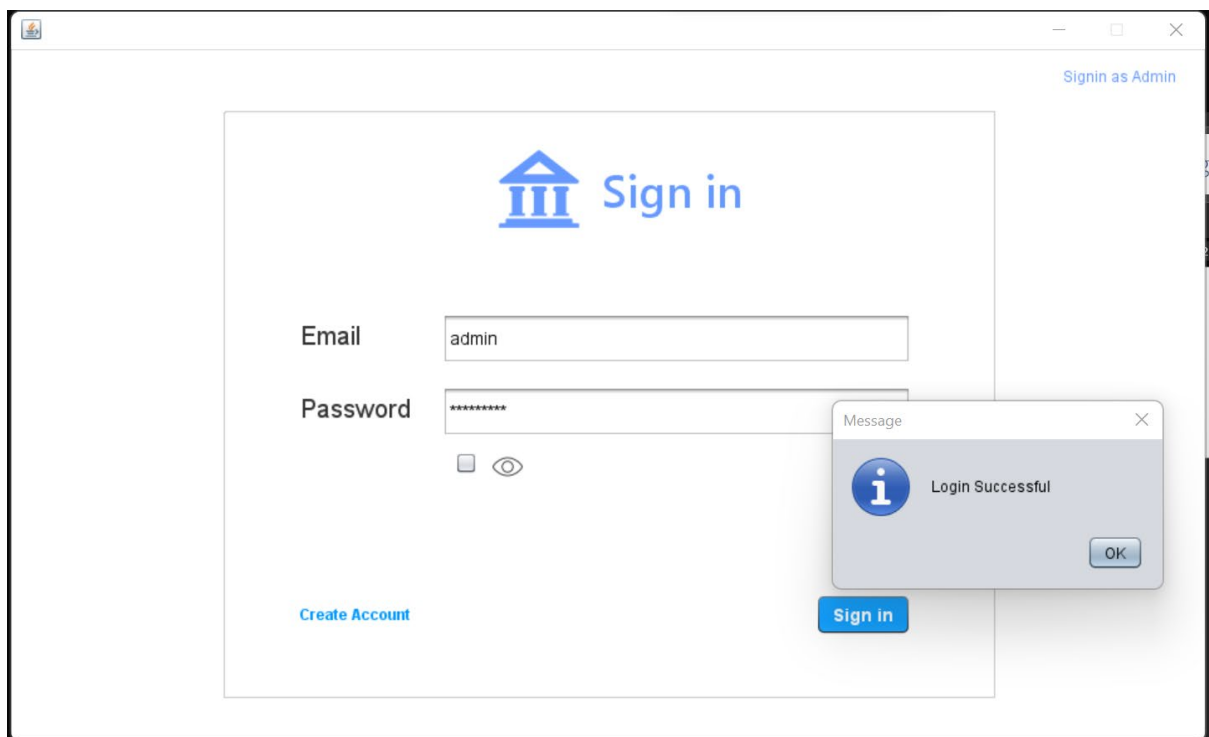
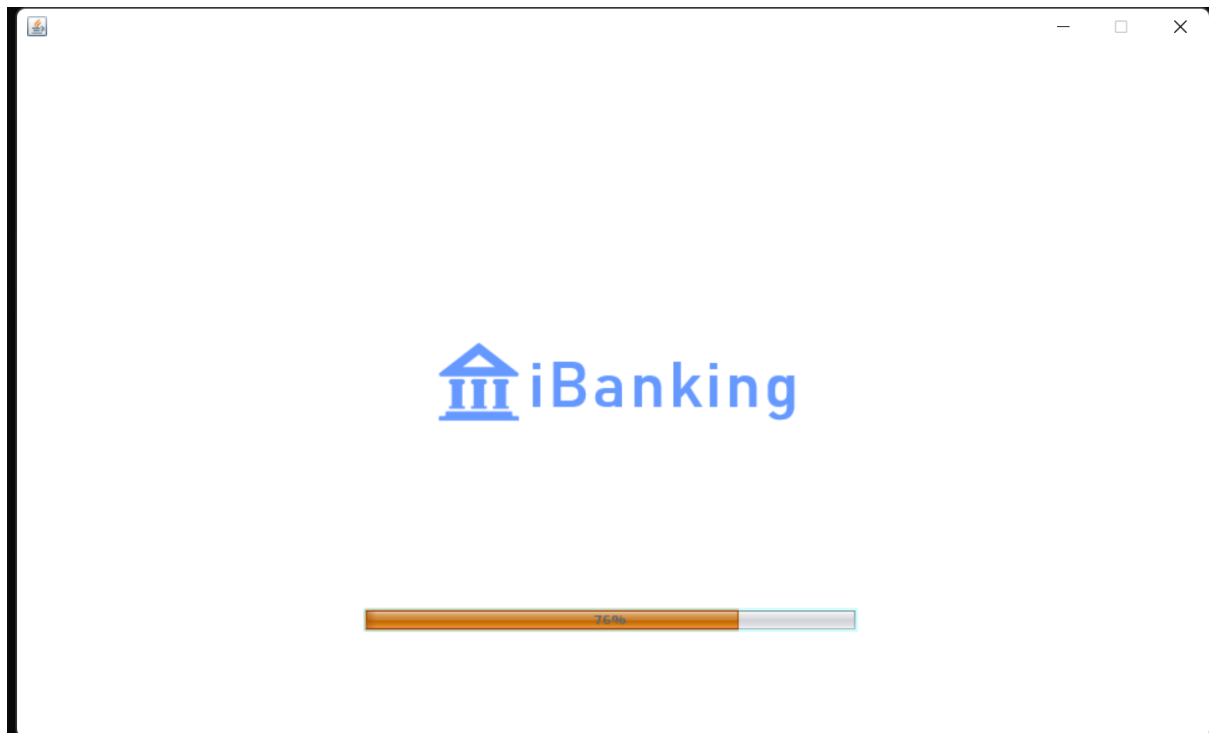
String branch=jTextField_branch.getText();
    System.out.println(branch);

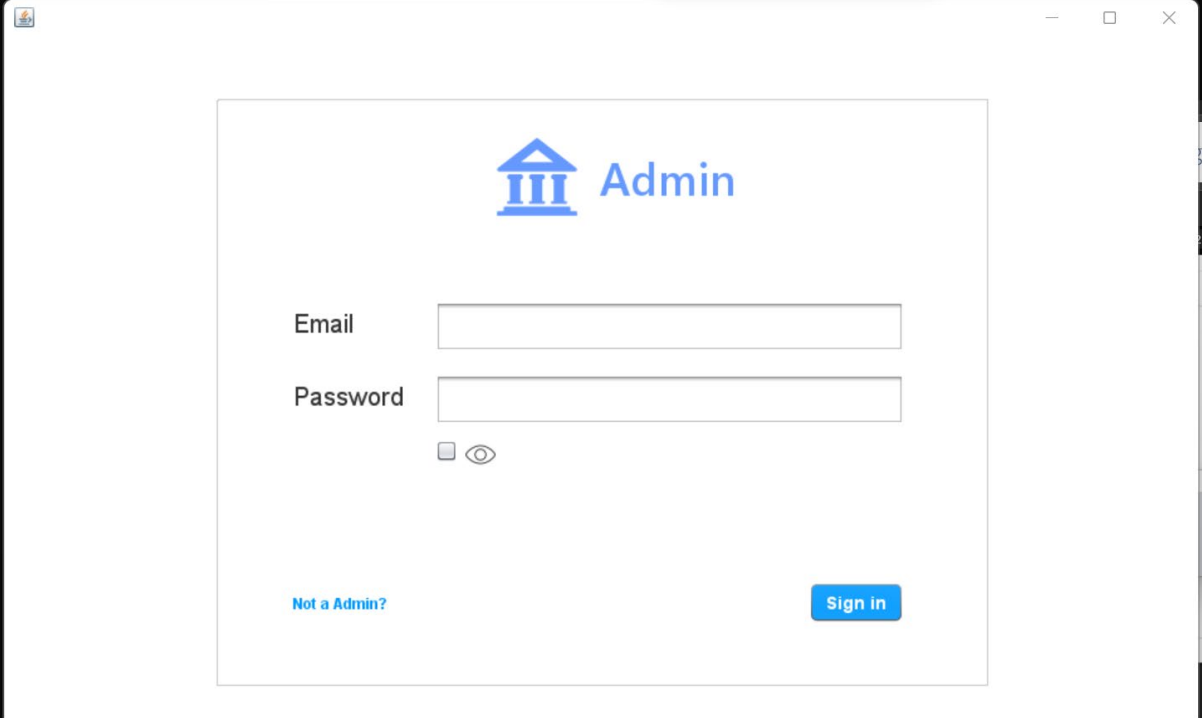
String password=jPasswordField1.getText();
    System.out.println(password);
//Giving database connection below
try {
    Class.forName("com.mysql.jdbc.Driver");
    Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/db_internetbanking","root","");
    String sql="insert into tbl_reg(name,email,branch,password,position)
values('"+name+"','"+email+"','"+branch+"','"+password+"','")";
    System.out.println(sql);
    Statement stmt=con.createStatement();

```

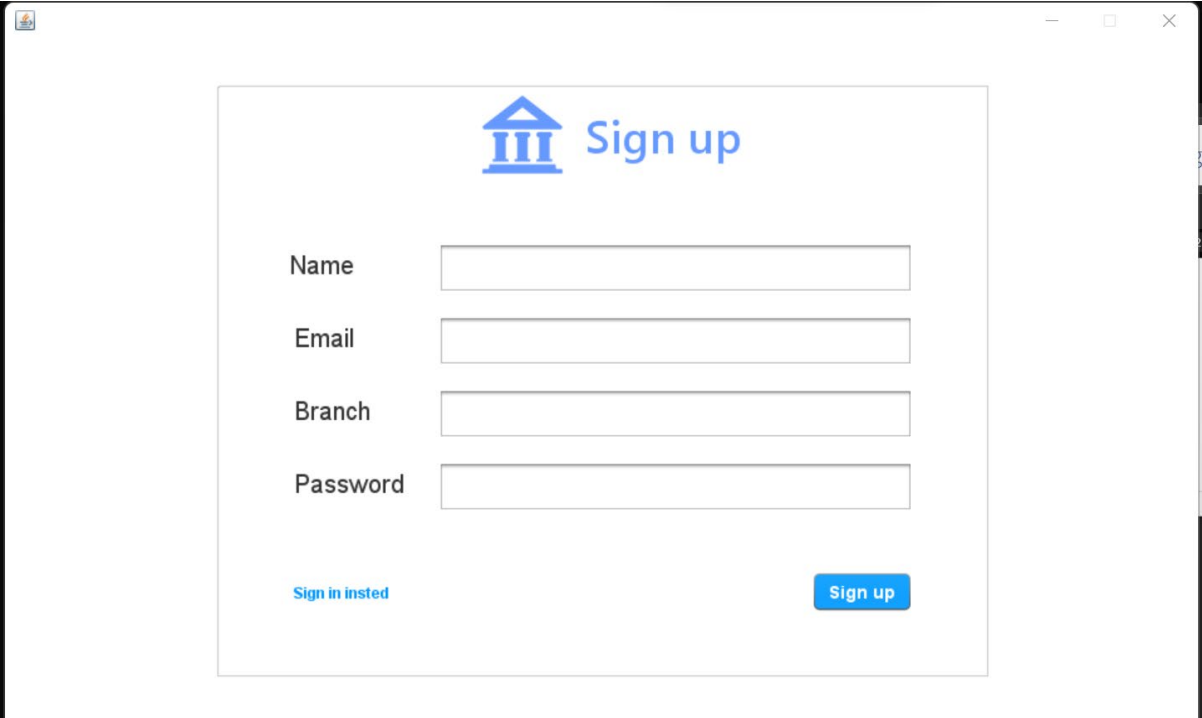
```
int rs=stmt.executeUpdate(sql);  
System.out.println(rs);  
if(rs==1){  
    JOptionPane.showMessageDialog(null, "Registration Successful");  
    login l=new login();  
    l.setVisible(true);  
    dispose();  
}  
} catch (Exception e) {  
    System.out.println(e);  
}
```

SCREENSHOTS






The image shows a web browser window with a login form titled "Admin". The form is centered on a light gray background. It features a blue icon of a classical building with three columns to the left of the title "Admin". Below the title, there are two input fields: "Email" and "Password". The "Password" field has a small eye icon to its right, which is currently closed. At the bottom left of the form, there is a link that says "Not a Admin?". At the bottom right, there is a blue button labeled "Sign in".



The image shows a web browser window with a sign-up form titled "Sign up". The form is centered on a light gray background. It features a blue icon of a classical building with three columns to the left of the title "Sign up". Below the title, there are four input fields: "Name", "Email", "Branch", and "Password". At the bottom left of the form, there is a link that says "Sign in insted". At the bottom right, there is a blue button labeled "Sign up".

EMPLOYEE

Logout

ACCOUNT DETAILS

WITHDRAW / DEPOSIT

FUND TRANSFER

Id	Account number	Name	Balance
1	2126253251	Albert	400
2	2147483648	Jhon	7501
3	2134683647	Sham	6650
5	2133573634	Robert	6500
6	1246794945	William	7650

Account Number


Name

Update

Insert

Delete

Refresh

EMPLOYEE

Logout

ACCOUNT DETAILS

WITHDRAW / DEPOSIT

FUND TRANSFER

Account no

Amount

Check Account Details

Deposit

Withdraw

iBanking
EMPLOYEE

Logout

ACCOUNT DETAILS

WITHDRAW / DEPOSIT

FUND TRANSFER

From A/c

To A/c

↔

Amount

Transfer

iBanking
ADMIN

Logout

ACCOUNT DETAILS

EMPLOYEE DETAILS

LOG HISTORY

Id	Account number	Name	Balance
1	2126253251	Albert	400
2	2147483648	Jhon	7501
3	2134683647	Sham	6650
5	2133573634	Robert	6500
6	1246794945	William	7650

Account Number

Name


Balance


Update


Insert


Delete

Refresh


iBanking
ADMIN
Logout


ACCOUNT DETAILS


EMPLOYEE DETAILS


LOG HISTORY

Id	Name	Position	Branch
1	admin	Branch Manager	New Delhi
3	Raj	Loan officer	New Delhi
4	Jorge	Staff	New Delhi
5	Thomas	Staff	Mumbai
7	Jobin	Staff	Mumbai
8	Alfred	Staff	New Delhi

Name

Position


Branch


Update

Insert


Delete

Refresh


iBanking
ADMIN
Logout


ACCOUNT DETAILS


EMPLOYEE DETAILS


LOG HISTORY

Transaction log

OK

1	Transferred from A/c 2126253250 to A/c 2147483647
2	Rs. 500 deposited to A/c 2126253250
3	Rs. 50 withdrawn from A/c 2147483648
4	Rs. 1000 transfered from A/c 2126253251 to A/c 2147483648
5	Rs. 200 deposited to A/c 2147483648
6	Rs. 6951 withdrawn from A/c 2126253251
7	Rs. 6665 withdrawn from A/c 2126253251
8	Rs. 1000 transfered from A/c 2147483648 to A/c 2126253251
9	Rs. 1 transfered from A/c 2126253251 to A/c 2147483648
10	Rs. 100 withdrawn from A/c 2126253251
11	Rs. 100 transfered from A/c 2126253251 to A/c 2147483648
12	Rs. 400 transfered from A/c 2126253251 to A/c 2147483648

Refresh

CONCLUSION

This project is developed for creating a software that can be use by employees and admin within a bank. The purpose of this application is to ease the process used at bank and to simply and make the UI up to the modern standards.

REFERENCE

For solving common errors: <https://stackoverflow.com/>

My UI inspiration: <https://www.codewithc.com/online-bank-management-system-project-java/>

Some UI interactions: https://www.youtube.com/watch?v=eSM_YkWeS7k