**Project: Secure Online Banking System: Name: Karthik Mohan**

**Pavan Nadkarni**

The features of the User Account in this Secure online banking system consists the features such as the

account :

1) **Registration New account:** This is the feature by which a customer can create new account if he

wants to ,in the bank through online.

2) **Balance:** This feature in the online banking system allows the customer to view his account balance

that is the money remaining in the bank .

3) **Profile:** This contains the details of the customer such as name, address and mobile number.

4) **Transfer Money :**This feature allows the customer to transfer money to whichever bank account he

wants. The details of the transaction taken place are shown in this page. Further the customer needs

OTP(one time password) to make any money transfer. By this the transaction becomes secure.

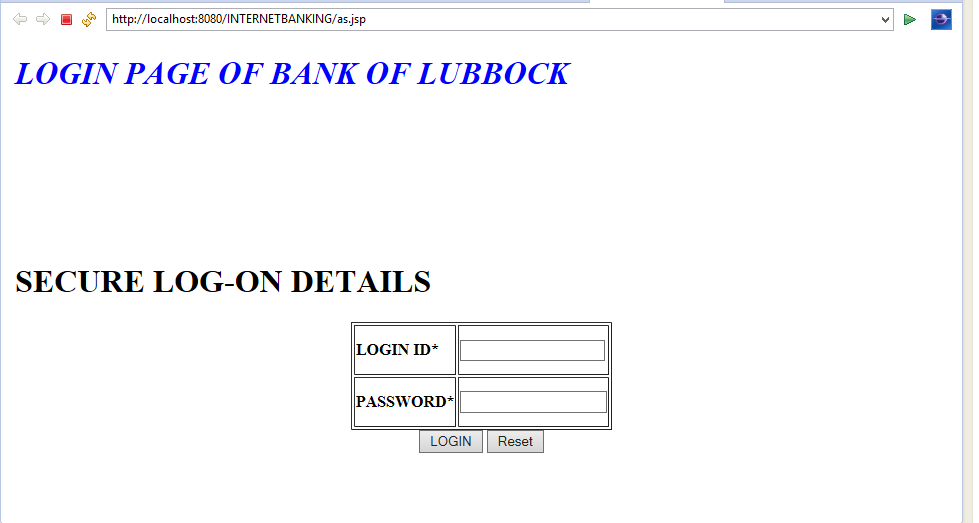
**5) Account Statement :** This feature allows the customer to check his past transactions which were in

the past 30days , so that the customer has a view his activities.

**6) Home:** This feature allows the customer to get back to the starting page of the banking system from

whichever page he is on the online banking system.

1.Intialy Secure Online Banking System displays Login page for user



Upon clicking login button, jsp page action calls Program1.java where if the below method executed succesfully **if**(userAction.equals("LOGIN")){

searchdetails(req,res);

}

it calls the methods searchdetails(req,res);

in searchdetails() method it gets the parameter from jsp web page using getparameter method as shown below:

String logid=req.getParameter("logid");

String password=req.getParameter("pass");

**Adding security feature:**

We make it secure by Encrypting password using 128 bit key as shown below.

//12bit key

String key = "TTU123car123secr"; // 128 bit key

//create key and cipher

Key aesKey = **new** SecretKeySpec(key.getBytes(), "AES");

Cipher cipher = Cipher.*getInstance*("AES");

//encrypting the entered password

cipher.init(Cipher.*ENCRYPT\_MODE*, aesKey);

**byte**[] encrypted = cipher.doFinal(password.getBytes());//encrypting pwd

System.*err*.println(**new** String(encrypted));

then **it establishes connection with MySql** database by using below code where userlogin is our database

**private** **void** estabcon()

{

**try**{

//access database .

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

con =DriverManager.*getConnection*("jdbc:mysql://localhost/userlogin","root","root");

3)Once the connection is established it retrieves the username and password by using below method from DetailsAccess class.

retrieve(string id,string pass)

ArrayList personlist=DetailsAccess.retrieve(logid,password);

HttpSession session=req.getSession(**true**);

session.setAttribute("ATT", personlist);

RequestDispatcher rd=**null**;

then it **decrypts** the password using the same key as shown below.

cipher.init(Cipher.*DECRYPT\_MODE*, aesKey);

String password1 = **new** String(cipher.doFinal(encrypted));//decrypted pwd

System.*err*.println(password);

then if the entered username and password doesn't match

**if**(personlist.size()==0){

req.setAttribute("wrong","AA");

rd=req.getRequestDispatcher("as.jsp");

}

it will through an **error**.

or else it will continue to the next webpage/options.

**else**{

req.setAttribute("wrong",**null**);

session.setAttribute("A",1);

rd=req.getRequestDispatcher("login22.jsp");

}

User can change the username and password and when they click on that option it the corresponding method will get call.

**else** **if**(userAction.equals("CHANGE NAME")){

nameu(req,res);

}

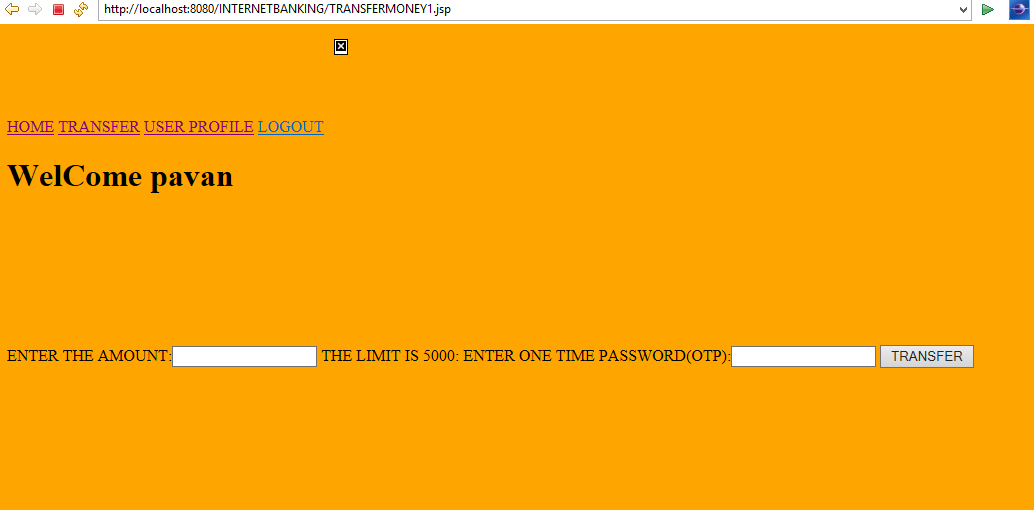
**else** **if**(userAction.equals("CHANGE PASSWORD")){

emailu(req,res);

}

**Online transfer:**

inorder to transfer online the user need OTP password. as shown below



package com.security.app.dao;

import java.util.List;

import org.hibernate.SessionFactory;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Repository;

import com.security.app.model.OTP;

@Repository("oTPDao")

public class OTPDaoImpl implements OTPDao {

@Autowired

private SessionFactory sessionFactory;

public void addOTP(OTP otp) {

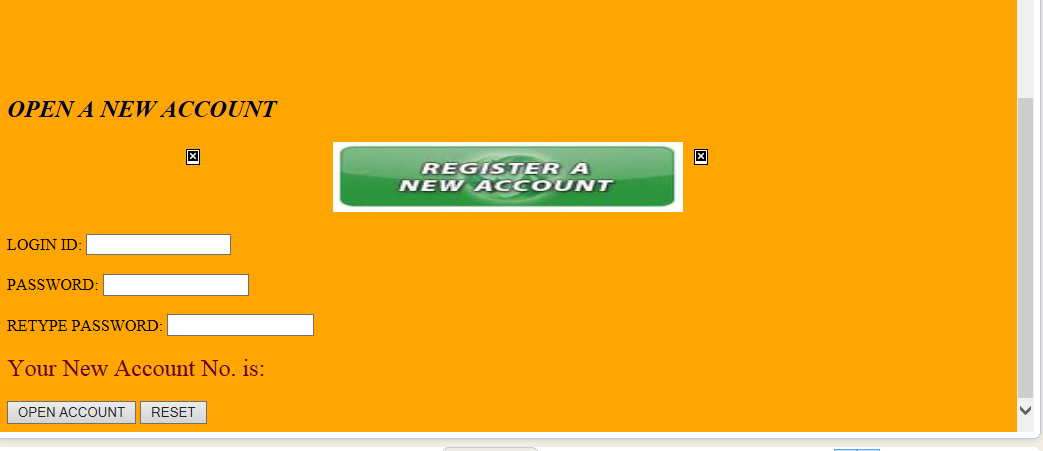
// TODO Auto-generated method stub\

sessionFactory.getCurrentSession().saveOrUpdate(otp);

}

The above code generate one time password it uses the package package com.security.app.dao;

**Register:**



**protected** **void** registration(HttpServletRequest req,HttpServletResponse res)**throws** ServletException, IOException {

**try**{

String log=req.getParameter("log");

String pas=req.getParameter("pas");

Integer nwac=Integer.*parseInt*(req.getParameter("acn"));

HttpSession session=req.getSession(**true**);

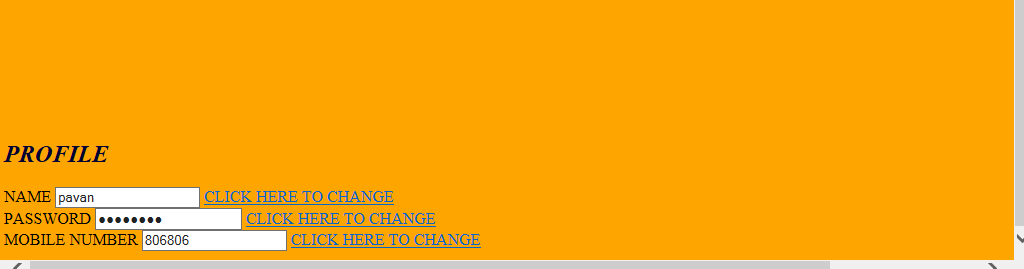
ArrayList personn=(ArrayList)session.getAttribute("ATT");

DetailsAccess DetailsAccess=**new** DetailsAccess();

Integer i=DetailsAccess.reg(personn,log,pas,nwac);

it get all the needed parameter and save it in the database.

**User Profile:**

****

We can view and also change the name, password and mobile number.

example method for changing name/updating :

**protected** **void** nameu(HttpServletRequest req,HttpServletResponse res)**throws** ServletException, IOException {

**try**{

PersonInfo person=**new** PersonInfo();

String name=req.getParameter("cname");

HttpSession session=req.getSession(**true**);

ArrayList personn=(ArrayList)session.getAttribute("ATT");

PersonInfo per=**null**;

person=(PersonInfo)personn.get(0);

PDAOU.nameupd(person.getid(),name);

Integer i=personn.size();

**for**(Integer j=0;j<i;j++)

{

person=(PersonInfo)personn.get(j);

person.setname(name);

personn.set(j, person);

}

Database that i have created :

