**SOURCE CODE**

**DBConnection.java**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package com.clouds.db;

import java.sql.Connection;

import java.sql.DriverManager;

/\*\*

\*

\* @author Ramu Maloth

\*/

public class DBConnection {

public static Connection con = null;

public static Connection getDBConnection(){

try {

DriverManager.registerDriver(new com.mysql.jdbc.Driver());

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

if(con!=null){

return con;

}

} catch (Exception e) {

System.out.println("Error at DBConnection "+e.getMessage());

}

return con;

}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package com.cbchain.actions;

import com.cbchain.models.CustomerDetails;

import java.io.IOException;

import java.io.PrintWriter;

import java.util.Random;

import javax.servlet.RequestDispatcher;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

/\*\*

\*

\* @author Ramu Maloth

\*/

public class InsuranceDetailsAction extends HttpServlet {

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

String loginuser = request.getParameter("loginuser").trim();

String insuranceuser = request.getParameter("insuranceuser").trim();

String mobile = request.getParameter("mobile").trim();

String email = request.getParameter("email").trim();

String locality = request.getParameter("locality").trim();

String address = request.getParameter("address").trim();

String insurtype = request.getParameter("insurtype").trim();

String insurancepremium = request.getParameter("insurancepremium").trim();

CustomerDetails cs = new CustomerDetails();

cs.setLoginuser(loginuser);

cs.setInsuranceuser(insuranceuser);

cs.setMobile(mobile);

cs.setEmail(email);

cs.setLocality(locality);

cs.setAddress(address);

cs.setInsurtype(insurtype);

cs.setInsurancepremium(insurancepremium);

String[] bank = {"SBI Bank","Union Bank","ICICI Bank","Axis Bank","Canara Bank","HDFC Bank","FDI Bank","Chase Bank"};

int rnd = new Random().nextInt(bank.length);

String bankName = bank[rnd];

cs.setBnakname(bankName);

request.setAttribute("cs", cs);

RequestDispatcher rd = request.getRequestDispatcher("./StartBlockChainProcess.jsp");

rd.forward(request, response);

}

/\*\*

\* Returns a short description of the servlet.

\*

\* @return a String containing servlet description

\*/

@Override

public String getServletInfo() {

return "Short description";

}// </editor-fold>

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package com.cbchain.actions;

import com.clouds.db.DBConnection;

import com.clouds.db.StartNode;

import com.dp.hyd.Block;

import java.io.IOException;

import java.io.PrintWriter;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.util.ArrayList;

import javax.servlet.RequestDispatcher;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

/\*\*

\*

\* @author Ramu Maloth

\*/

public class StartBlockChainAction extends HttpServlet {

/\*\*

\* Handles the HTTP <code>POST</code> method.

\*

\* @param request servlet request

\* @param response servlet response

\* @throws ServletException if a servlet-specific error occurs

\* @throws IOException if an I/O error occurs

\*/

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

// Transaction Details

String loginuser = request.getParameter("loginuser").trim();

String insuranceuser = request.getParameter("insuranceuser").trim();

String mobile = request.getParameter("mobile").trim();

String email = request.getParameter("email").trim();

String locality = request.getParameter("locality").trim();

String insurancetype = request.getParameter("insurancetype").trim();

String insurancepremium = request.getParameter("insurancepremium").trim();

String bnakname = request.getParameter("bnakname").trim();

String nameoncard = request.getParameter("nameoncard").trim();

String totalamount = request.getParameter("totalamount").trim();

String cardnumber = request.getParameter("cardnumber").trim();

String cvv = request.getParameter("cvv").trim();

String cardexpiry = request.getParameter("cardexpiry").trim();

String blkplatform = request.getParameter("blkplatform").trim();

String msg= insuranceuser+" sending "+totalamount+ " to "+bnakname;

//StartBlockTransactionUtil sbu = new StartBlockTransactionUtil();

//sbu.startProcess(Float.parseFloat(totalamount));

PreparedStatement ps = null;

try(Connection con = DBConnection.getDBConnection()){

String sqlQuery = "insert into insurancedetails(loginuser,insuranceuser,mobile,email,locality,insurancetype,insurancepremium,bnakname) values(?,?,?,?,?,?,?,?)";

ps = con.prepareStatement(sqlQuery);

ps.setString(1, loginuser);

ps.setString(2,insuranceuser);

ps.setString(3,mobile);

ps.setString(4,email);

ps.setString(5,locality);

ps.setString(6,insurancetype);

ps.setString(7,insurancepremium);

ps.setString(8,bnakname);

ps.executeUpdate();

String trnxQuery = "insert into trnxdetails(loginuser,nameoncard,totalamount,cardnumber,cvv,cardexpiry,blkplatform) values(?,?,?,?,?,?,?)";

ps = con.prepareStatement(trnxQuery);

ps.setString(1, loginuser);

ps.setString(2, nameoncard);

ps.setString(3, totalamount);

ps.setString(4, cardnumber);

ps.setString(5, cvv);

ps.setString(6, cardexpiry);

ps.setString(7, blkplatform);

ps.executeUpdate();

StartNode sn = new StartNode();

ArrayList<Block> blockchain = sn.startProcess(msg);

int size = blockchain.size();

Block b = null;

if(size==1){

b = blockchain.get(0);

}else{

b = blockchain.get(blockchain.size() - 1);

}

String sqlBlk = "insert into blockchain(loginuser,currenthash,previoushash,data,trnstimestamp,nonce) values(?,?,?,?,?,?)";

PreparedStatement ps1 = con.prepareStatement(sqlBlk);

ps1.setString(1, loginuser);

ps1.setString(2, b.getHash());

ps1.setString(3, b.getPreviousHash());

ps1.setString(4, b.getData());

ps1.setLong(5, b.getTimeStamp());

ps1.setInt(6, b.getNonce());

ps1.executeUpdate();

request.setAttribute("block", blockchain);

RequestDispatcher rd = getServletContext().getRequestDispatcher("/UserTransactionDetails.jsp");

rd.forward(request, response);

}catch(Exception ex){

System.out.println("Exception at Trabsactions "+ex.getMessage());

ex.printStackTrace();

}

}

/\*\*

\* Returns a short description of the servlet.

\*

\* @return a String containing servlet description

\*/

@Override

public String getServletInfo() {

return "Short description";

}// </editor-fold>

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package com.cbchain.actions;

import com.clouds.db.DBConnection;

import java.io.IOException;

import java.io.PrintWriter;

import java.sql.Connection;

import java.sql.PreparedStatement;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

/\*\*

\*

\* @author Ramu Maloth

\*/

public class UserRegisterActions extends HttpServlet {

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

String custname = request.getParameter("custname").trim();

String loginid = request.getParameter("loginid").trim();

String pswd = request.getParameter("pswd").trim();

String mobile = request.getParameter("mobile").trim();

String email = request.getParameter("email").trim();

String locality = request.getParameter("locality").trim();

String address = request.getParameter("address").trim();

String city = request.getParameter("city").trim();

String state = request.getParameter("state").trim();

String status = "waiting";

//Connection con = null;

PreparedStatement ps = null;

try(Connection con = DBConnection.getDBConnection()){

String sqlQuery = "insert into registrations(custname,loginid,pswd,mobile,email,locality,address,city,state,status) values(?,?,?,?,?,?,?,?,?,?)";

ps = con.prepareStatement(sqlQuery);

ps.setString(1, custname);

ps.setString(2, loginid);

ps.setString(3, pswd);

ps.setString(4, mobile);

ps.setString(5, email);

ps.setString(6, locality);

ps.setString(7, address);

ps.setString(8, city);

ps.setString(9, state);

ps.setString(10, status);

int no = ps.executeUpdate();

if(no>0){

response.sendRedirect("Registrations.jsp?msg=success");

}else{

response.sendRedirect("Registrations.jsp?msg=failed");

}

}catch(Exception ex){

System.out.println("Exception at Registrations = "+ex.getMessage());

response.sendRedirect("Registrations.jsp?msg=failed");

}

}

/\*\*

\* Returns a short description of the servlet.

\*

\* @return a String containing servlet description

\*/

@Override

public String getServletInfo() {

return "Short description";

}// </editor-fold>

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package com.cbchain.models;

/\*\*

\*

\* @author Ramu Maloth

\*/

public class CustomerDetails {

private String loginuser;

private String insuranceuser;

private String mobile;

private String email;

private String locality;

private String address;

private String insurtype;

private String insurancepremium;

private String bnakname;

public String getLoginuser() {

return loginuser;

}

public void setLoginuser(String loginuser) {

this.loginuser = loginuser;

}

public String getInsuranceuser() {

return insuranceuser;

}

public void setInsuranceuser(String insuranceuser) {

this.insuranceuser = insuranceuser;

}

public String getMobile() {

return mobile;

}

public void setMobile(String mobile) {

this.mobile = mobile;

}

public String getEmail() {

return email;

}

public void setEmail(String email) {

this.email = email;

}

public String getLocality() {

return locality;

}

public void setLocality(String locality) {

this.locality = locality;

}

public String getAddress() {

return address;

}

public void setAddress(String address) {

this.address = address;

}

public String getInsurtype() {

return insurtype;

}

public void setInsurtype(String insurtype) {

this.insurtype = insurtype;

}

public String getInsurancepremium() {

return insurancepremium;

}

public void setInsurancepremium(String insurancepremium) {

this.insurancepremium = insurancepremium;

}

public String getBnakname() {

return bnakname;

}

public void setBnakname(String bnakname) {

this.bnakname = bnakname;

}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package com.clouds.db;

import com.dp.hyd.Block;

import com.dp.hyd.StringUtil;

import java.util.ArrayList;

/\*\*

\*

\* @author Ramu Maloth

\*/

public class StartNode {

public static ArrayList<Block> blockchain = new ArrayList<Block>();

public static int difficulty = 5;

public ArrayList<Block> startProcess(String msg){

System.out.println("Trying to Mine block 1... ");

//addBlock(new Block("Hi im the first block", "0"));

if(blockchain.size()==0){

addBlock(new Block(msg,"0"));

}else{

String blk = blockchain.get(blockchain.size()-1).hash;

addBlock(new Block(msg,blk));

}

//System.out.println("Trying to Mine block 2... ");

//addBlock(new Block("Yo im the second block",blockchain.get(blockchain.size()-1).hash));

//System.out.println("Trying to Mine block 3... ");

//addBlock(new Block("Hey im the third block",blockchain.get(blockchain.size()-1).hash));

System.out.println("\nBlockchain is Valid: " + isChainValid());

String blockchainJson = StringUtil.getJson(blockchain);

System.out.println("\nThe block chain: ");

System.out.println(blockchainJson);

return blockchain;

}

public static Boolean isChainValid() {

Block currentBlock;

Block previousBlock;

String hashTarget = new String(new char[difficulty]).replace('\0', '0');

//loop through blockchain to check hashes:

for(int i=1; i < blockchain.size(); i++) {

currentBlock = blockchain.get(i);

previousBlock = blockchain.get(i-1);

//compare registered hash and calculated hash:

if(!currentBlock.hash.equals(currentBlock.calculateHash()) ){

System.out.println("Current Hashes not equal");

return false;

}

//compare previous hash and registered previous hash

if(!previousBlock.hash.equals(currentBlock.previousHash) ) {

System.out.println("Previous Hashes not equal");

return false;

}

//check if hash is solved

if(!currentBlock.hash.substring( 0, difficulty).equals(hashTarget)) {

System.out.println("This block hasn't been mined");

return false;

}

}

return true;

}

public static void addBlock(Block newBlock) {

newBlock.mineBlock(difficulty);

blockchain.add(newBlock);

}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package com.dp.hyd;

/\*\*

\*

\* @author Ramu Maloth

\*/

import java.util.Date;

public class Block {

public String hash;

public String previousHash;

private String data; //our data will be a simple message.

private long timeStamp; //as number of milliseconds since 1/1/1970.

private int nonce; //The nonce is the number that blockchain miners

//Block Constructor.

public Block(String data,String previousHash ) {

this.data = data;

this.previousHash = previousHash;

this.timeStamp = new Date().getTime();

this.hash = calculateHash(); //Making sure we do this after we set the other values.

}

//Calculate new hash based on blocks contents

public String calculateHash() {

String calculatedhash = StringUtil.applySha256(

previousHash +

Long.toString(timeStamp) +

Integer.toString(nonce) +

data

);

return calculatedhash;

}

//Increases nonce value until hash target is reached.

public void mineBlock(int difficulty) {

String target = StringUtil.getDificultyString(difficulty); //Create a string with difficulty \* "0"

while(!hash.substring( 0, difficulty).equals(target)) {

nonce ++;

hash = calculateHash();

}

System.out.println("Block Mined!!! : " + hash);

}

public String getHash() {

return hash;

}

public void setHash(String hash) {

this.hash = hash;

}

public String getPreviousHash() {

return previousHash;

}

public void setPreviousHash(String previousHash) {

this.previousHash = previousHash;

}

public String getData() {

return data;

}

public void setData(String data) {

this.data = data;

}

public long getTimeStamp() {

return timeStamp;

}

public void setTimeStamp(long timeStamp) {

this.timeStamp = timeStamp;

}

public int getNonce() {

return nonce;

}

public void setNonce(int nonce) {

this.nonce = nonce;

}

}

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package com.dp.hyd;

/\*\*

\*

\* @author Ramu Maloth

\*/

import java.security.MessageDigest;

import com.google.gson.GsonBuilder;

public class StringUtil {

//Applies Sha256 to a string and returns the result.

public static String applySha256(String input){

try {

MessageDigest digest = MessageDigest.getInstance("SHA-256");

//Applies sha256 to our input,

byte[] hash = digest.digest(input.getBytes("UTF-8"));

StringBuffer hexString = new StringBuffer(); // This will contain hash as hexidecimal

for (int i = 0; i < hash.length; i++) {

String hex = Integer.toHexString(0xff & hash[i]);

if(hex.length() == 1) hexString.append('0');

hexString.append(hex);

}

return hexString.toString();

}

catch(Exception e) {

throw new RuntimeException(e);

}

}

//Short hand helper to turn Object into a json string

public static String getJson(Object o) {

return new GsonBuilder().setPrettyPrinting().create().toJson(o);

}

//Returns difficulty string target, to compare to hash. eg difficulty of 5 will return "00000"

public static String getDificultyString(int difficulty) {

return new String(new char[difficulty]).replace('\0', '0');

}

}

Index.jsp

<!DOCTYPE html>

<html lang="en">

<head>

<title>Cloud Computing and Blockchain</title>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<link href="https://fonts.googleapis.com/css?family=Work+Sans:300,400,500,600,700" rel="stylesheet">

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">

<link rel="stylesheet" href="css/animate.css">

<link rel="stylesheet" href="css/owl.carousel.min.css">

<link rel="stylesheet" href="css/owl.theme.default.min.css">

<link rel="stylesheet" href="css/magnific-popup.css">

<link rel="stylesheet" href="css/flaticon.css">

<link rel="stylesheet" href="css/style.css">

</head>

<body>

<nav class="navbar navbar-expand-lg navbar-dark ftco\_navbar bg-dark ftco-navbar-light" id="ftco-navbar">

<div class="container">

<a class="navbar-brand" href="index.jsp">Towards Cloud Computing and Blockchain Integrated Applications</a>

<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#ftco-nav" aria-controls="ftco-nav" aria-expanded="false" aria-label="Toggle navigation">

<span class="oi oi-menu"></span> Menu

</button>

<jsp:include page="mainmenu.html"></jsp:include>

</div>

</nav>

<!-- END nav -->

<div class="hero-wrap js-fullheight">

<div class="overlay"></div>

<div class="container-fluid px-0">

<div class="row d-md-flex no-gutters slider-text align-items-center js-fullheight justify-content-start">

<img class="one-third js-fullheight align-self-end order-md-first img-fluid" src="images/undraw\_co-working\_825n.svg" alt="">

<div class="one-forth d-flex align-items-center ftco-animate js-fullheight">

<div class="text mt-5">

<span class="subheading">Cloud Computing</span>

<h1 class="mb-3"><span>Blockchain and </span> <span>Integrated </span> <span>Applications</span></h1>

<p>Cloud Computing is a technology widely used in academia and industry, providing varied services on demand.</p>

<!--<p><a href="#" class="btn btn-secondary px-4 py-3">Get in touch</a></p>-->

</div>

</div>

</div>

</div>

</div>

<section class="ftco-section services-section bg-light">

<div class="container">

<div class="row justify-content-center mb-5">

<div class="col-md-8 text-center heading-section ftco-animate">

<h2 class="mb-3">Cloud Services</h2>

</div>

</div>

<div class="row">

<div class="col-lg-4 col-md-6 d-flex align-self-stretch ftco-animate">

<div class="media block-6 services d-flex">

<div class="icon d-flex align-items-center justify-content-center">

<span class="flaticon-cloud"></span>

</div>

<div class="media-body pl-4">

<h3 class="heading">Cloud databases</h3>

<p class="mb-0">Even the all-powerful Pointing has no control about the blind texts it is an almost unorthographic.</p>

</div>

</div>

</div>

<div class="col-lg-4 col-md-6 d-flex align-self-stretch ftco-animate">

<div class="media block-6 services d-flex">

<div class="icon d-flex align-items-center justify-content-center">

<span class="flaticon-server"></span>

</div>

<div class="media-body pl-4">

<h3 class="heading">Website Hosting</h3>

<p class="mb-0">Even the all-powerful Pointing has no control about the blind texts it is an almost unorthographic.</p>

</div>

</div>

</div>

<div class="col-lg-4 col-md-6 d-flex align-self-stretch ftco-animate">

<div class="media block-6 services d-flex">

<div class="icon d-flex align-items-center justify-content-center">

<span class="flaticon-customer-service"></span>

</div>

<div class="media-body pl-4">

<h3 class="heading">File Storage</h3>

<p>Even the all-powerful Pointing has no control about the blind texts it is an almost unorthographic.</p>

</div>

</div>

</div>

<div class="col-lg-4 col-md-6 d-flex align-self-stretch ftco-animate">

<div class="media block-6 services d-flex">

<div class="icon d-flex align-items-center justify-content-center">

<span class="flaticon-life-insurance"></span>

</div>

<div class="media-body pl-4">

<h3 class="heading">Forex Trading</h3>

<p>Even the all-powerful Pointing has no control about the blind texts it is an almost unorthographic.</p>

</div>

</div>

</div>

<div class="col-lg-4 col-md-6 d-flex align-self-stretch ftco-animate">

<div class="media block-6 services d-flex">

<div class="icon d-flex align-items-center justify-content-center">

<span class="flaticon-cloud-computing"></span>

</div>

<div class="media-body pl-4">

<h3 class="heading">File Backups</h3>

<p>Even the all-powerful Pointing has no control about the blind texts it is an almost unorthographic.</p>

</div>

</div>

</div>

<div class="col-lg-4 col-md-6 d-flex align-self-stretch ftco-animate">

<div class="media block-6 services d-flex">

<div class="icon d-flex align-items-center justify-content-center">

<span class="flaticon-settings"></span>

</div>

<div class="media-body pl-4">

<h3 class="heading">Remote Desktop</h3>

<p>Even the all-powerful Pointing has no control about the blind texts it is an almost unorthographic.</p>

</div>

</div>

</div>

</div>

</div>

</section>

<footer class="ftco-footer ftco-bg-dark ftco-section">

<div class="container">

<div class="row">

<div class="col-md-12 text-center">

<p>

Copyright &copy;<script>document.write(new Date().getFullYear());</script> All rights reserved | This template is made with <i class="fa fa-heart" aria-hidden="true"></i> by <a href="#" target="\_blank">Alex Corporations</a>

</div>

</div>

</div>

</footer>

<div id="ftco-loader" class="show fullscreen"><svg class="circular" width="48px" height="48px"><circle class="path-bg" cx="24" cy="24" r="22" fill="none" stroke-width="4" stroke="#eeeeee"/><circle class="path" cx="24" cy="24" r="22" fill="none" stroke-width="4" stroke-miterlimit="10" stroke="#F96D00"/></svg></div>

<script src="js/jquery.min.js"></script>

<script src="js/jquery-migrate-3.0.1.min.js"></script>

<script src="js/popper.min.js"></script>

<script src="js/bootstrap.min.js"></script>

<script src="js/jquery.easing.1.3.js"></script>

<script src="js/jquery.waypoints.min.js"></script>

<script src="js/jquery.stellar.min.js"></script>

<script src="js/owl.carousel.min.js"></script>

<script src="js/jquery.magnific-popup.min.js"></script>

<script src="js/jquery.animateNumber.min.js"></script>

<script src="js/scrollax.min.js"></script>

<script src="https://maps.googleapis.com/maps/api/js?key=AIzaSyBVWaKrjvy3MaE7SQ74\_uJiULgl1JY0H2s&sensor=false"></script>

<script src="js/google-map.js"></script>

<script src="js/main.js"></script>

</body>

</html>