Name – Tejas pandit

ENROLLMENT NO – 1810DMBCSE03320

Subject –Advance java

MST ASSIGNMENT

Given the current situation of Covid-19, a lot of citizens are require to travel from one district to another or one state to another state. As travelling is prohibited due to lockdown, each state government is providing e-passes to the citizens having emergency. One such e-pass service is provided by Government of Madhya Pradesh at URL https://mapit.gov.in/covid-19/ . Please visit this URL and study the procedure of issuing e-pass to the citizens. Suppose you need to build this application by using JSP/servlet.

1. Provide the database schema (tables and relations) for this project.

ANSWER- TABLES USED IN DATABASE FOR THE PROGRAMME

DATABASE – cust.sql

Table name – cust

Table columns – 1.FIRST NAME NOT NULL;

2.LAST NAME ;

3.DISTRICT;

4.DESTINATION

5.ADDRESS

6.CONTACT NO.

SCHEMA-

|CUST TABLE|

FIRSTNAME

LASTNMAE

DISTRICT

DESTINATION

ADDRESS

CONTACT NO

2) Describe all the servlet classes and their functionalities that you need to build this project.

ANSWER- Source code of servlet class

package net.javaguides.paSSHmanagement.web;

import java.io.IOException;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import net.javaguides.paSSHmanagement.dao.PaSSHDao;

import net.javaguides.paSSHmanagement.model.PaSSH;

/\*\*

\* @email Ramesh Fadatare

\*/

@WebServlet("/register")

public class PassServlet extends HttpServlet {

private static final long serialVersionUID = 1 L;

private PaSSHDao paSSHDao;

public void init() {

paSSHDao = new PaSSHDao();

}

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

String firstName = request.getParameter("firstName");

String lastName = request.getParameter("lastName");

String username = request.getParameter("username");

String password = request.getParameter("password");

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

String contact = request.getParameter("contact");

PaSSH paSSH = new PaSSH();

paSSH.setFirstName(firstName);

paSSH.setLastName(lastName);

paSSH.setUsername(username);

paSSH.setPassword(password);

paSSH.setContact(contact);

paSSH.setAddress(address);

try {

paSSHDao.registerPaSSH(paSSH);

} catch (Exception e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

response.sendRedirect("paSSHdetails.jsp");

}

}

Functionality of servlet class-

PassServlet class to process HTTP request parameters and redirect to the appropriate JSP page after request data stored in the database:

3) Describe all the non-servlet java classes and their functionalities that you will use in servlet classes.

ANSWER- There are two non servlet class in the programme –

1.Pass.java

2. Passdao.java

FUNCTIONALITIES-

1.Pass.java functionalities – paSSH javaBean class which we will use in JSP action tags.

Answer-

package net.javaguides.jsp.jdbc.bean;

import java.io.Serializable;

/\*\*

\* JavaBean class used in jsp action tags.

\* @author Ramesh Fadatare

\*/

public class PaSSH implements Serializable {

/\*\*

\*

\*/

private static final long serialVersionUID = 1 L;

private String firstName;

private String lastName;

private String username;

private String password;

private String address;

private String contact;

public String getFirstName() {

return firstName;

}

public void setFirstName(String firstName) {

this.firstName = firstName;

}

public String getLastName() {

return lastName;

}

public void setLastName(String lastName) {

this.lastName = lastName;

}

public String getUsername() {

return username;

}

public void setUsername(String username) {

this.username = username;

}

public String getPassword() {

return password;

}

public void setPassword(String password) {

this.password = password;

}

public String getAddress() {

return address;

}

public void setAddress(String address) {

this.address = address;

}

public String getContact() {

return contact;

}

public void setContact(String contact) {

this.contact = contact;

}

}

2.Passdao.java Functionalities - PaSSHDao class which contains JDBC code to connect with MySQL database. Add the following code to an PaSSHDao class.

Source code –

package net.javaguides.jsp.jdbc.database;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.SQLException;

import net.javaguides.jsp.jdbc.bean.PaSSH;

public class PaSSHDao {

public int registerPaSSH(PaSSH paSSH) throws ClassNotFoundException {

String INSERT\_USERS\_SQL = "INSERT INTO paSSH" +

" (id, first\_name, last\_name, username, password, address, contact) VALUES " +

" (?, ?, ?, ?, ?,?,?);";

int result = 0;

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

try (Connection connection = DriverManager

.getConnection("jdbc:mysql://localhost:3306/paSSHs?useSSL=false", "root", "root");

// Step 2:Create a statement using connection object

PreparedStatement preparedStatement = connection.prepareStatement(INSERT\_USERS\_SQL)) {

preparedStatement.setInt(1, 1);

preparedStatement.setString(2, paSSH.getFirstName());

preparedStatement.setString(3, paSSH.getLastName());

preparedStatement.setString(4, paSSH.getUsername());

preparedStatement.setString(5, paSSH.getPassword());

preparedStatement.setString(6, paSSH.getAddress());

preparedStatement.setString(7, paSSH.getContact());

System.out.println(preparedStatement);

// Step 3: Execute the query or update query

result = preparedStatement.executeUpdate();

} catch (SQLException e) {

// process sql exception

printSQLException(e);

}

return result;

}

private void printSQLException(SQLException ex) {

for (Throwable e: ex) {

if (e instanceof SQLException) {

e.printStackTrace(System.err);

System.err.println("SQLState: " + ((SQLException) e).getSQLState());

System.err.println("Error Code: " + ((SQLException) e).getErrorCode());

System.err.println("Message: " + e.getMessage());

Throwable t = ex.getCause();

while (t != null) {

System.out.println("Cause: " + t);

t = t.getCause();

}

}

}

}

}

4) Describe all the major challenges that you may face to build this project .

ANSWER – the major challenge that I will face during this project is verification of the user i.e to generate otp for mobile numbers and verify the exact location o

5) Build this application and submit the java source file and database .sql file.

Source code –

JSP code-

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"

pageEncoding="ISO-8859-1"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="ISO-8859-1">

<title>Insert title here</title>

</head>

<body>

<div align="center">

<h1>Employee Register Form</h1>

<form action="<%= request.getContextPath() %>/register" method="post">

<table style="with: 80%">

<tr>

<td>First Name</td>

<td><input type="text" name="firstName" /></td>

</tr>

<tr>

<td>Last Name</td>

<td><input type="text" name="lastName" /></td>

</tr>

<tr>

<td>UserName</td>

<td><input type="text" name="username" /></td>

</tr>

<tr>

<td>Password</td>

<td><input type="password" name="password" /></td>

</tr>

<tr>

<td>Address</td>

<td><input type="text" name="address" /></td>

</tr>

<tr>

<td>Contact No</td>

<td><input type="text" name="contact" /></td>

</tr>

</table>

<input type="submit" value="Submit" />

</form>

</div>

</body>

</html>

9. Create an epassdetails.jsp

After an epass successfully registered then this page show a successful message on screen:

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"

pageEncoding="ISO-8859-1"%>

<%@page import="net.javaguides.employeemanagement.dao.\*"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="ISO-8859-1">

<title>Insert title here</title>

</head>

<body>

<h1>User successfully registered!</h1>

</body>

</html>