package com.learnersacademy.admin;

import java.io.IOException;

import java.util.List;

import javax.annotation.Resource;

import javax.servlet.RequestDispatcher;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.Cookie;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.sql.DataSource;

import com.learnersacademy.models.Student;

import com.learnersacademy.models.Subject;

import com.learnersacademy.models.Teacher;

import com.learnersacademy.models.Class;

/\*\*

\* Servlet class AdminServlet

\*/

@WebServlet("/AdminServlet")

public class AdminServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

private DbRetrieve dbRetrieve;

@Resource(name = "jdbc\_database")

private DataSource datasource;

@Override

public void init() throws ServletException {

super.init();

try {

dbRetrieve = new DbRetrieve(datasource);

} catch (Exception e) {

throw new ServletException(e);

}

}

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public AdminServlet() {

super();

}

@Override

protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

doGet(req, resp);

}

/\*\*

\* @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse

\* response)

\*/

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

try {

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

if (command == null) {

command = "CLASSES";

}

if (!getCookies(request, response) && (!command.equals("LOGIN"))) {

response.sendRedirect("/Admin-Portal/login.jsp");

} else {

switch (command) {

case "STUDENTS":

studentsList(request, response);

break;

case "TEACHERS":

teachersList(request, response);

break;

case "SUBJECTS":

subjectList(request, response);

break;

case "CLASSES":

classestList(request, response);

break;

case "ST\_LIST":

classStudentsList(request, response);

break;

case "LOGIN":

login(request, response);

break;

default:

classestList(request, response);

}

}

} catch (Exception e) {

throw new ServletException(e);

}

}

private void studentsList(HttpServletRequest request, HttpServletResponse response) throws Exception {

List<Student> students = dbRetrieve.getStudents();

request.setAttribute("STUDENT\_LIST", students);

RequestDispatcher dispatcher = request.getRequestDispatcher("/list-students.jsp");

dispatcher.forward(request, response);

}

private void teachersList(HttpServletRequest request, HttpServletResponse response) throws Exception {

List<Teacher> teachers = dbRetrieve.getTeachers();

request.setAttribute("TEACHERS\_LIST", teachers);

RequestDispatcher dispatcher = request.getRequestDispatcher("/teachers-list.jsp");

dispatcher.forward(request, response);

}

private void subjectList(HttpServletRequest request, HttpServletResponse response) throws Exception {

List<Subject> subjects = dbRetrieve.getSubjects();

request.setAttribute("SUBJECTS\_LIST", subjects);

RequestDispatcher dispatcher = request.getRequestDispatcher("/subjects-list.jsp");

dispatcher.forward(request, response);

}

private void classestList(HttpServletRequest request, HttpServletResponse response) throws Exception {

List<Class> classes = dbRetrieve.getClasses();

request.setAttribute("CLASSES\_LIST", classes);

RequestDispatcher dispatcher = request.getRequestDispatcher("/classes-list.jsp");

dispatcher.forward(request, response);

}

private void login(HttpServletRequest request, HttpServletResponse response) throws Exception {

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

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

if (username.toLowerCase().equals("admin") && password.toLowerCase().equals("admin")) {

Cookie cookie = new Cookie(username, password);

cookie.setMaxAge(86400); // 86400 seconds in a day

response.addCookie(cookie);

classestList(request, response);

} else {

RequestDispatcher dispatcher = request.getRequestDispatcher("/login.jsp");

dispatcher.forward(request, response);

}

}

private void classStudentsList(HttpServletRequest request, HttpServletResponse response) throws Exception {

int classId = Integer.parseInt(request.getParameter("classId"));

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

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

List<Student> students = dbRetrieve.loadClassStudents(classId);

request.setAttribute("STUDENTS\_LIST", students);

request.setAttribute("SECTION", section);

request.setAttribute("SUBJECT", subject);

RequestDispatcher dispatcher = request.getRequestDispatcher("/class-students.jsp");

dispatcher.forward(request, response);

}

private boolean getCookies(HttpServletRequest request, HttpServletResponse response) throws Exception {

boolean check = false;

Cookie[] cookies = request.getCookies();

// Find the cookie of interest in arrays of cookies

for(int i=0;cookies!=null&&i<cookies.length;i++) {

if (cookies[i].getName().equals("admin") && cookies[i].getValue().equals("admin")) {

check = true;

break; }

}

return check;

}

}