package com.zumba.controller;

import java.io.IOException;

import java.util.List;

import java.io.PrintWriter;

import javax.servlet.RequestDispatcher;

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 javax.servlet.http.HttpSession;

import com.zumba.bean.MemberEvent;

import com.zumba.bean.Member;

import com.zumba.bean.Event;

import com.zumba.service.MemberService;

import com.zumba.service.EventService;

import com.zumba.service.MemberEventService;

@WebServlet("/memberController")

public class MemberController extends HttpServlet {

private static final long serialVersionUID = 1L;

public MemberController() {

super();

}

private MemberService ms = new MemberService();

private EventService es = new EventService();

private MemberEventService mes = new MemberEventService();

protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

List<Member> listOfMember = ms.viewAllMemberDetails();

HttpSession hs = request.getSession();

hs.setAttribute("listAllMember", listOfMember);

response.sendRedirect("viewAllMember.jsp");

response.getWriter().append("Served at: ").append(request.getContextPath());

}

protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

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

if (userAction .equals("viewSpecificMember")) {

int memberID = Integer.parseInt(request.getParameter("MID"));

Member im = ms.viewSpecificMember(memberID);

HttpSession hs = request.getSession();

hs.setAttribute("specificMember", im);

List<Event> listOfEventForMember = es.viewEventsForMember(memberID);

hs.setAttribute("listOfEvent", listOfEventForMember);

response.sendRedirect("viewSpecificMember.jsp");

}else if (userAction .equals("addMember")) {

String memberF\_name = request.getParameter("f\_name");

String memberL\_name = request.getParameter("l\_name");

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

Member nm = new Member();

nm.setF\_name(memberF\_name);

nm.setL\_name(memberL\_name);

nm.setEmail(memberEmail);

ms.addNewMember(nm);

doGet(request, response);

}else if (userAction .equals("deleteMember")) {

Member rm = new Member();

int memberID = Integer.parseInt(request.getParameter("MID"));

rm.setMID(memberID);

ms.removeMember(rm);

doGet(request, response); // after removing a member takes user back to list of members

}

else if (userAction .equals("removeMemberFromEvent")) {

MemberEvent rmfe = new MemberEvent();

HttpSession hs = request.getSession();

int memberID = Integer.parseInt(request.getParameter("MID"));

int eventID = Integer.parseInt(request.getParameter("EID"));

rmfe.setMID(memberID);

rmfe.setEID(eventID);

mes.removeMemberFromEvent(rmfe);

Member im = ms.viewSpecificMember(memberID);

hs.setAttribute("specificMember", im);

List<Event> listOfEventForMember = es.viewEventsForMember(memberID);

hs.setAttribute("listOfEvent", listOfEventForMember);

response.sendRedirect("viewSpecificMember.jsp");

}

}

}