package com.zumba.controller;

import java.io.IOException;

import java.io.PrintWriter;

import java.util.List;

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.Event;

import com.zumba.bean.Member;

import com.zumba.bean.MemberEvent;

import com.zumba.service.EventService;

import com.zumba.service.MemberService;

import com.zumba.service.MemberEventService;

@WebServlet("/eventController")

public class EventController extends HttpServlet {

private static final long serialVersionUID = 1L;

private MemberService ms;// = new MemberService();

private EventService es;// = new EventService();

private MemberEventService mes;// = new MemberEventService();

public EventController() {

super();

ms = new MemberService();

es = new EventService();

mes = new MemberEventService();

}

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

List<Event> listOfEvent = es.viewAlleventDetails();

HttpSession hs = request.getSession();

hs.setAttribute("listAllEvent", listOfEvent);

response.sendRedirect("viewAllEvent.jsp");

}

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

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

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

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

Event ie = es.viewSpecificEvent(eventID);

HttpSession hs = request.getSession();

hs.setAttribute("specificEvent", ie);

List<Member> listOfMemberInEvent = ms.viewAllMembersInEvent(eventID);

hs.setAttribute("listOfMember", listOfMemberInEvent);

response.sendRedirect("viewSpecificEvent.jsp");

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

String eventDay = request.getParameter("day");

String eventTime = request.getParameter("time");

String eventLocation = request.getParameter("location");

Event ne = new Event();

ne.setDay(eventDay);

ne.setTime(eventTime);

ne.setLocation(eventLocation);

es.addNewEvent(ne);

doGet(request, response);

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

MemberEvent nme = new MemberEvent();

HttpSession hs = request.getSession();

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

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

nme.setMID(memberID);

nme.setEID(eventID);

mes.addMemberToEvent(nme);

//putting this back in the session so that in the event of added functionality this will still display the member you put in to the event and the members already in the event.

Event ie = es.viewSpecificEvent(eventID);

hs.setAttribute("specificEvent", ie);

List<Member> listOfMemberInEvent = ms.viewAllMembersInEvent(eventID);

hs.setAttribute("listOfMember", listOfMemberInEvent);

response.sendRedirect("viewSpecificEvent.jsp");

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

Event re = new Event();

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

re.setEID(eventID);

es.removeEvent(re);

doGet(request, response); //return user to viewing list of all events by calling doGet method.

}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);

Event ie = es.viewSpecificEvent(eventID);

hs.setAttribute("specificEvent", ie);

List<Member> listOfMemberInEvent = ms.viewAllMembersInEvent(eventID);

hs.setAttribute("listOfMember", listOfMemberInEvent);

response.sendRedirect("viewSpecificEvent.jsp");

}else {

doGet(request, response);

}

}

}