```java
package webapplication.rest;

import java.net.URISyntaxException;


@Path("/userservice")
public class UserService {
    @Context
    HttpServletRequest request;

    OperatoerDAO dao = new MySQLOperatoerDAO();

    @POST
    @Path("/getusers")
    @Produces(MediaType.APPLICATION_JSON)
    public List<OperatoerDTO> getData(){
        List<OperatoerDTO> response = null;
        //System.out.println(request.getSession().getAttribute("user"));
        OperatoerDTO user = (OperatoerDTO) request.getSession().getAttribute("user"); //Session attribute

        for(String role : user.getRoles()){
            if(role.equals("admin")){
                try {
                    response = dao.getOperatoerList();
                    return response;
                } catch (DALException e) {
                    e.printStackTrace();
                }
            }
        }
        try{
            response = new ArrayList<OperatoerDTO>();
            response.add(dao.getOperatoer(user.getOprId()));
        }catch(DALException e) {
            e.printStackTrace();
        }
        return response;
    }

    /**
     * @param id
     * @return
     * @throws DALException
     * @throws URISyntaxException
     */
    @POST
    @Path("/delete")
    @Consumes("application/x-www-form-urlencoded")
    public Response deleteUser(
        @FormParam("submit") int id
            ) throws DALException, URISyntaxException {

        OperatoerDTO user = (OperatoerDTO) request.getSession().getAttribute("user"); //Session attribute

        java.net.URI location = new java.net.URI("../user_view.html");

        OperatoerDTO temp = dao.getOperatoer(id);

        if(temp.getOprNavn().equals("admin")){
            return Response.temporaryRedirect(location).build();
        }

        for(String role : user.getRoles()){
            if(role.equals("admin")){
                OperatoerDTO opr = dao.getOperatoer(id);
                opr.setOprActive(false);
                try { dao.updateOperatoer(opr); }
                catch (DALException e) { e.printStackTrace(); }
                return Response.temporaryRedirect(location).build();

            }
        }
        return Response.temporaryRedirect(location).build();
    }

    @POST
    @Path("/insert")
    @Consumes("application/x-www-form-urlencoded")
    public Response insertUser(
        @FormParam("username") String userName,
        @FormParam("password") String password,
        @FormParam("ini") String ini,
        @FormParam("role") String role
            ) throws DALException, URISyntaxException {

        OperatoerDTO user = (OperatoerDTO) request.getSession().getAttribute("user"); //Session attribute

        java.net.URI location = new java.net.URI("../user_view.html");

        for(String adminrole : user.getRoles()){
            if(adminrole.equals("admin")){

                List<OperatoerDTO> operatoers = dao.getOperatoerList();
                int id = 0;
                for(OperatoerDTO o:operatoers){
                    if(o.getOprId()>=id){id=o.getOprId()+1;}
                }

                OperatoerDTO opr = new OperatoerDTO(id, userName, ini, password, true);
                opr.setRoles(role);
                dao.createOperatoer(opr);
            }
        }
        return Response.temporaryRedirect(location).build();
    }

    @POST
    @Path("/updategetuser")
    @Consumes(MediaType.APPLICATION_JSON)
    @Produces(MediaType.APPLICATION_JSON)
    public OperatoerDTO updateGetUser(IdMapper map) throws URISyntaxException, DALException{
        //System.out.println(map.getContent());
        return dao.getOperatoer(map.getContent());
    }

    @POST
    @Path("/update")
    @Consumes("application/x-www-form-urlencoded")
    public Response updateUser(
            @FormParam("userid") String userId,
            @FormParam("username") String userName,
            @FormParam("password") String password,
            @FormParam("ini") String ini,
            @FormParam("role") String role,
            @FormParam("oprActive") String active
            ) throws URISyntaxException, DALException{

        OperatoerDTO user = (OperatoerDTO) request.getSession().getAttribute("user"); //Session attribute

        java.net.URI location = new java.net.URI("../user_view.html");

        OperatoerDTO temp = dao.getOperatoer(Integer.parseInt(userId));

        if(temp.getOprNavn().equals("admin")){
            return Response.temporaryRedirect(location).build();
        }

        for(String adminrole : user.getRoles()){
            if(adminrole.equals("admin")){
                OperatoerDTO opr = new OperatoerDTO(Integer.parseInt(userId), userName, ini, password, Boolean.parseBoolean(active));
                opr.setRoles(role);

                dao.updateOperatoer(opr);
                location = new java.net.URI("../user_view.html");
                return Response.temporaryRedirect(location).build();
            }
        }
        if(Integer.parseInt(userId) == user.getOprId()){
            OperatoerDTO opr = new OperatoerDTO(user.getOprId(), userName, ini, password, temp.getOprActive());
            opr.setRoles(temp.getRoles());

            dao.updateOperatoer(opr);
            location = new java.net.URI("../user_view.html");
        }
        return Response.temporaryRedirect(location).build();
    }


}
```