

UserService.java

```

1 package webapplication.rest;
2
3 import java.net.URISyntaxException;
20
21
22 @Path("/userservice")
23 public class UserService {
24     @Context
25     HttpServletRequest request;
26
27     OperatoerDAO dao = new MySQLOperatoerDAO();
28
29     @POST
30     @Path("/getusers")
31     @Produces(MediaType.APPLICATION_JSON)
32     public List<OperatoerDTO> getData(){
33         List<OperatoerDTO> response = null;
34         //System.out.println(request.getSession().getAttribute("user"));
35         OperatoerDTO user = (OperatoerDTO) request.getSession().getAttribute("user");
36         //Session attribute
37
38         for(String role : user.getRoles()){
39             if(role.equals("admin")){
40                 try {
41                     response = dao.getOperatoerList();
42                     return response;
43                 } catch (DAException e) {
44                     e.printStackTrace();
45                 }
46             }
47         }
48         try{
49             response = new ArrayList<OperatoerDTO>();
50             response.add(dao.getOperatoer(user.getOprId()));
51         }catch(DAException e) {
52             e.printStackTrace();
53         }
54         return response;
55     }
56
57     /**
58     * @param id
59     * @return
60     * @throws DAException
61     * @throws URISyntaxException
62     */
63     @POST
64     @Path("/delete")
65     @Consumes("application/x-www-form-urlencoded")
66     public Response deleteUser(
67         @FormParam("submit") int id
68         ) throws DAException, URISyntaxException {
69
70         OperatoerDTO user = (OperatoerDTO) request.getSession().getAttribute("user");
71         //Session attribute
72
73         java.net.URI location = new java.net.URI("../user_view.html");
74
75         OperatoerDTO temp = dao.getOperatoer(id);
76
77         if(temp.getOprNavn().equals("admin")){
78             return Response temporaryRedirect(location).build();
79         }
80     }
81 }

```

UserService.java

```

77     }
78
79     for(String role : user.getRoles()){
80         if(role.equals("admin")){
81             OperatoerDTO opr = dao.getOperatoer(id);
82             opr.setOprActive(false);
83             try { dao.updateOperatoer(opr); }
84             catch (DAException e) { e.printStackTrace(); }
85             return Response temporaryRedirect(location).build();
86         }
87     }
88     }
89     return Response temporaryRedirect(location).build();
90 }
91
92 @POST
93 @Path("/insert")
94 @Consumes("application/x-www-form-urlencoded")
95 public Response insertUser(
96     @FormParam("username") String userName,
97     @FormParam("password") String password,
98     @FormParam("ini") String ini,
99     @FormParam("role") String role
100     ) throws DAException, URISyntaxException {
101
102     OperatoerDTO user = (OperatoerDTO) request.getSession().getAttribute("user");
103     //Session attribute
104     java.net.URI location = new java.net.URI("../user_view.html");
105
106     for(String adminrole : user.getRoles()){
107         if(adminrole.equals("admin")){
108
109             List<OperatoerDTO> operators = dao.getOperatoerList();
110             int id = 0;
111             for(OperatoerDTO o:operators){
112                 if(o.getOprId()>=id){id=o.getOprId()+1;}
113             }
114
115             OperatoerDTO opr = new OperatoerDTO(id, userName, ini, password, true);
116             opr.setRoles(role);
117             dao.createOperatoer(opr);
118         }
119     }
120     return Response temporaryRedirect(location).build();
121 }
122
123 @POST
124 @Path("/updategetuser")
125 @Consumes(MediaType.APPLICATION_JSON)
126 @Produces(MediaType.APPLICATION_JSON)
127 public OperatoerDTO updateGetUser(IdMapper map) throws URISyntaxException,
128     DAException{
129     //System.out.println(map.getContent());
130     return dao.getOperatoer(map.getContent());
131 }
132
133 @POST
134 @Path("/update")
135 @Consumes("application/x-www-form-urlencoded")
136 public Response updateUser(
137     @FormParam("userid") String userId,

```

UserService.java

```

137         @RequestParam("username") String userName,
138         @RequestParam("password") String password,
139         @RequestParam("ini") String ini,
140         @RequestParam("role") String role,
141         @RequestParam("oprActive") String active
142         ) throws URISyntaxException, DALException{
143
144         OperatoerDTO user = (OperatoerDTO) request.getSession().getAttribute("user");
145         //Session attribute
146         java.net.URI location = new java.net.URI("../user_view.html");
147
148         OperatoerDTO temp = dao.getOperatoer(Integer.parseInt(userId));
149
150         if(temp.getOprNavn().equals("admin")){
151             return Response temporaryRedirect(location).build();
152         }
153
154         for(String adminrole : user.getRoles()){
155             if(adminrole.equals("admin")){
156                 OperatoerDTO opr = new OperatoerDTO(Integer.parseInt(userId), userName,
157                 ini, password, Boolean.parseBoolean(active));
158                 opr.setRoles(role);
159
160                 dao.updateOperatoer(opr);
161                 location = new java.net.URI("../user_view.html");
162                 return Response temporaryRedirect(location).build();
163             }
164         }
165         if(Integer.parseInt(userId) == user.getOprId()){
166             OperatoerDTO opr = new OperatoerDTO(user.getOprId(), userName, ini, password,
167             temp.getOprActive());
168             opr.setRoles(temp.getRoles());
169
170             dao.updateOperatoer(opr);
171             location = new java.net.URI("../user_view.html");
172             return Response temporaryRedirect(location).build();
173         }
174     }
175 }
176

```