Source Code:

```
package com.VaccinationCenter.Controller;

import java.util.ArrayList;
import java.util.List;
import java.util.logging.Logger;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.servlet.ModelAndView;
import org.springframework.web.servlet.mvc.support.RedirectAttributes;

import com.VaccinationCenter.Entities.Citizen;
import com.VaccinationCenter.Entities.VaccinationCenter;
import com.VaccinationCenter.Service.CitizenService;
import com.VaccinationCenter.Service.VaccinationCenterService;
@Controller
@RequestMapping("/admin/citizen")
public class CitizenController {
      @Autowired
      CitizenService citizenService;

      @Autowired
      VaccinationCenterService vaccinationService;

      @RequestMapping("/addCitizen")
      public ModelAndView AddNewCitizen(HttpServletRequest request,
HttpServletResponse response) {
           ModelAndView mv = new ModelAndView();
           List<VaccinationCenter> vaccinationCenters =
vaccinationService.getAllCenters();
           List<VaccinationCenter> inactiveCenters = new ArrayList<>();
          for (VaccinationCenter center : vaccinationCenters) {
              String status = center.getStatus();
              if (status == null) {
                  inactiveCenters.add(center);
              }
          }
           mv.addObject("centers", inactiveCenters);
           mv.setViewName("addCitizen");
           return mv;
      }

      @RequestMapping("/viewCitizen")
      public ModelAndView ViewCitizens(HttpServletRequest request,
HttpServletResponse response) {
           ModelAndView mv = new ModelAndView();
        List<Citizen> citizens = citizenService.getAllCitizens(); //
Retrieve all centers from the service
          int citizenCount = citizens.size();
          mv.addObject("citizens", citizens);
          mv.addObject("citizenCount", citizenCount > 0 ? citizenCount
: 0);

           mv.setViewName("viewCitizen");
           return mv;
      }


         @RequestMapping("/editCitizen")
          public ModelAndView editCitizen(@RequestParam("id") int id
,HttpServletRequest request, HttpServletResponse response) {
              ModelAndView mv = new ModelAndView();
             Citizen citizen = citizenService.getCitizenById(id);
                List<VaccinationCenter> vaccinationCenters =
vaccinationService.getAllCenters();
                List<VaccinationCenter> inactiveCenters = new
ArrayList<>();
               for (VaccinationCenter center : vaccinationCenters) {
                   String status = center.getStatus();
                   if (status == null) {
                       inactiveCenters.add(center);
                   }
               }
                mv.addObject("centers", inactiveCenters);

             mv.addObject("citizen", citizen);
             mv.setViewName("editCitizen");
             return mv;
          }


         @RequestMapping("/updateCitizen")
        public ModelAndView updateCitizen(
            @RequestParam("id") int id,
            @RequestParam("name") String name,
            @RequestParam("centerName") String centerName,
            @RequestParam("city") String city,
            @RequestParam("doses") int doses,
            HttpServletRequest request,
            HttpServletResponse response,
            RedirectAttributes redirectAttributes
        ) {
            ModelAndView mv = new ModelAndView();
            Citizen citizen = citizenService.getCitizenById(id);
            if (citizen != null) {
                citizen.setName(name);
                citizen.setCity(city);

                // Prevent changing the centerName
                if
(!centerName.equals(citizen.getCenter().getCenterName())) {

mv.setViewName("redirect:/admin/citizen/editCitizen?id=" + id);

                    String message = "You cannot change the clinic.";
                    redirectAttributes.addFlashAttribute("message",
message);
                    return mv;
                }

                citizen.setDoses(doses);

                // Update vaccination status based on doses
                if (doses == 0) {
                    citizen.setVaccinationStatus("Not Vaccinated");
                } else if (doses == 1) {
                    citizen.setVaccinationStatus("First Dose");
                } else if (doses == 2) {
                    citizen.setVaccinationStatus("Fully Vaccinated");
                }

                citizenService.updateCitizen(citizen);
                mv.setViewName("redirect:/admin/citizen/viewCitizen");
                String message = name+" IS UPDATED";
                redirectAttributes.addFlashAttribute("message",
message);
            } else {
```

```
mv.setViewName("redirect:/admin/vaccination/editCenter?id=" + id);
                 redirectAttributes.addFlashAttribute("message", 
"Failed to update citizen.");
             }

             return mv;
         }

         @RequestMapping("/deleteCitizen")
          public ModelAndView deleteCitizen(@RequestParam("id") int id, 
HttpServletRequest request, HttpServletResponse response, 
RedirectAttributes redirectAttributes) {
             ;
              citizenService.deleteCitizenById(id);
              ModelAndView mv = new ModelAndView();
              mv.setViewName("redirect:/admin/citizen/viewCitizen");
              redirectAttributes.addFlashAttribute("message","CITIZEN 
DELETED");
              return mv;

          }

      @RequestMapping("/insertCitizen")
      public ModelAndView insertCitizen(HttpServletRequest request, 
HttpServletResponse response, Model model) {
          ModelAndView mv = new ModelAndView();
          Citizen citizen = new Citizen();
          String name = request.getParameter("name");
          String city = request.getParameter("city");
          String centerName = request.getParameter("centerName");

          if (name == null || name.isEmpty() || city == null || 
city.isEmpty() || centerName == null || centerName.isEmpty()) {
              mv.setViewName("addCitizen");
              mv.addObject("message", "Invalid or fill every block ");
          } else {
              try {
                  citizen.setName(name);
                  citizen.setCity(city);
                  String vaccinationStatus = "Not Vaccinated";

                  citizen.setVaccinationStatus(vaccinationStatus);
                  VaccinationCenter center = new VaccinationCenter();
                  center.setCenterName(centerName);
                  String status = "active";
```

```
                center.setStatus(status);
                center.setCity(city);
                citizen.setCenter(center);
                citizenService.saveCitizen(citizen);

mv.setViewName("redirect:/admin/citizen/viewCitizen");
                mv.addObject("citizen", citizen);
            } catch (Exception e) {
                mv.setViewName("addCitizen");
                mv.addObject("message", "Failed to save Citizen");
            }
        }

        return mv;
    }

    @RequestMapping("/viewCitizenById")
    public ModelAndView viewCitizen(@RequestParam("id") int id) {
        ModelAndView mv = new ModelAndView();
        Citizen citizen = citizenService.getCitizenById(id);
        if (citizen != null) {
            mv.addObject("citizen", citizen);
            mv.setViewName("viewCitizenById");
        } else {
            mv.setViewName("error");
            mv.addObject("message", "Citizen not found");
        }

        return mv;
    }

}


} }

package com.VaccinationCenter.Controller;

import java.util.logging.Logger;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.servlet.ModelAndView;
```

```
@Controller
public class HomeController {

	Logger log = Logger.getAnonymousLogger();

	@RequestMapping("/")
	public ModelAndView basepage(HttpServletRequest request,
HttpServletResponse response) {
		ModelAndView mv = new ModelAndView();
		log.info("login page");

		mv.setViewName("redirect:/admin/login");
		return mv;
	}
}
package com.VaccinationCenter.Controller;

import java.util.List;
import java.util.logging.Logger;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.dao.DataIntegrityViolationException;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.servlet.ModelAndView;
import org.springframework.web.servlet.mvc.support.RedirectAttributes;

import com.VaccinationCenter.Entities.Admin;
import com.VaccinationCenter.Entities.VaccinationCenter;
import com.VaccinationCenter.Service.AdminService;
import com.VaccinationCenter.Service.VaccinationCenterService;
@Controller
@RequestMapping("/admin")
public class LoginController {

	@Autowired
	AdminService service;

	@Autowired
	VaccinationCenterService vaccinationService;
```

```
     @RequestMapping("/login")
     public ModelAndView basepage(HttpServletRequest request,
HttpServletResponse response) {
           ModelAndView mv = new ModelAndView();

           mv.setViewName("login");
           return mv;
     }

     @RequestMapping("/register")
     public ModelAndView register(HttpServletRequest request,
HttpServletResponse response) {
           ModelAndView mv = new ModelAndView();
           mv.setViewName("register");
           return mv;
     }


     @RequestMapping("/logout")
     public ModelAndView Logout(HttpServletRequest request,
HttpServletResponse response) {
         ModelAndView mv = new ModelAndView();
         HttpSession session = request.getSession(false); // Retrieve
the existing session (if any)

         if (session != null) {
             session.invalidate(); // Invalidate the session if it
exists
         }

         mv.addObject("message", "Your session has been logged out.");
         mv.setViewName("login");

         return mv;
     }

     @RequestMapping("/insert")
     public ModelAndView insert(HttpServletRequest request,
HttpServletResponse response,RedirectAttributes redirectAttributes) {
           ModelAndView mv = new ModelAndView();
           Admin admin = new Admin();
           String name = request.getParameter("name");
           String email = request.getParameter("email");
           String password = request.getParameter("password");
```

```
            if (name.isEmpty() || name == null ||  email == null ||
password == null || email.isEmpty() || password.isEmpty()) {
                mv.setViewName("register");
                mv.addObject("message", "Invalid name or password");

           } else {
                 try {
                       admin.setName(name);
                       admin.setEmail(email);
                       admin.setPassword(password);
                       service.addAdmin(admin);
                       mv.setViewName("redirect:/");
                       mv.addObject(admin);
                   } catch (DataIntegrityViolationException e) {
                       mv.setViewName("register");
                       mv.addObject("message", "Email already exists");
                   } catch (Exception e) {
                       mv.setViewName("register");
                       mv.addObject("message", "Failed to save admin");
                   }
           }

           return mv;
     }


     @RequestMapping("/dashboard")
     public ModelAndView DashboardLogin(HttpServletRequest request,
HttpServletResponse response) {
           ModelAndView mv = new ModelAndView();
           String email = request.getParameter("email");
           String password = request.getParameter("password");

           if (email == null || email.isEmpty() && password.isEmpty())
{
                mv.setViewName("redirect:/"); //homecontroller login
                String message = "invalid";
               mv.addObject("message", message);
           }else {
             //check login by email and password
                 Admin authenticatedAdmin =
service.findByEmailAndPassword(email, password);
                   if (authenticatedAdmin != null) {
                       Admin admin = service.getAdminByEmail(email);
```

```
                      mv.setViewName("dashboard");
                      HttpSession httpsession = request.getSession();
                      httpsession.setAttribute("userid", 
admin.getName());
                           mv.addObject("admin",admin);
                   } else {
                       mv.setViewName("login");
                       String message = "Invalid email or password";
                           mv.addObject("message", message);
           }
                   }
           return mv;

           }


     @RequestMapping("viewCenter")
     public ModelAndView ViewCenter(HttpServletRequest request, 
HttpServletResponse response) {
         ModelAndView mv = new ModelAndView();
         List<VaccinationCenter> centers = 
vaccinationService.getAllCenters(); // Retrieve all centers from the 
service
         mv.addObject("centers", centers); // Pass the list of centers 
to the view
         mv.setViewName("viewCenter");
         return mv;
        }
     }

package com.VaccinationCenter.Controller;
import java.util.ArrayList;
import java.util.List;
import java.util.logging.Logger;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.servlet.ModelAndView;
import org.springframework.web.servlet.mvc.support.RedirectAttributes;

import com.VaccinationCenter.Entities.Citizen;
import com.VaccinationCenter.Entities.VaccinationCenter;
import com.VaccinationCenter.Service.CitizenService;
import com.VaccinationCenter.Service.VaccinationCenterService;

@Controller
@RequestMapping("/admin/vaccination")
public class VaccinationCenterController {

     @Autowired
     VaccinationCenterService vaccinationService;
     @Autowired
     CitizenService citizenService;

     @RequestMapping("/addCenter")
     public ModelAndView AddNewCenter(HttpServletRequest request,
HttpServletResponse response) {
           ModelAndView mv = new ModelAndView();
           mv.setViewName("addCenter");
           return mv;
     }

     @RequestMapping("/viewCenterList")
     public ModelAndView viewCenter() {
         ModelAndView mv = new ModelAndView();

         List<VaccinationCenter> vaccinationCenters =
vaccinationService.getAllCenters();

         // Separate vaccination centers into different categories
         List<VaccinationCenter> activeCenters = new ArrayList<>();
         List<VaccinationCenter> inactiveCenters = new ArrayList<>();

         for (VaccinationCenter center : vaccinationCenters) {
             String status = center.getStatus();
             if (status == null) {
            inactiveCenters.add(center);
             }
         }
         mv.setViewName("viewCenter");
         mv.addObject("centers", inactiveCenters);
         int centerCount =  inactiveCenters.size();
        mv.addObject("centerCount", centerCount > 0 ? centerCount :
0);
         return mv;
     }
```

```
      @RequestMapping("/viewCenterById")
        public ModelAndView viewCenterById(@RequestParam("id") int id,
@RequestParam("centerName") String centerName, RedirectAttributes
redirectAttributes) {
            ModelAndView mv = new ModelAndView();
            VaccinationCenter center = vaccinationService.getbyid(id);

            if (center != null) {
                List<Citizen> citizens =
citizenService.getAllCitizens();
                mv.addObject("center", center);
                mv.addObject("citizens", citizens);
                mv.setViewName("viewCenterById");
            } else {
              redirectAttributes.addFlashAttribute("message","ID IS NOT
AVAIABLE");

mv.setViewName("redirect:/admin/vaccination/viewCenter");
            }

            return mv;
        }

     @RequestMapping("/insertCenter")
     public ModelAndView insertCenter(HttpServletRequest request,
HttpServletResponse response) {
           ModelAndView mv = new ModelAndView();
           VaccinationCenter center = new VaccinationCenter();
           String centerName = request.getParameter("centerName");
           String city = request.getParameter("city");



           if (centerName == null || centerName.isEmpty() &&
city.isEmpty()) {
                mv.setViewName("addCenter");
                mv.addObject("message", "Failed to save Center");

           } else {



                 try {
                      center.setCenterName(centerName);
```

```
                       center.setCity(city);
                       vaccinationService.AddCenter(center);

mv.setViewName("redirect:/admin/vaccination/viewCenterList");
                       VaccinationCenter vaccinationCenter =
vaccinationService.getCenterByName(centerName);
                       mv.addObject(vaccinationCenter);
                   } catch (Exception e) {
                       mv.setViewName("addCenter");
                       mv.addObject("message", "Failed to save
Center");
                   }
           }

           return mv;
      }


        @RequestMapping("/editCenter")
         public ModelAndView editCenter(@RequestParam("id") int id
,HttpServletRequest request, HttpServletResponse response) {
              ModelAndView mv = new ModelAndView();
             VaccinationCenter center =
vaccinationService.getbyid(id);
             mv.addObject("center", center);
             mv.setViewName("editCenter");
             return mv;
         }

         @RequestMapping("/deleteCenter")
         public ModelAndView deleteCenter(@RequestParam("id") int id,
HttpServletRequest request, HttpServletResponse
response,RedirectAttributes redirectAttributes) {
             vaccinationService.deletebyid(id);
             ModelAndView mv = new ModelAndView();

mv.setViewName("redirect:/admin/vaccination/viewCenterList");
             String message = " IS DELETED";
             redirectAttributes.addFlashAttribute(id + message);
             return mv;
         }

         @RequestMapping("/updateCenter")
         public ModelAndView updateCenter(@RequestParam("id") int id
,@RequestParam("centerName") String centerName
```

```
,@RequestParam("city")String city  ,HttpServletRequest request,
HttpServletResponse response,RedirectAttributes redirectAttributes) {
              ModelAndView mv = new ModelAndView();
              VaccinationCenter center =
vaccinationService.getbyid(id);
           if(center!=null ) {
              center.setCenterName(centerName);
              center.setCity(city);
              vaccinationService.updateCenter(center);

mv.setViewName("redirect:/admin/vaccination/viewCenterList");
              String message = centerName+" IS UPDATED";
              redirectAttributes.addFlashAttribute("message"+ message);
              }else {

mv.setViewName("redirect:/admin/vaccination/editCenter?id=" + id); //
Redirect to the editCenter request mapping with the id parameter
             mv.addObject("message", "Failed to Update Center");
           }
             return mv;
         }
}
package com.VaccinationCenter.Entities;

import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;

import lombok.AllArgsConstructor;
import lombok.Data;
import lombok.NoArgsConstructor;

@Data
@AllArgsConstructor
@NoArgsConstructor
@Entity
public class Admin {
    @Id
    @GeneratedValue(strategy=GenerationType.IDENTITY)
    private Integer id;

    private String name;

    @Column(unique = true)
    private String email;

    @Column(nullable = false)
    private String password;


}
package com.VaccinationCenter.Entities;
import javax.persistence.CascadeType;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.JoinColumn;
import javax.persistence.JoinTable;
import javax.persistence.OneToOne;

import lombok.AllArgsConstructor;
import lombok.Data;
import lombok.NoArgsConstructor;


@Data
@Entity
@AllArgsConstructor
@NoArgsConstructor
public class Citizen {
    @Id
    @GeneratedValue(strategy = GenerationType.AUTO)
    private int id;
    private String name;
    private String city;
    private int doses;
    private String vaccinationStatus;

    @OneToOne(cascade = CascadeType.ALL)
     @JoinTable(name="Citizen_vaccination",
     //joinColumns relation is going to specify the present table
identity
           joinColumns = {@JoinColumn(name="cid",referencedColumnName
="id")},
     //inverseJoinColumns establish the relation with the aggregate
class - foreign key
           inverseJoinColumns =
{@JoinColumn(name="vid",referencedColumnName ="id")}
                )
```

```java
    private VaccinationCenter center;
    @Override
    public String toString() {
        return "Citizen [name=" + name + ", city=" + city + "]";
    }
}

package com.VaccinationCenter.Entities;

import java.util.List;

import javax.persistence.CascadeType;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.JoinColumn;
import javax.persistence.JoinTable;
import javax.persistence.OneToOne;

import lombok.AllArgsConstructor;
import lombok.Data;
import lombok.NoArgsConstructor;

@Data
@Entity
@AllArgsConstructor
@NoArgsConstructor
public class VaccinationCenter {
     @Id
     @GeneratedValue(strategy = GenerationType.AUTO)
     private int id;
     private String centerName;
     private String city;
     //check status
     private String status;

     @OneToOne(mappedBy = "center")
     private Citizen citizen;

     @Override
     public String toString() {
         return "VaccinationCenter [centerName=" + centerName + "]";
     }

}
```

```java
package com.VaccinationCenter.Repository;

import java.util.List;

//import java.util.List;

import org.springframework.data.jpa.repository.Query;
import org.springframework.data.repository.CrudRepository;
import org.springframework.data.repository.query.Param;

import com.VaccinationCenter.Entities.Admin;
public interface AdminRepository extends CrudRepository<Admin,
Integer> {


    public Admin findByName(String name);

      @Query("SELECT r FROM Admin r WHERE r.email = :email AND
r.password = :password")
      public boolean checkLoginRepo(@Param("email")String email, String
password);

      Admin findByEmailAndPassword(String email, String password);

      List<Admin> findByEmail(String email);

}

package com.VaccinationCenter.Repository;

import java.util.List;

import org.springframework.data.jpa.repository.JpaRepository;

import com.VaccinationCenter.Entities.Citizen;

public interface CitizenRepository extends JpaRepository<Citizen,
Integer> {

      List<Citizen> findByCenterCenterName(String centerName);

}
package com.VaccinationCenter.Repository;

import java.util.List;
import org.springframework.data.repository.CrudRepository;


import com.VaccinationCenter.Entities.VaccinationCenter;

public interface VaccinationCenterRepository extends
CrudRepository<VaccinationCenter, Integer> {

      List<VaccinationCenter> findByCenterName(String centerName);

      List<VaccinationCenter> findAll();
}
package com.VaccinationCenter.Service;

import java.util.List;
import java.util.logging.Logger;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;
import org.springframework.web.bind.annotation.RequestParam;

import com.VaccinationCenter.Entities.Admin;
import com.VaccinationCenter.Repository.AdminRepository;


@Service
public class AdminService {


      @Autowired
      AdminRepository adminRepo ;

      public void addAdmin(Admin admin){
         adminRepo.save(admin);
    }

      public boolean checkLogin(@RequestParam("email") String email ,
@RequestParam("password") String password) {
            return adminRepo.checkLoginRepo(email, password);
    }

      public Admin getAdminByEmail(String email) {
         List<Admin> admins = adminRepo.findByEmail(email);
         if (!admins.isEmpty()) {
             return admins.get(0); // Returning the first admin from
the list
```

```
        }
        return null; // No admin found with the given email
    }

     public Admin findByEmailAndPassword(String email,String password)
{
           Admin authenticatedAdmin =
adminRepo.findByEmailAndPassword(email,password);

           return authenticatedAdmin;
     }
}
package com.VaccinationCenter.Service;

import java.util.List;
import java.util.logging.Logger;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import com.VaccinationCenter.Entities.Citizen;
import com.VaccinationCenter.Entities.VaccinationCenter;
import com.VaccinationCenter.Repository.CitizenRepository;

@Service
public class CitizenService {
        @Autowired
         private CitizenRepository citizenRepository;


        public Citizen saveCitizen(Citizen citizen) {
             return citizenRepository.save(citizen);
         }


         public Citizen updateCitizen(Citizen citizen) {
             return citizenRepository.save(citizen);
         }


         public List<Citizen> getAllCitizens() {
             return citizenRepository.findAll();
         }

         public void addCitizen(Citizen citizen) {
             citizenRepository.save(citizen);
         }
         public Citizen getCitizenById(int id) {
             return citizenRepository.findById(id)
                     .orElseThrow(() -> new
IllegalArgumentException("Invalid citizen Id:" + id));
         }


         public void deleteCitizenById(int id) {
             Citizen citizen = getCitizenById(id);
             citizenRepository.delete(citizen);
         }

         public Citizen getCitizensCenterByCenterName(String
centerName) {
                List<Citizen> citizen =
citizenRepository.findByCenterCenterName(centerName);
             if (!citizen.isEmpty()) {
                 return citizen.get(0); // Returning the first admin
from the list

             }
             return null; // No admin found with the given email
           }
}
package com.VaccinationCenter.Service;

import java.util.List;
import java.util.Optional;
import java.util.logging.Logger;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import com.VaccinationCenter.Entities.VaccinationCenter;
import com.VaccinationCenter.Repository.VaccinationCenterRepository;

@Service
public class VaccinationCenterService {


     @Autowired
     VaccinationCenterRepository vaccinationRepo;
     public void AddCenter(VaccinationCenter center){
        vaccinationRepo.save(center);
    }
     public VaccinationCenter getCenterByName(String centerName) {
           List<VaccinationCenter> centers = 
vaccinationRepo.findByCenterName(centerName);
        if (!centers.isEmpty()) {
            return centers.get(0); // Returning the first vaccination 
center from the list

        }
        return null; // no vaccination center found.
     }

      public List<VaccinationCenter> getAllCenters() {
             return vaccinationRepo.findAll(); // Retrieve all centers
         }


     public VaccinationCenter getbyid(int id) {

           Optional<VaccinationCenter> optionalCenter = 
vaccinationRepo.findById(id);
        return optionalCenter.orElse(null);

     }
     public void deletebyid(int id) {

           vaccinationRepo.deleteById(id);
     }
     public void updateCenter(VaccinationCenter center) {
        vaccinationRepo.save(center);
    }
}
```

```
server.port=8080
spring.jpa.hibernate.ddl-auto=update
spring.datasource.url=jdbc:mysql://localhost:3306/vaccination
spring.datasource.username=root
spring.datasource.password=Nisha@1623

logging.level.org.springframework.web: DEBUG
spring.mvc.view.prefix=/pages/
spring.mvc.view.suffix=.jsp
```

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
        <%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"
%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>ADD VACCINATION CENTER</title>
</head>
<body>

<form action="insertCenter">
<h2 align="center" ><b>ADD NEW VACCINATION CENTER</b></h2>
      <table class="container">

           <tbody>

                 <tr>
                      <td><label for="centerName">CENTER NAME:
</label></td>
                      <td><input type="text" name="centerName"
placeholder="ENTER CLINIC NAME" ><br></td>

                 </tr>
                 <tr>
                 <td><label for="city">SELECT CITY:</label></td>
               <td><select name="city" class="box" >
                 <option value="">--Please choose an city--</option>
              <option value="Bengalore">Bangalore</option>
              <option value="Chennai">Chennai</option>
              <option value="Hyderbad">Hyderbad</option>
              <option value="Pune">Pune</option>
              <option value="Madurai">Madurai</option>
              <option value="Punjab">Punjab</option>
              <option value="Ladak">Ladak</option>
              <option value="Coimbatore">Coimbatore</option>
            </select></td>
                 </tr>
           </tbody>
           <tfoot>
                 <tr>
                      <th></th>
                      <th><input type="submit" value="Submit"/></th>
```

```
                </tr>
           </tfoot>
      </table>

</form>
<div class="error-message" align="center">
    <c:if test="${not empty message}">
        <span style="color: red;">${message}</span>
    </c:if>
</div>
</body>
</html>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html>
<html>
<head>
    <meta charset="ISO-8859-1">
    <title>Add citizen</title>

</head>
<body>


<form action="insertCitizen" method="POST">
    <h2 align="center"><b>ADD NEW CITIZEN</b></h2>
    <table class="container">

        <tbody>

        <tr>
            <td><label for="name">Citizen Name:</label></td>
            <td><input type="text" name="name" placeholder="ENTER
NAME"><br></td>

        </tr>
        <tr>
            <td><label for="city">CITIZEN CITY:</label></td>
            <td>
                <select name="city" id="city">
                    <option value="">--Please choose a city--</option>
                    <option value="Bangalore">Bangalore</option>
                    <option value="Chennai">Chennai</option>
                    <option value="Hyderabad">Hyderabad</option>
                    <option value="Pune">Pune</option>
                    <option value="Madurai">Madurai</option>
                    <option value="Punjab">Punjab</option>
                    <option value="Ladakh">Ladakh</option>
                    <option value="Coimbatore">Coimbatore</option>
                </select>
            </td>
        </tr>
        <tr>
            <td><label for="centerName">CITIZEN CLINIC:</label></td>
             <td>
                <select name="centerName" id="centerName">
                    <option value="">--Please choose a clinic--
</option>
                    <c:forEach items="${centers}" var="center">
                        <option
value="${center.centerName}">${center.centerName}</option>
                    </c:forEach>
                </select>
            </td>
        </tr>
        </tbody>
        <tfoot>
        <tr>
            <th></th>
            <th><input type="submit" value="Submit"/></th>
        </tr>
        </tfoot>
    </table>

    <div class="error-message" align="center">
    <c:if test="${not empty message}">
        <span style="color: red;">${message}</span>
    </c:if>
</div>
</form>
</body>
</html>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
     <%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Dashboard</title>
</head>
```

```
<body>

<div align="right">
       <% String userid = (String) session.getAttribute("userid");%>
       <p align="Right">User Name:<span style="color:#FF0066"> <%=
userid %></span><p>
       <a href="<c:url value='/admin/logout' />">logout</a>
</div>

                  <h1 align="center">VACCINATION CENTER | WELCOME</h1>

<hr><br>
    <div class="link-container">
            <ul>
                <li><a href="<c:url value='/admin/citizen/viewCitizen'
/>?userid=${userid}">Citizens</a></li>
                <li><a href="<c:url
value='/admin/vaccination/viewCenterList'
/>?userid=${userid}">Vaccination Centers</a></li>
           </ul>
     </div><br>
</body>
</html>
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Edit Center</title>

</head>
<body>
                              <h2 align="center" ><b>Edit
Center</b></h2>
   <form action="updateCenter" method="post">
        <input type="hidden" name="id" value="${center.id}">
        <table class="container">

           <tbody>

                <tr>
                      <td><label for="centerName">CENTER NAME:
</label></td>
```

```
                      <td><input type="text" name="centerName"
value="${center.centerName}" placeholder="ENTER CLINIC NAME"
><br></td>
                </tr>
                <tr>
                <td><label for="city">SELECT CITY:</label></td>
               <td><select name="city" id="city">
                 <option value="city">${center.city}</option>
              <option value="Bengalore">Bangalore</option>
              <option value="Chennai">Chennai</option>
              <option value="Hyderbad">Hyderbad</option>
              <option value="Pune">Pune</option>
              <option value="Madurai">Madurai</option>
              <option value="Punjab">Punjab</option>
              <option value="Ladak">Ladak</option>
              <option value="Coimbatore">Coimbatore</option>
            </select></td>
                </tr>
           </tbody>
           <tfoot>
                <tr>
                      <td></td>
                      <td><input type="submit" value="Update"/></td>

                </tr>
           </tfoot>
      </table>

  </form>


<div class="error-message" align="center">
    <c:if test="${not empty message}">
        <span style="color: red;">${message}</span>
    </c:if>
</div>

</body>
</html>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
       <%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"
%>
<!DOCTYPE html>
<html>
<head>
```

```
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>


<h1 align="center">EDIT CITIZEN</h1>
<form action="updateCitizen">
        <input type="hidden" name="id" value="${citizen.id}">

     <table class="container">

           <tbody>

                <tr>
                      <td><label for="name">Citizen Name:</label></td>
                      <td><input type="text" name="name"
value="${citizen.name}" placeholder="ENTER NAME" ><br></td>

                </tr>
                <tr>
                <td><label for="city">CITIZEN CITY:</label></td>
               <td><select name="city" id="city">
                <option
value="${citizen.city}">${citizen.city}</option>

              <option value="Bengalore">Bangalore</option>
              <option value="Chennai">Chennai</option>
              <option value="Hyderbad">Hyderbad</option>
              <option value="Pune">Pune</option>
              <option value="Madurai">Madurai</option>
              <option value="Punjab">Punjab</option>
              <option value="Ladak">Ladak</option>
              <option value="Coimbatore">Coimbatore</option>
            </select></td>
                </tr>

                <tr>
                <td><label for="doses">No. Of Doses</label></td>
               <td><select name="doses" id="doses">
               <option
value="${citizen.doses}">${citizen.doses}</option>
                 <option value="0">None</option>
              <option value="1">1</option>
              <option value="2">2</option>
            </select></td>
                 </tr>
                 <tr>
                 <td><label for="centerName">CLINIC:</label></td>
               <td>
                <select name="centerName" id="centerName">
                     <option
value="${citizen.center.centerName}">${citizen.center.centerName}</opt
ion>
                    <c:forEach items="${centers}" var="center">

                        <option
value="${center.centerName}">${center.centerName}</option>
                    </c:forEach>
                </select>
            </td>
                 </tr>
           </tbody>
           <tfoot>
                 <tr>
                      <th></th>
                      <th><input type="submit" value="Submit"/></th>

                 </tr>
           </tfoot>
      </table>

      <div class="error-message" align="center">
    <c:if test="${not empty message}">
        <span style="color: red;">${message}</span>
    </c:if>
</div>
</form>
</body>
</html>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
     <%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Login Page</title>

</head>
<body>
```

```
<header><h1>VACCINATION CENTER LOGIN</h1></header>
<hr>
<form action="dashboard" method="post">

<h2><b>Login</b></h2>
      <table>

            <tbody>
                  <tr>
                        <td><label for="email">Email:</label></td>
                        <td><input type="email" name="email"
placeholder="email" ><br></td>
                  </tr>
                  <tr>
                     <td><label for="password">Enter
Password:</label></td>
                        <td><input type="password" name= "password"
placeholder="Enter Password"><br></td>

                  </tr>
            </tbody>
            <tfoot>
                  <tr>
                        <td></td><td><input type="submit" value="LOGIN"
/></td>
                  </tr>
            </tfoot>
      </table>

</form>

<div class="error-message">
    <c:if test="${not empty message}">
        <span style="color: red;">${message}</span>
    </c:if>
</div>

<div>
<form action="register">
      <button>REGISTER</button>
</form>
</div>
</body>
</html>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
    <%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Registeration</title>

</head>
<body>

<form action="insert" method="post">
<h2 align="center" ><b>REGISTRATION</b></h2>
      <table class="container">

           <tbody>
                 <tr>
                      <td><label for="name">NAME:</label></td>
                      <td><input name="name" placeholder="Your Name"
type="text"/><br></td>
                 </tr>
                 <tr>
                      <td><label for="email">EMAIL:</label></td>
                      <td><input type="email" name="email"
placeholder="email" ><br></td>

                 </tr>
                 <tr>

                      <td><label for="password">PASSWORD:</label></td>
                      <td><input type="password" name= "password"
placeholder="Enter Password"><br></td>
                 </tr>
           </tbody>
           <tfoot>
                 <tr>
                      <th></th>
                      <th><input type="submit" value="Submit"/></th>

                 </tr>
           </tfoot>
      </table>

</form>
<div class="error-message" align="center">
    <c:if test="${not empty message}">
        <span style="color: red;">${message}</span>
    </c:if>
</div>
</body>
</html>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1" %>
<!DOCTYPE html>
  <%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<html>
<head>
<meta charset="ISO-8859-1">
<title>VIEW ALL CENTER</title>

</head>
<body>
                    <%@include file="dashboard.jsp" %>

             <h2 align="center"><i>VACCINATION CENTER</i></h2>

    <table>
        <thead>
        <tr>
        <td colspan="4">
        <form id="form-1" action="addCenter">
            <input type="submit" value="ADD NEW CENTER" id="button-1"
/>
        </form>
        </td>
    </tr>
        </thead>
        <tbody>
        <tr>
            <th>ID</th>
            <th>NAME</th>
            <th>City</th>
            <th>Action</th>
        </tr>
        <c:forEach items="${centers}" var="center">
            <tr>
                <td>${center.id}</td>
                <td>${center.centerName}</td>
                <td>${center.city}</td>
                <td class="view-edit-delete">
                <a href="<c:url value='viewCenterById'
/>?id=${center.id}&centerName=${ center.centerName}">View</a>
```

```
                <a href="<c:url
value='editCenter?id=${center.id}'/>">Edit</a>
                <a href="<c:url
value='deleteCenter?id=${center.id}'/>">Delete</a>
                </td>
            </tr>
        </c:forEach>
        </tbody>
       <tfoot>
        <tr>
            <td colspan="4">Total ${centerCount} Vaccination Centers
found.</td>
       </tr>
       </tfoot>
    </table>

    <div class="error-message" align="center">
    <c:if test="${not empty message}">
        <span style="color: red;">${message}</span>
    </c:if>
</div>
</body>
</html>

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html>
<html>
<head>
    <meta charset="ISO-8859-1">
    <title>VIEW CITIZEN BY ID</title>

</head>
<body>
<%@include file="dashboard.jsp" %>

    <h1 class="head1" align="center">CENTER INFORMATION</h1>
    <table border="1" align="center">
        <!-- Display center details -->
        <tr>
            <td>ID</td>
            <td>${center.id}</td>
        </tr>
        <tr>
            <td>Name</td>
            <td>${center.centerName}</td>
        </tr>
        <tr>
            <td>City</td>
            <td>${center.city}</td>
        </tr>
            </table>
            <h2 align="center">ALL CITIZEN</h2>
            <table border="1" align="center">
        <!-- Iterate over citizens list -->
        <!-- Display citizen details  -->
        <c:forEach items="${citizens}" var="citizen" >
            <c:if test="${center.centerName ==
citizen.center.centerName}">
                <tr>
                    <td>ID</td>
                    <td>${citizen.id}</td>
                </tr>
                <tr>
                    <td>Name</td>
                    <td>${citizen.name}</td>
                </tr>
                <tr>
                <td>Action</td>
                 <td><a href="<c:url
value='/admin/citizen/viewCitizenById?id=${citizen.id}'/>">View</a></t
d>
                </tr>
            </c:if>
            <br>
        </c:forEach>
</table>

    <!-- Error message display -->
<div class="error-message" align="center">
    <c:if test="${not empty message}">
        <span style="color: red;">${message}</span>
    </c:if>
</div>
</body>
</html>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
      <%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html>
<html>
```

```
<head>
<meta charset="ISO-8859-1">
<title>VIEW CITIZEN</title>
</head>
<body>
     <!-- include dashboard -->
          <%@include file="dashboard.jsp" %>


       <h2 align = "center"><i>CITIZENS</i></h2>

        <table >
         <thead>
        <tr>
        <td colspan="7">
       <form id="form-1" action="addCitizen" method="post">
            <input type="submit" value="ADD NEW CITIZENS" id="button-
1"/>
        </form>
        </td>
    </tr>
        </thead>
        <tbody>
        <tr>
            <th>ID</th>
            <th>NAME</th>
            <th>CITY</th>
            <th>NO. OF DOSES</th>
            <th>VACCINATION STATUS</th>
            <th>VACCINATION CENTER</th>
            <th>ACTION</th>
        </tr>
        <c:forEach items="${citizens}" var="citizen" varStatus="loop">
            <tr>
            <td>${citizen.id}</td>
            <td>${citizen.name}</td>
            <td>${citizen.city}</td>
            <td>${citizen.doses}</td>
            <td>${citizen.vaccinationStatus}</td>
            <td>${citizen.center.centerName}</td>
                <td class="view-edit-delete">

                <a href="<c:url
value='viewCitizenById?id=${citizen.id}'/>">View</a>
                <a href="<c:url
value='editCitizen?id=${citizen.id}'/>">Edit</a>
```

```
                <a href="<c:url
value='deleteCitizen?id=${citizen.id}'/>">Delete</a>

                </td>
            </tr>
        </c:forEach>
        </tbody>
        <tfoot>
        <tr>
            <td colspan="7">  Total ${citizenCount} citizens
found.</td>
        </tr>
        </tfoot>
   </table>

<div class="error-message" align="center">
    <c:if test="${not empty message}">
        <span style="color: red;">${message}</span>
    </c:if>
</div>




</body>
</html>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>

<!DOCTYPE html>
<html>
<head>
    <meta charset="ISO-8859-1">
    <title>VIEW CITIZEN BY ID</title>

</head>
<body>

<%@include file="dashboard.jsp" %>

    <h1 align="center">Citizen Information</h1>
    <table >
        <tr>
            <td>ID</td>
            <td>${citizen.id}</td>
        </tr>
        <tr>
            <td>NAME</td>
            <td>${citizen.name}</td>
        </tr>
        <tr>
            <td>CITY</td>
            <td>${citizen.city}</td>
        </tr>
        <tr>
            <td>NO. OF DOSES</td>
            <td>${citizen.doses}</td>
        </tr>
        <tr>
            <td>VACCINATION STATUS</td>
            <td>${citizen.vaccinationStatus}</td>
        </tr>
        <tr>
            <td>VACCINATION CENTER</td>
             <td >
             <a href="<c:url value='/admin/vaccination/viewCenterById'
/>?id=${citizen.center.id}&centerName=${
citizen.center.centerName}">${citizen.center.centerName}</a>
             </td>
        </tr>
    </table>


  <div class="error-message" align="center">
    <c:if test="${not empty message}">
        <span style="color: red;">${message}</span>
    </c:if>
</div>
</body>
</html>
```