

Vaccination Center

1. addCitizens.jsp:

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%> <!--taglib directive for <forEach> tag -->
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%> <!-- this taglib for "form:form"
tags -->
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
</head>
<body>
    <div class="jumbotron">
        <c:choose>
            <c:when test="${not empty addCitizen.id}">
                <h1 class="text-center">Edit Citizen</h1>
            </c:when>
            <c:otherwise>
                <!-- Render ID field only for update operation -->
                <h1 class="text-center">Add New Citizen</h1>
            </c:otherwise>
        </c:choose>
    </div>
    <div class="row">
        <div class="col-md-4"></div>
        <div class="col-md-4">
            <form:form method="POST" action="addNewCitizen"
modelAttribute="addCitizen" id="addCitizenForm"> <!-- action="branch" mapping should be same in
controller class also -->
                <form:input type="hidden" name="id" path="id" value="${addCitizen.id}"
class="form-control" />

                Citizen Name:<form:input path="name" class="form-control" />
                Citizen City:<form:input path="city" class="form-control" />
                Number of Doses:<form:input path="noOfDoses"

class="form-control" />

                <u>Vaccination</u> Status:<form:input path="vaccinationStatus"

class="form-control" />

                <u>Vaccination</u> Center:<form:input path="vaccinationCenter"

class="form-control" />

                <!-- Path variable (hidden input) -->
```

```

        <c:choose>
        <c:when test="\${not empty addCitizen.id}">
        <input type=submit name="modify" class="btn btn-warning"
value="Submit" />
        </c:when>
        <c:otherwise>
        <!-- Render ID field only for update operation -->
        <div>
                <input type=submit name="add" class="btn btn-success"
value="Submit" />
        </div>
        </c:otherwise>
        </c:choose>
        </form:form>
    </div>
    <div class="col-md-4"></div>
</div>

<script>
function prepareModify() {
var id = document.getElementById("id").value; // Assuming bid field contains the ID value
document.getElementById("id").value = id; // Set the ID value in the hidden input
document.getElementById("addVaccinationCenterForm").submit(); // Submit the form
}
</script>
</body>
</html>

```

2. addNewVaccinationCenter.jsp:

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%> <!--taglib directive for <forEach> tag -->
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%> <!-- this taglib for "form:form"
tags -->
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
</head>
<body>
    <div class="jumbotron">
        <c:choose>
        <c:when test="\${not empty addVaccinationCenter.id}">
            <h1 class="text-center">Edit ${addVaccinationCenter.name} </h1>

```

```

</c:when>
<c:otherwise>
<!-- Render ID field only for update operation -->
        <h1 class="text-center">Add New Vaccination Center</h1>

</c:otherwise>
</c:choose>
</div>
<div class="row">
    <div class="col-md-4"></div>
    <div class="col-md-4">
        <form:form method="POST" action="addNewVaccinationCenter"
modelAttribute="addVaccinationCenter" id="addVaccinationCenterForm"> <!-- action="branch" mapping
should be same in controller class also -->
            <c:choose>
                <c:when test="{not empty addVaccinationCenter.id}">
                    <form:input type="hidden" name="id" path="id"
value="{addVaccinationCenter.id}" class="form-control" />
                    </c:when>
                    <c:otherwise>
                        <!-- Render ID field only for update operation -->
                        <div>

                                </div>
                            </c:otherwise>
                        </c:choose>

                        Center Name:<form:input path="name" class="form-control" />
                        Center City:<form:input path="city" class="form-control" />

                        <!-- Path variable (hidden input) -->
                        <c:choose>
                            <c:when test="{not empty addVaccinationCenter.id}">
                                <input type="submit" name="modify" class="btn btn-warning"
value="Submit" />
                            </c:when>
                            <c:otherwise>
                                <!-- Render ID field only for update operation -->
                                <div>

                                        <input type="submit" name="add" class="btn btn-success"
value="Submit" />
                                    </div>
                                </c:otherwise>
                            </c:choose>

                        </form:form>
                    </div>
                <div class="col-md-4"></div>
            </div>

</div>

<script>

```

```

function prepareModify() {
var id = document.getElementById("id").value; // Assuming bid field contains the ID value
document.getElementById("id").value = id; // Set the ID value in the hidden input
    document.getElementById("addVaccinationCenterForm").submit(); // Submit the form
}
</script>
</body>
</html>

```

3. citizens.jsp:

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%> <!--taglib directive for <forEach> tag -->
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%> <!-- this taglib for "form:form"
tags -->
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Vaccination Center</title>
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
</head>
<body>
    <div class="row">
        <div class="col-md-2"></div>
        <div class="col-md-8"><br>
            <table class="table table-bordered table-striped table-hover">
                <thead style="text-decoration:underline">
                    <th><a href="/citizens">Citizens</a></th>
                    <th><a href="/vaccinationcenter">Vaccination Centers</a></th>
                    <th><a href="/login">Logout</a></th>
                    <th>Welcome, ${user.name}</th>
                </thead>
                <tr>
                    <td></td>
                    <td></td>
                    <td></td>
                    <td></td>
                </tr>
            </table>
        </div>
        <div class="col-md-2"></div>
    </div>
    <br>
    <br>
    <h2>Citizens</h2> <br>
    <div class="row">
        <div class="col-md-2"></div>
        <div class="col-md-8">

```

```

<table class="table table-bordered table-striped table-hover">
<thead>
    <th>ID</th>
    <th>Name</th>
    <th>City</th>
    <th>No. of Doses</th>
    <th>Vaccination Status</th>
    <th>Vaccination Center</th>
    <th>Actions</th>
</thead>
<tbody>
<tr style="text-align:right">
<!--          <input type=submit name="add" class="btn btn-success" value="Add" />
-->
                <a href="/addCitizens" class="btn btn-primary">Add New
Citizens</a>

</tr>
<tr></tr>
        <c:forEach var="c" items="${citizensList}">
            <tr>
                <td> ${c.id}</td>
                <td> ${c.name}</td>
                <td> ${c.city}</td>
                <td> ${c.noOfDoses}</td>
                <td> ${c.vaccinationStatus}</td>
                <td> ${c.vaccinationCenter}</td>

                <td>
                    <a href="/selectCitizen?cid=${c.id }">Edit</a>
                    <a href="/deleteCitizen?cid=${c.id }">Delete</a>
                    <a href="/viewCitizen?cid=${c.id }">View</a>
                </td>
            </tr>
        </c:forEach>
    </tbody>
</table>

</div>
<div class="col-md-2"></div>
</div>

</body>
</html>

```

4. index.jsp:

```

<a href="/registration">admin Registration</a>

```

```

<br></br>
<a href="/login">admin login</a>
<br></br>
<br></br>

```

5. login.jsp:

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%> <!--taglib directive for <forEach> tag -->
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%> <!-- this taglib for "form:form"
tags -->
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
</head>
<body>
    <div class="jumbotron">
        <h1 class="text-center">Login Form</h1>
    </div>
    <div class="row">
        <div class="col-md-4"></div>
        <div class="col-md-4">
            <form:form method="POST" action="/login" modelAttribute="user"
id="loginForm">
                UserName:<form:input path="name" class="form-control" />
                Password:<form:input path="password" class="form-control" />
<br/>
                <!-- Path variable (hidden input) -->
                <input type="submit" name="login" class="btn btn-success"
value="login" />
                <a href="/registration" class="btn btn-primary">Register</a>
            </form:form>
        </div>
    </div>
</body>
</html>

```

6. registration.jsp:

```

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

```

```

<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%> <!-- this taglib for "form:form"
tags -->
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
</head>
<body>
    <div class="jumbotron">
        <h1 class="text-center">Admin Registration Form</h1>
    </div>
    <div class="row">
        <div class="col-md-4"></div>
        <div class="col-md-4">
            <form:form method="POST" action="registration" modelAttribute="user"
id="registrationForm"> <!-- action="branch" mapping should be same in controller class also -->
                Name:<form:input path="name" class="form-control" />
                Email:<form:input path="email" class="form-control" />
                Password:<form:input path="password" class="form-control" />
<br/>

                <!-- Path variable (hidden input) -->
                <input type="submit" name="registration" class="btn btn-success"
value="Register" />

            </form:form>
        </div>
    </div>
</body>
</html>

```

7. success.jsp:

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%> <!--taglib directive for <forEach> tag -->
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%> <!-- this taglib for "form:form"
tags -->
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Success Page</title>
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">

```

```

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
</head>
<body>
    <div class="jumbotron">
        <h1 class="text-center">registration successfull </h1>
    </div>
</body>
</html>

```

8. vaccinationcenter.jsp:

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%> <!--taglib directive for <forEach> tag -->
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%> <!-- this taglib for "form:form"
tags -->
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Vaccination Center</title>
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
</head>
<body>
    <div class="row">
        <div class="col-md-2"></div>
        <div class="col-md-8"><br>
            <table class="table table-bordered table-striped table-hover">
                <thead style="text-decoration:underline">
                    <th><a href="/citizens">Citizens</a></th>
                    <th><a href="/vaccinationcenter">Vaccination Centers</a></th>
                    <th><a href="/login"> Logout</a></th>
                    <th><u>Welcome, ${user.name}</th>
                </thead>
                <tr>
                </tr>
            </table>
        </div>
        <div class="col-md-2"></div>
    </div>
    <br>
    <br>
    <h2>Centers</h2> <br>
    <div class="row">
        <div class="col-md-2"></div>

```



```

<div class="col-md-8">
  <table class="table table-bordered table-striped table-hover">
    <thead>
      <th>ID</th>
      <th>Name</th>
      <th>City</th>
      <th>Actions</th>
    </thead>
    <tbody>
      <tr style="text-align:right">
        <a href="/addVaccinationCenter" class="btn btn-primary">Add
new Center</a>
      </tr>
      <tr></tr>
      <c:forEach var="v" items="{vaccinationCentersList}">
        <tr>
          <td> ${v.id}</td>
          <td> ${v.name}</td>
          <td> ${v.city}</td>
          <td>
            <a href="/select?vid=${v.id }">Edit</a>
            <a href="/delete?vid=${v.id }">Delete</a>
            <a href="/view?vid=${v.id }">View</a>
          </td>
        </tr>
      </c:forEach>
    </tbody>
  </table>
</div>
<div class="col-md-2"></div>
</div>

</body>
</html>

```

9. viewCitizen.jsp:

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%> <!--taglib directive for <forEach> tag -->
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%> <!-- this taglib for "form:form"
tags -->
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">

```

```

<title>View Citizens</title>
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
</head>
<body>
    <div class="jumbotron">
        <h1 class="text-center">View Citizens</h1>
    </div>
    <div class="row">
        <div class="col-md-2"></div>
        <div class="col-md-8">
            <table class="table table-bordered table-striped table-hover">
                <tbody>
                    <tr>
                        <td colspan="2" style="text-align:center"> <h3>${viewCitizen.name}</h3></td>
                    </tr>
                    <tr>
                        <th>ID</th>
                        <td>${viewCitizen.id}</td>
                    </tr>
                    <tr>
                        <th>City</th>
                        <td>${viewCitizen.city}</td>
                    </tr>
                    <tr>
                        <th>Number of Vaccines</th>
                        <td>${viewCitizen.noOfDoses}</td>
                    </tr>
                    <tr>
                        <th>Vaccination Status</th>
                        <td>${viewCitizen.vaccinationStatus}</td>
                    </tr>
                    <tr>
                        <th>Allocated Vaccination Center</th>
                        <td>${viewCitizen.vaccinationCenter}</td>
                    </tr>
                </tbody>
            </table>
        </div>
        <div class="col-md-2"></div>
    </div>
</body>
</html>

```

10. viewVaccinationCenter.jsp:

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>

```

```

<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%> <!--taglib directive for <forEach> tag -->
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form"%> <!-- this taglib for "form:form"
tags -->
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>View Vaccination Center</title>
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
</head>
<body>
    <div class="jumbotron">
        <h1 class="text-center">View Vaccination Center</h1>
    </div>
    <div class="row">
        <div class="col-md-2"></div>
        <div class="col-md-8">
            <table class="table table-bordered table-striped table-hover">
                <thead>
                    <th>ID</th>
                    <th>City</th>
                    <th>Name</th>
                    <th></th>
                </thead>

                <tr>
                    <td> ${viewVaccinationCenter.id}</td>
                    <td> ${viewVaccinationCenter.name}</td>
                    <td> ${viewVaccinationCenter.city}</td>
                    <td></td>
                </tr>

            </table>

        </div>
        <div class="col-md-2"></div>
    </div>
</body>
</html>

```

11. citizensControllers.java:

```
package com.example.demo.controller;
```

```
import com.example.demo.dao.citizensDao;
```

```
import com.example.demo.entity.citizens;
```

```
import com.example.demo.entity.vaccinationCenter;
```

```
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.ui.ModelMap;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.ModelAttribute;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.servlet.ModelAndView;
```

```
@Controller
```

```
public class citizensController {
```

```
    private final citizensDao citizensDao;
```

```
    @Autowired
```

```
    public citizensController(citizensDao citizensDao) {
        this.citizensDao = citizensDao;
    }
```

```
    @GetMapping("/citizens")
```

```
    public ModelAndView citizensHomePage() {
        ModelAndView mv = new ModelAndView();
        System.out.println("citizens " + citizensDao.read());
        mv.addObject("citizensList", citizensDao.read());
        mv.addObject("citizens", new citizens());
        mv.setViewName("citizens");
        return mv;
    }
```

```
    @GetMapping("/addCitizens")
```

```
    public ModelAndView addaddCitizens( ) {
        ModelAndView mv = new ModelAndView();
        mv.addObject("addCitizen", new citizens());
    }
```

```

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

```

```

    @GetMapping("/selectCitizen")
    public ModelAndView select(@RequestParam("cid") Long id) {
        ModelAndView mv = new ModelAndView();
        mv.addObject("addCitizen", citizensDao.read(id).orElse(null));
        mv.setViewName("addCitizens");
        return mv;
    }

```

```

    @GetMapping("/viewCitizen")
    public ModelAndView view(@RequestParam("cid") Long id) {
        ModelAndView mv = new ModelAndView();
        mv.addObject("viewCitizen", citizensDao.read(id).orElse(null));
        mv.setViewName("viewCitizen");
        return mv;
    }

```

```

    @GetMapping("/deleteCitizen")
    public ModelAndView delete(@RequestParam("cid") Long id) {
        ModelAndView mv = new ModelAndView();
        citizensDao.delete(id);
        System.out.println("citizens " + citizensDao.read());
        mv.addObject("citizensList", citizensDao.read());
        mv.addObject("citizens", new citizens());
        mv.setViewName("citizens");
        return mv;
    }

```

```

    @PostMapping("/addNewCitizen")
    public ModelAndView processForm(@ModelAttribute("addCitizen")
citizens citizens,
                                     @RequestParam(value
= "add", required = false) String add,

```

```

String modify,
String delete) {
    @RequestParam(value = "modify", required = false)
    @RequestParam(value = "delete", required = false)

    if (add != null) {
        // Process add operation
        citizensDao.create(citizens);
        // ...
    } else if (modify != null) {
        // Process modify operation
        System.out.println("modify citizen center");
        System.out.println(citizens);
        citizensDao.update(citizens);
        // ...
    } else if (delete != null) {
        // Process delete operation
        Long id = citizens.getId(); // Get the 'id' from the branch object
        // Use the "id" parameter to delete the branch with the specified ID
        citizensDao.delete(id);
        // ...
    }
    return citizensHomePage();
}
}

```

12. pacUserController.java:

```

package com.example.demo.controller;

import com.example.demo.dao.pacUserDao;
import com.example.demo.entity.pacUser;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.ui.ModelMap;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.ModelAttribute;
import org.springframework.web.bind.annotation.PostMapping;

```

```

import org.springframework.web.servlet.ModelAndView;

@Controller
public class pacUserController {

    private final pacUserDao pacUserDao;

    @Autowired
    public pacUserController(pacUserDao pacUserDao) {
        this.pacUserDao = pacUserDao;
    }

    @GetMapping("/login")
    public ModelAndView showLoginForm() {
        ModelAndView mv = new ModelAndView();
        mv.addObject("users", pacUserDao.read());
        mv.addObject("user", new pacUser());
        mv.setViewName("login");
        return mv;
    }

    @PostMapping("/login")
    public ModelAndView login(@ModelAttribute("user") pacUser user) {
        ModelAndView mv = new ModelAndView();

        // Check if the user exists in the database
        pacUser existingUser =
pacUserDao.fetchUserByNameAndPassword(user.getName(),
user.getPassword());
        System.out.println("login existing " + existingUser);
        if (existingUser != null) {
            // User authentication successful
            mv.setViewName("vaccinationcenter");
            mv.addObject("message", "Login successful!");
            mv.addObject("user", existingUser);
        } else {
            // User authentication failed

```

```

        mv.setViewName("login");
//        mv.setViewName("vaccinationcenter");
        mv.addObject("error", "Invalid username or password!");
//        return showLoginForm();
    }
    return mv;
}

```

```

@GetMapping("/registration")
public ModelAndView showRegistrationForm() {
    ModelAndView mv = new ModelAndView();
    mv.addObject("users", pacUserDao.read());
    mv.addObject("user", new pacUser());
    mv.setViewName("registration");
    return mv;
}

```

```

@PostMapping("/registration")
public String register(@ModelAttribute("user") pacUser user) {
    ModelAndView mv = new ModelAndView();

    // Check if the username is already taken
    pacUser existingUser = pacUserDao.read(user.getName());
    if (existingUser != null) {
        // Username is already taken
        mv.setViewName("registration");
        mv.addObject("error", "Username already exists!");
    } else {
        // Register the user
        pacUser createdUser = pacUserDao.create(user);
        mv.setViewName("success");
        mv.addObject("message", "Registration successful!");
        mv.addObject("user", createdUser);
    }
    return "success";
}

```


13. vaccinationCenterController.java:

```
package com.example.demo.controller;
```

```
import com.example.demo.dao.vaccinationCenterDao;
```

```
import com.example.demo.entity.pacUser;
```

```
import com.example.demo.entity.vaccinationCenter;
```

```
import org.springframework.beans.factory.annotation.Autowired;
```

```
import org.springframework.stereotype.Controller;
```

```
import org.springframework.ui.ModelMap;
```

```
import org.springframework.web.bind.annotation.GetMapping;
```

```
import org.springframework.web.bind.annotation.ModelAttribute;
```

```
import org.springframework.web.bind.annotation.PostMapping;
```

```
import org.springframework.web.bind.annotation.RequestMapping;
```

```
import org.springframework.web.bind.annotation.RequestMethod;
```

```
import org.springframework.web.bind.annotation.RequestParam;
```

```
import org.springframework.web.servlet.ModelAndView;
```

```
@Controller
```

```
public class vaccinationCenterController {
```

```
    private final vaccinationCenterDao vaccinationCenterDao;
```

```
    @Autowired
```

```
    public vaccinationCenterController(vaccinationCenterDao  
vaccinationCenterDao) {
```

```
        this.vaccinationCenterDao = vaccinationCenterDao;
```

```
    }
```

```
    @GetMapping("/vaccinationcenter")
```

```
    public ModelAndView vaccinationCenterHomePage() {
```

```
        ModelAndView mv = new ModelAndView();
```

```
        System.out.println("vaccinationList " + vaccinationCenterDao.read());
```

```
        mv.addObject("vaccinationCentersList",
```

```
vaccinationCenterDao.read());
```

```
        mv.addObject("vaccinationCenter", new vaccinationCenter());
```

```
        mv.setViewName("vaccinationcenter");  
        return mv;  
    }
```

```
@GetMapping("/addVaccinationCenter")  
public ModelAndView addVaccinationCenter() {  
    ModelAndView mv = new ModelAndView();  
    mv.addObject("addVaccinationCenter", new vaccinationCenter());  
    mv.setViewName("addNewVaccinationCenter");  
    return mv;  
}
```

```
@GetMapping("/select")  
public ModelAndView select(@RequestParam("vid") Long id) {  
    ModelAndView mv = new ModelAndView();  
    mv.addObject("addVaccinationCenter",  
vaccinationCenterDao.read(id).orElse(null));  
    mv.setViewName("addNewVaccinationCenter");  
    return mv;  
}
```

```
@GetMapping("/view")  
public ModelAndView view(@RequestParam("vid") Long id) {  
    ModelAndView mv = new ModelAndView();  
    mv.addObject("viewVaccinationCenter",  
vaccinationCenterDao.read(id).orElse(null));  
    mv.setViewName("viewVaccinationCenter");  
    return mv;  
}
```

```
@GetMapping("/delete")  
public ModelAndView delete(@RequestParam("vid") Long id) {  
    ModelAndView mv = new ModelAndView();  
    vaccinationCenterDao.delete(id);  
    mv.addObject("vaccinationCentersList",  
vaccinationCenterDao.read());  
}
```

```
mv.addObject("vaccinationCenter", new vaccinationCenter());
mv.setViewName("vaccinationcenter");
return mv;
```

```
}
```

```
@PostMapping("/addNewVaccinationCenter")
public ModelAndView
processForm(@ModelAttribute("addVaccinationCenter") vaccinationCenter
vaccinationCenter,
                                                    @RequestParam(value
= "add", required = false) String add,
                                                    @RequestParam(value = "modify", required = false)
String modify,
                                                    @RequestParam(value = "delete", required = false)
String delete) {

    if (add != null) {
        // Process add operation
        vaccinationCenterDao.create(vaccinationCenter);
        // ...
    } else if (modify != null) {
        // Process modify operation
        System.out.println("modify vaccination center");
        System.out.println(vaccinationCenter);

        vaccinationCenterDao.update(vaccinationCenter);
        // ...
    } else if (delete != null) {
        // Process delete operation
        Long id = vaccinationCenter.getId(); // Get the 'id' from the branch
object
        // Use the "id" parameter to delete the branch with the specified ID
        vaccinationCenterDao.delete(id);
        // ...
    }
}
```

```
        return vaccinationCenterHomePage();
    }

}
```

14. citizensDao.java:

```
package com.example.demo.dao;

import java.util.List;
import java.util.Optional;

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

import com.example.demo.entity.pacUser;
import com.example.demo.entity.citizens;
import com.example.demo.repository.citizensRepository;

@Repository
public class citizensDao {

    private final citizensRepository citizensRepository;

    @Autowired
    public citizensDao(citizensRepository citizensRepository) {
        this.citizensRepository = citizensRepository;
    }

    public citizens create(citizens citizens) {
        return citizensRepository.save(citizens);
    }

    public List<citizens> read() {
        return citizensRepository.findAll();
    }
}
```

```

    public Optional<citizens> read(Long id) {
        return citizensRepository.findById(id);
    }

    public citizens update(citizens citizens) {
        return citizensRepository.save(citizens);
    }

    public void delete(Long id) {
        citizensRepository.deleteById(id);
    }
}

```

15. pacUserDao.java:

```

package com.example.demo.dao;

import java.util.List;
import java.util.Optional;

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

import com.example.demo.entity.pacUser;
import com.example.demo.repository.pacUserRepository;

@Repository
public class pacUserDao {

    private final pacUserRepository pacUserRepository;

    @Autowired
    public pacUserDao(pacUserRepository pacUserRepository) {
        this.pacUserRepository = pacUserRepository;
    }
}

```

```

    public pacUser create(pacUser pacUser) {
        return pacUserRepository.save(pacUser);
    }

    public List<pacUser> read() {
        return pacUserRepository.findAll();
    }

    public Optional<pacUser> read(Long id) {
        return pacUserRepository.findById(id);
    }

    public pacUser read(String name) {
        return pacUserRepository.findByName(name);
    }

    public pacUser update(pacUser pacUser) {
        return pacUserRepository.save(pacUser);
    }

    public void delete(Long id) {
        pacUserRepository.deleteById(id);
    }

    public pacUser fetchUserByNameAndPassword(String name, String
password) {
        return pacUserRepository.findByNameAndPassword(name,
password);
    }
}

```

16. vaccinationCenterDao.java:
package com.example.demo.dao;

```

import java.util.List;
import java.util.Optional;

```

```
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Repository;

import com.example.demo.entity.pacUser;
import com.example.demo.entity.vaccinationCenter;
import com.example.demo.repository.vaccinationCenterRepository;
```

```
@Repository
public class vaccinationCenterDao {
    @Autowired
    private final vaccinationCenterRepository vaccinationCenterRepository;

    @Autowired
    public vaccinationCenterDao(vaccinationCenterRepository
vaccinationCenterRepository) {
        this.vaccinationCenterRepository = vaccinationCenterRepository;
    }

    public vaccinationCenter create(vaccinationCenter vaccinationCenter) {
        return vaccinationCenterRepository.save(vaccinationCenter);
    }

    public List<vaccinationCenter> read() {
        return vaccinationCenterRepository.findAll();
    }

    public Optional<vaccinationCenter> read(Long id) {
        return vaccinationCenterRepository.findById(id);
    }

    public vaccinationCenter update(vaccinationCenter vaccinationCenter) {
        return vaccinationCenterRepository.save(vaccinationCenter);
    }
}
```

```

        public void delete(Long id) {
            vaccinationCenterRepository.deleteById(id);
        }
    }
}

```

17. citizens.java:

```

package com.example.demo.entity;
import jakarta.persistence.Entity;
import jakarta.persistence.GeneratedValue;
import jakarta.persistence.GenerationType;
import jakarta.persistence.Id;
import jakarta.persistence.ManyToOne;
@Entity
public class citizens {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;
    private String name;
    private String city;
    private int noOfDoses;
    private String vaccinationStatus;
    @ManyToOne
    private vaccinationCenter vaccinationCenter;

    public citizens() {}
    public citizens(Long id, String name, String city, int noOfDoses, String vaccinationStatus,
        com.example.demo.entity.vaccinationCenter vaccinationCenter) {
        super();
        this.id = id;
        this.name = name;
        this.city = city;
        this.noOfDoses = noOfDoses;
        this.vaccinationStatus = vaccinationStatus;
        this.vaccinationCenter = vaccinationCenter;
    }
    public Long getId() {
        return id;
    }
    public void setId(Long id) {
        this.id = id;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
}

```



```

    }
    public String getCity() {
        return city;
    }
    public void setCity(String city) {
        this.city = city;
    }
    public int getNoOfDoses() {
        return noOfDoses;
    }
    public void setNoOfDoses(int noOfDoses) {
        this.noOfDoses = noOfDoses;
    }
    public String getVaccinationStatus() {
        return vaccinationStatus;
    }
    public void setVaccinationStatus(String vaccinationStatus) {
        this.vaccinationStatus = vaccinationStatus;
    }
    public vaccinationCenter getVaccinationCenter() {
        return vaccinationCenter;
    }
    public void setVaccinationCenter(vaccinationCenter vaccinationCenter) {
        this.vaccinationCenter = vaccinationCenter;
    }
    @Override
    public String toString() {
        return "citizens [id=" + id + ", name=" + name + ", city=" + city + ", noOfDoses=" +
noOfDoses
        + ", vaccinationStatus=" + vaccinationStatus + ", vaccinationCenter=" +
vaccinationCenter + "]";
    }
}

```

18. pacUser.java:

```

package com.example.demo.entity;
import jakarta.persistence.Entity;
import jakarta.persistence.GeneratedValue;
import jakarta.persistence.GenerationType;
import jakarta.persistence.Id;
@Entity
public class pacUser {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;

```

```

private String name;
private String email;
private String password;

public PacUser() {}
public PacUser(Long id, String name, String email, String password) {
    super();
    this.id = id;
    this.name = name;
    this.email = email;
    this.password = password;
}
public Long getId() {
    return id;
}
public void setId(Long id) {
    this.id = id;
}
public String getName() {
    return name;
}
public void setName(String name) {
    this.name = name;
}
public String getEmail() {
    return email;
}
public void setEmail(String email) {
    this.email = email;
}
public String getPassword() {
    return password;
}
public void setPassword(String password) {
    this.password = password;
}
@Override
public String toString() {
    return "PacUser [id=" + id + ", name=" + name + ", email=" + email + ", password=" +
password + "]\n";
}
}

```

19. vaccinationCenter.java:

```
package com.example.demo.entity;
```

```
import jakarta.persistence.Entity;
```

```
import jakarta.persistence.GeneratedValue;
import jakarta.persistence.GenerationType;
import jakarta.persistence.Id;
```

```
@Entity
```

```
public class vaccinationCenter {
```

```
    @Id
```

```
    @GeneratedValue(strategy = GenerationType.IDENTITY)
```

```
    private Long id;
```

```
    private String name;
```

```
    private String city;
```

```
    public vaccinationCenter() {}
```

```
    public vaccinationCenter(Long id, String name, String city) {
```

```
        super();
```

```
        this.id = id;
```

```
        this.name = name;
```

```
        this.city = city;
```

```
    }
```

```
    public Long getId() {
```

```
        return id;
```

```
    }
```

```
    public void setId(Long id) {
```

```
        this.id = id;
```

```
    }
```

```
    public String getName() {
```

```
        return name;
```

```
    }
```

```
    public void setName(String name) {
```

```
        this.name = name;
```

```
    }
```

```
    public String getCity() {
```

```
        return city;
```

```
    }
```

```

        public void setCity(String city) {
            this.city = city;
        }
        @Override
        public String toString() {
            return name;
        }
    }
}

```

20. citizensRepository.java:

```

package com.example.demo.repository;

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

import com.example.demo.entity.citizens;

public interface citizensRepository extends JpaRepository<citizens, Long>
{

}

```

21. pacUserRepository.java:

```

package com.example.demo.repository;

import com.example.demo.entity.pacUser;

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

public interface pacUserRepository extends JpaRepository<pacUser,
Long> {
    pacUser findByName(String name);
    pacUser findByNameAndPassword(String name, String password);
}

```

22. vaccinationCenterRepository.java:

```

package com.example.demo.repository;

```

```

import com.example.demo.entity.vaccinationCenter;
import org.springframework.data.jpa.repository.JpaRepository;

public interface vaccinationCenterRepository extends
JpaRepository<vaccinationCenter, Long> {

}

```

23. ServletInitializer.java:

```

package com.example;

import org.springframework.boot.builder.SpringApplicationBuilder;
import
org.springframework.boot.web.servlet.support.SpringBootServletInitializer;

public class ServletInitializer extends SpringBootServletInitializer {

    @Override
    protected SpringApplicationBuilder
configure(SpringApplicationBuilder application) {
        return
application.sources(SpringBootVaccinationCenterApplication.class);
    }

}

```

24. SpringBootVaccinationCenterApplication.java:

```

package com.example;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class SpringBootVaccinationCenterApplication {

    public static void main(String[] args) {

```

```

        SpringApplication.run(SpringBootVaccinationCenterApplication.class,
        args);
    }

}

```

25. applications.properties:

```

spring.mvc.view.prefix=/WEB-INF/jsp/
spring.mvc.view.suffix=.jsp
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
spring.datasource.url=jdbc:mysql://localhost:3306/vaccination
spring.datasource.username=root
spring.datasource.password=Srishti@1234
spring.jpa.show-sql=true
spring.jpa.hibernate.ddl-auto=update
spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL8Dialect

```

26. pom.xml:

```

<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>
    <parent>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-parent</artifactId>
        <version>3.1.0</version>
        <relativePath/> <!-- lookup parent from repository -->
    </parent>
    <groupId>com.example</groupId>
    <artifactId>spring-boot-vaccination-center</artifactId>
    <version>0.0.1-SNAPSHOT</version>
    <name>spring-boot-vaccination-center</name>
    <description>Demo project for Spring Boot</description>
    <properties>
        <java.version>17</java.version>
    </properties>
    <dependencies>
        <dependency>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-starter-data-jpa</artifactId>
        </dependency>
    </dependencies>

```

```

<!--
https://mvnrepository.com/artifact/jakarta.servlet.jsp.jstl/jakarta.servlet.jsp.jstl-api -->
<dependency>
    <groupId>jakarta.servlet.jsp.jstl</groupId>
    <artifactId>jakarta.servlet.jsp.jstl-api</artifactId>
    <version>3.0.0</version>
</dependency>
<!-- JSP support in Spring -->
<dependency>
    <groupId>org.apache.tomcat.embed</groupId>
    <artifactId>tomcat-embed-jasper</artifactId>
    <scope>provided</scope>
</dependency>
<!-- https://mvnrepository.com/artifact/commons-codec/commons-codec --> <!-- apache
commons may not be needed here added by fault -->
<dependency>
    <groupId>commons-codec</groupId>
    <artifactId>commons-codec</artifactId>
    <version>1.15</version>
</dependency>
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
</dependency>
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-devtools</artifactId>
    <scope>runtime</scope>
    <optional>true</optional>
</dependency>

<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-test</artifactId>
    <scope>test</scope>
</dependency>
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-tomcat</artifactId>
    <scope>provided</scope>
</dependency>
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter</artifactId>
</dependency>
<dependency>
    <groupId>com.h2database</groupId>
    <artifactId>h2</artifactId>
    <scope>runtime</scope>

```

```

        </dependency>
<!--
    <dependency>
        <groupId>com.mysql</groupId>
        <artifactId>mysql-connector-j</artifactId>
        <scope>runtime</scope>
    </dependency>-->
</dependencies>
<build>
    <plugins>
        <plugin>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-maven-plugin</artifactId>
        </plugin>
    </plugins>
</build>
</project>

```

SQL Query

```

create database vaccination;
use vaccination;

```

```

create table citizens(
    id bigint primary key,
    name varchar(255) not null,
    city varchar(255) not null,
    noOfDoses int not null,
    vaccinationStatus VARCHAR(255) NOT NULL,
    vaccination_center_id bigint );

```

```

CREATE TABLE USER(
    id bigint primary key,
    name varchar(255) not null,
    email VARCHAR(255) NOT NULL,
    password VARCHAR(255) NOT NULL );

```

```

CREATE TABLE vaccination_center(
    id bigint primary key,
    name varchar(255) not null,
    city VARCHAR(255) NOT NULL );
SELECT * FROM citizens;

```



```
SELECT * FROM USER;  
SELECT * FROM vaccination_center;
```