

CODE

VaccineCenterController

```
package com.ecommerce.controller;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.*;
import com.ecommerce.service.CitizenService;
import com.ecommerce.service.VaccineCenterService;
import java.util.List;

@Controller
@RequestMapping("/vaccine-centers")
public class VaccineCenterController {

    @Autowired
    private VaccineCenterService vaccineCenterService;

    @Autowired
    private CitizenService citizenService;
}
```

UserController

```
package com.ecommerce.controller;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.*;
import com.ecommerce.service.UserService;

@Controller
@RequestMapping("/")
public class UserController {
```

```
@Autowired
private UserService userService;
}
```

VaccineCenterController

```
package com.ecommerce.controller;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.*;
import com.ecommerce.service.CitizenService;
import com.ecommerce.service.VaccineCenterService;
import java.util.List;
```

```
@Controller
@RequestMapping("/vaccine-centers")
public class VaccineCenterController {
```

```
@Autowired
private VaccineCenterService vaccineCenterService;
```

```
@Autowired
private CitizenService citizenService;
}
```

Citizen

```
package com.ecommerce.entity;

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

```
import jakarta.persistence.GeneratedValue;
import jakarta.persistence.GenerationType;
import jakarta.persistence.Id;
import jakarta.persistence.JoinColumn;
import jakarta.persistence.ManyToOne;
import jakarta.persistence.Table;
import lombok.AllArgsConstructor;
import lombok.NoArgsConstructor;

@Entity
@Table(name = "citizens")
public class Citizen {

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

    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 VaccineCenter getVaccinationCenter() {  
    return vaccinationCenter;  
}  
public void setVaccinationCenter(VaccineCenter vaccinationCenter) {  
    this.vaccinationCenter = vaccinationCenter;  
}  
@ManyToOne(fetch = FetchType.LAZY)  
@JoinColumn(name = "vaccination_center_id")  
private VaccineCenter vaccinationCenter;
```

```

        @Override
        public String toString() {
            return "Citizen [id=" + id + ", name=" + name + ", city=" + city + ", noOfDoses=" +
noOfDoses
                                + ", vaccinationStatus=" + vaccinationStatus + ",
vaccinationCenter=" + vaccinationCenter + "]";
        }
    }
}

```

User

```

package com.ecommerce.entity;

import jakarta.persistence.Column;
import jakarta.persistence.Entity;
import jakarta.persistence.GeneratedValue;
import jakarta.persistence.GenerationType;
import jakarta.persistence.Id;
import jakarta.persistence.Table;
import lombok.AllArgsConstructor;
import lombok.NoArgsConstructor;

@Entity
@Table(name = "users")
public class User {

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

    @Column(nullable = false, unique = true)
    private String username;
}

```

```
@Column(nullable = false, unique = true)
private String email;
@Column(nullable = false)
private String password;

    public Long getId() {
        return id;
    }

    public void setId(Long id) {
        this.id = id;
    }

    public String getUsername() {
        return username;
    }

    public void setUsername(String username) {
        this.username = username;
    }

    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 "User [id=" + id + ", username=" + username + ", email=" + email + ",
password=" + password + "]";
        }
    }
}
```

VaccineCenter

```
package com.ecommerce.entity;

import jakarta.persistence.Entity;
import jakarta.persistence.GeneratedValue;
import jakarta.persistence.GenerationType;
import jakarta.persistence.Id;
import jakarta.persistence.Table;
import lombok.AllArgsConstructor;
import lombok.NoArgsConstructor;

@Entity
@Table(name = "vaccine_centers")
public class VaccineCenter {

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

    public Long getId() {
        return id;
    }

    public void setId(Long id) {
        this.id = id;
    }
}
```

```

    public String getCenter() {
        return Center;
    }

    public void setCenter(String center) {
        Center = center;
    }

    public String getCity() {
        return City;
    }

    public void setCity(String city) {
        City = city;
    }

    @Override
    public String toString() {
        return "VaccineCenter [id=" + id + ", center=" + Center + ", city=" + City + "];"
    }
}

```

CitizenRepository

```

package com.ecommerce.repository;

import java.util.List;

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

import com.ecommerce.entity.Citizen;
import com.ecommerce.entity.VaccineCenter;

public interface CitizenRepository extends JpaRepository<Citizen, Long> {

    List<Citizen> findByVaccinationCenter(VaccineCenter vaccineCenter);

}

```


UserRepository

```
package com.ecommerce.repository;

import org.springframework.data.jpa.repository.JpaRepository;
import com.ecommerce.entity.User;

public interface UserRepository extends JpaRepository<User, Long> {

    User findByUsername(String username);
}
```

VaccineCenterRepository

```
package com.ecommerce.repository;

import org.springframework.data.jpa.repository.JpaRepository;
import com.ecommerce.entity.VaccineCenter;

public interface VaccineCenterRepository extends JpaRepository<VaccineCenter, Long> {

}
```

CitizenService

```
package com.ecommerce.service;

import java.util.List;
import com.ecommerce.entity.Citizen;
import com.ecommerce.entity.VaccineCenter;

public interface CitizenService {

    void addCitizen(Citizen citizen);

    void updateCitizen(Citizen citizen);

    void deleteCitizen(Long id);

    Citizen getCitizenById(Long id);

    List<Citizen> getAllCitizens();

    List<Citizen> getCitizensByVaccinationCenter(VaccineCenter vaccineCenter);
}
```

CitizenServiceImpl

```
package com.ecommerce.service;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;
import com.ecommerce.entity.Citizen;
import com.ecommerce.entity.VaccineCenter;
import com.ecommerce.repository.CitizenRepository;
import java.util.List;

@Service
public class CitizenServiceImpl implements CitizenService {

    // Inject CitizenRepository
    @Autowired
    private CitizenRepository citizenRepository;

    @Override
    public void addCitizen(Citizen citizen) {
        citizenRepository.save(citizen);
    }

    @Override
    public void updateCitizen(Citizen citizen) {
        citizenRepository.save(citizen);
    }

    @Override
    public void deleteCitizen(Long id) {
        citizenRepository.deleteById(id);
    }

    @Override
    public Citizen getCitizenById(Long id) {
        return citizenRepository.findById(id)
            .orElseThrow(() -> new NotFoundException("Citizen not found"));
    }
}
```

```

    }

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

    @Override
    public List<Citizen> getCitizensByVaccinationCenter(VaccineCenter vaccineCenter) {
        return citizenRepository.findByVaccinationCenter(vaccineCenter);
    }

    // Custom exception class
    public class NotFoundException extends RuntimeException {
        public NotFoundException(String message) {
            super(message);
        }
    }
}

```

UserService

```

package com.ecommerce.service;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;
import com.ecommerce.entity.User;
import com.ecommerce.repository.UserRepository;

@Service
public class UserService {

    // Inject UserRepository
    @Autowired
    private UserRepository userRepository;

    public void registerUser(User user) {
        userRepository.save(user);
    }
}

```

```
}  
}
```

VaccineCenterService

```
package com.ecommerce.service;  
  
import java.util.List;  
  
import com.ecommerce.entity.VaccineCenter;  
  
public interface VaccineCenterService {  
  
    VaccineCenter addVaccineCenter(VaccineCenter vaccineCenter);  
  
    void updateVaccineCenter(VaccineCenter updatedVaccineCenter);  
  
    void deleteVaccineCenter(Long id);  
  
    VaccineCenter getVaccineCenterById(Long id);  
  
    List<VaccineCenter> getAllVaccineCenters();  
  
}
```

VaccineCenterServiceImpl

```
package com.ecommerce.service;  
  
import org.springframework.beans.factory.annotation.Autowired;  
  
import org.springframework.stereotype.Service;  
  
import com.ecommerce.entity.VaccineCenter;  
  
import com.ecommerce.repository.VaccineCenterRepository;  
  
import java.util.List;  
  
@Service  
  
public class VaccineCenterServiceImpl implements VaccineCenterService {  
  
    // Inject VaccineCenterRepository  
  
    @Autowired  
  
    private VaccineCenterRepository vaccineCenterRepository;  
  
    @Override  
  
    public VaccineCenter addVaccineCenter(VaccineCenter vaccineCenter) {
```

```

        return vaccineCenterRepository.save(vaccineCenter);
    }

    @Override
    public void updateVaccineCenter(VaccineCenter updatedVaccineCenter) {
        VaccineCenter existingCenter =
            vaccineCenterRepository.findById(updatedVaccineCenter.getId()).orElseThrow(() -> new
            NotFoundException("Vaccine center not found"));

        existingCenter.setCenter(updatedVaccineCenter.getCenter());
        existingCenter.setCity(updatedVaccineCenter.getCity());
        vaccineCenterRepository.save(existingCenter);
    }

    @Override
    public void deleteVaccineCenter(Long id) {
        VaccineCenter existingCenter = vaccineCenterRepository.findById(id)
            .orElseThrow(() -> new NotFoundException("Vaccine center not found"));

        vaccineCenterRepository.delete(existingCenter);
    }

    @Override
    public VaccineCenter getVaccineCenterById(Long id) {
        return vaccineCenterRepository.findById(id)
            .orElseThrow(() -> new NotFoundException("Vaccine center not found"));
    }

    @Override
    public List<VaccineCenter> getAllVaccineCenters() {
        return vaccineCenterRepository.findAll();
    }

    // Custom exception class
    public class NotFoundException extends RuntimeException {
        public NotFoundException(String message) {

```

```
        super(message);
    }
}
}
```

SpringVaccineCertificateApplication

```
package com.example.demo;

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

@SpringBootApplication
public class SpringVaccineCertificateApplication {

    public static void main(String[] args) {

        SpringApplication.run(SpringVaccineCertificateApplication.class, args);

    }

}
```

ServletInitializer

```
package com.example.demo;

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(SpringVaccineCertificateApplication.class);

    }

}
```

Application

server.port=8082

#jpa-hibernate

spring.jpa.hibernate.ddl-auto=create

spring.jpa.show-sql=true

spring.jpa.hibernate.dialect=org.hibernate.dialect.MySQLDialect

#datasource

spring.datasource.driver-class-name=com.mysql.jdbc.Driver

spring.datasource.url=jdbc:mysql://localhost:3306/vaccination

spring.datasource.username=root

spring.datasource.password=nareshrao

Add.Citizen

```
<!DOCTYPE html>
```

```
<html xmlns:th="http://www.thymeleaf.org">
```

```
  <head>
```

```
    <title>Add Citizen</title>
```

```
    <link rel="stylesheet"
```

```
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" />
```

```
  </head>
```

```
  <body>
```

```
    <div class="container">
```

```
      <h1 class="mt-4">Add Citizen</h1>
```

```
      <form action="/citizens/add" method="POST">
```

```
        <div class="form-group">
```

```
          <label for="name">Name:</label>
```

```
        <input type="text" class="form-control" id="name" name="name" required />
    </div>
    <div class="form-group">
        <label for="city">City:</label>
        <input type="text" class="form-control" id="city" name="city" required />
    </div>
    <div class="form-group">
        <label for="noOfDoses">No. of Doses:</label>
        <select class="form-control" id="noOfDoses" name="noOfDoses" required>
            <option value="0">0</option>
            <option value="1">1</option>
            <option value="2">2</option>
        </select>
    </div>
    <div class="form-group">
        <label for="vaccinationStatus">Vaccination Status:</label>
        <input type="text" class="form-control" id="vaccinationStatus"
name="vaccinationStatus" required readonly />
    </div>
    <script>
        const noOfDosesSelect = document.getElementById("noOfDoses");
        const vaccinationStatusInput = document.getElementById("vaccinationStatus");
        noOfDosesSelect.addEventListener("change", function () {
            const selectedDoses = parseInt(noOfDosesSelect.value);
            if (selectedDoses === 0) {
                vaccinationStatusInput.value = "Not Vaccinated";
            } else if (selectedDoses === 1) {
                vaccinationStatusInput.value = "One Vaccinated";
            } else if (selectedDoses === 2) {
```



```

        vaccinationStatusInput.value = "Fully Vaccinated";
    }
});
</script>
<div class="form-group">
    <label for="vaccineCenterId">Vaccine Center:</label>
    <select class="form-control" id="vaccineCenterId" name="vaccineCenterId"
required>
        <!-- Populate the options with available Vaccine Centers -->
        <option th:each="vaccineCenter : ${vaccineCenters}"
th:value="${vaccineCenter.id}" th:text="${vaccineCenter.center}"></option>
    </select>
</div>
<button type="submit" class="btn btn-primary">Add Citizen</button>
<a href="/citizens" class="btn btn-secondary">Back</a>
</form>
</div>
<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
<script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
</body>
</html>

```

Add-Vaccine

```

<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
    <head>
        <title>Add Citizen</title>
        <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" />

```

```
</head>
<body>
  <div class="container">
    <h1 class="mt-4">Add Citizen</h1>
    <form action="/citizens/add" method="POST">
      <div class="form-group">
        <label for="name">Name:</label>
        <input type="text" class="form-control" id="name" name="name" required />
      </div>
      <div class="form-group">
        <label for="city">City:</label>
        <input type="text" class="form-control" id="city" name="city" required />
      </div>
      <div class="form-group">
        <label for="noOfDoses">No. of Doses:</label>
        <select class="form-control" id="noOfDoses" name="noOfDoses" required>
          <option value="0">0</option>
          <option value="1">1</option>
          <option value="2">2</option>
        </select>
      </div>
      <div class="form-group">
        <label for="vaccinationStatus">Vaccination Status:</label>
        <input type="text" class="form-control" id="vaccinationStatus"
name="vaccinationStatus" required readonly />
      </div>
      <script>
        const noOfDosesSelect = document.getElementById("noOfDoses");
        const vaccinationStatusInput = document.getElementById("vaccinationStatus");
```

```

noOfDosesSelect.addEventListener("change", function () {
    const selectedDoses = parseInt(noOfDosesSelect.value);
    if (selectedDoses === 0) {
        vaccinationStatusInput.value = "Not Vaccinated";
    } else if (selectedDoses === 1) {
        vaccinationStatusInput.value = "One Vaccinated";
    } else if (selectedDoses === 2) {
        vaccinationStatusInput.value = "Fully Vaccinated";
    }
});
</script>
<div class="form-group">
    <label for="vaccineCenterId">Vaccine Center:</label>
    <select class="form-control" id="vaccineCenterId" name="vaccineCenterId"
required>
        <!-- Populate the options with available Vaccine Centers -->
        <option th:each="vaccineCenter : ${vaccineCenters}"
th:value="${vaccineCenter.id}" th:text="${vaccineCenter.center}"></option>
    </select>
</div>
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
    <head>
        <title>Add Vaccination Center</title>
        <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" />
    </head>
    <body>
        <div class="container">
            <h1 class="mt-4">Add Vaccination Center</h1>

```

```

        <form th:action="@{/vaccine-centers/add}" method="POST">
            <div class="form-group">
                <label for="center">Center:</label>
                <input type="text" class="form-control" id="center" name="center"
required />
            </div>
            <div class="form-group">
                <label for="city">City:</label>
                <input type="text" class="form-control" id="city" name="city" required />
            </div>
            <button type="submit" class="btn btn-primary">Submit</button>
        </form>
    </div>
    <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
    <script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
    </body>
</html>

    <button type="submit" class="btn btn-primary">Add Citizen</button>
    <a href="/citizens" class="btn btn-secondary">Back</a>
</form>
</div>
<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
<script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
</body>
</html>

```

Citizen-Details

```
<!DOCTYPE html>

<html xmlns:th="http://www.thymeleaf.org">

  <head>

    <title>Citizen Details</title>

    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min. css" />

  </head>

  <body>

    <div class="container">

      <h1 class="mt-4">Citizen Details</h1>

      <a href="/citizens/add" class="btn btn-success">Add Citizen</a>

      <a href="/vaccine-centers" class="btn btn-secondary">Back to Home</a><br />

      <br />

      <table class="table">

        <thead>

          <tr>

            <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>

          </tr>

        </thead>

        <tbody>

          <tr th:each="citizen : ${citizens}">

            <td th:text="${citizen.id}"></td>
```

```
<td th:text="{citizen.name}"></td>
<td th:text="{citizen.city}"></td>
<td th:text="{citizen.noOfDoses}"></td>
<td th:text="{citizen.vaccinationStatus}"></td>
<td th:text="{citizen.vaccinationCenter?.center}"></td>
<td>
    <a th:href="@{/citizens/{id}(id={citizen.id})}" class="btn
btn-primary">View</a>
    <a th:href="@{/citizens/{id}/edit(id={citizen.id})}" class="btn
btn-secondary">Edit</a>
    <a th:href="@{/citizens/{id}/delete(id={citizen.id})}" class="btn
btn-danger">Delete</a>
</td>
</tr>
</tbody>
</table>
<h6 style="text-align: left;">
    <strong>Total </strong>
    <strong th:text="{#lists.size(citizens)}"></strong>
    <strong> Citizens found</strong>
</h6>
</div>
<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
<script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
</body>
</html>
```

Edit-Citizen

```
<!DOCTYPE html>

<html xmlns:th="http://www.thymeleaf.org">

  <head>

    <title>Edit Citizen</title>

    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" />

  </head>

  <body>

    <div class="container">

      <h1 class="mt-4">Edit Citizen</h1>

      <form th:action="@{/citizens/{id}/edit(id=${citizen.id})}" method="POST">

        <div class="form-group">

          <label for="name">Name:</label>

          <input type="text" class="form-control" id="name" name="name"
th:value="${citizen.name}" required />

        </div>

        <div class="form-group">

          <label for="city">City:</label>

          <input type="text" class="form-control" id="city" name="city"
th:value="${citizen.city}" required />

        </div>

        <div class="form-group">

          <label for="noOfDoses">No. of Doses:</label>

          <select class="form-control" id="noOfDoses" name="noOfDoses" required>

            <option value="0" th:selected="${citizen.noOfDoses == 0}">0</option>

            <option value="1" th:selected="${citizen.noOfDoses == 1}">1</option>

            <option value="2" th:selected="${citizen.noOfDoses == 2}">2</option>

          </select>

        </div>

      </form>

    </div>

  </body>

</html>
```

```

<div class="form-group">
    <label for="vaccinationStatus">Vaccination Status:</label>
    <input type="text" class="form-control" id="vaccinationStatus"
name="vaccinationStatus" th:value="${citizen.vaccinationStatus}" required readonly />
</div>
<div class="form-group">
    <label for="vaccineCenterId">Vaccine Center:</label>
    <select class="form-control" id="vaccineCenterId" name="vaccineCenterId"
required>
        <!-- Populate the options with available Vaccine Centers -->
        <option th:each="vaccineCenter : ${vaccineCenters}"
th:value="${vaccineCenter.id}" th:text="${vaccineCenter.center}" th:selected="${vaccineCenter.id
== citizen.vaccinationCenter.id}"></option>
    </select>
</div>
<button type="submit" class="btn btn-primary">Update Citizen</button>
<a href="/citizens" class="btn btn-secondary">Back</a>
</form>
</div>
<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js
"></script>
<script>
    const noOfDosesSelect = document.getElementById("noOfDoses");
    const vaccinationStatusInput = document.getElementById("vaccinationStatus");
    noOfDosesSelect.addEventListener("change", function () {
        const selectedDoses = parseInt(noOfDosesSelect.value);
        if (selectedDoses === 0) {
            vaccinationStatusInput.value = "Not Vaccinated";
        } else if (selectedDoses === 1) {
            vaccinationStatusInput.value = "One Vaccinated";

```



```

        } else if (selectedDoses === 2) {
            vaccinationStatusInput.value = "Fully Vaccinated";
        }
    });
</script>
</body>
</html>

```

Edit-Vaccine-Center

```

<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
    <head>
        <title>Edit Vaccine Center</title>
        <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" />
    </head>
    <body>
        <div class="container">
            <h1 class="mt-4">Edit Vaccine Center</h1>
            <form th:action="@{/vaccine-centers/{id}/edit(id=${vaccineCenter.id})}" method="POST">
                <div class="form-group">
                    <label for="center">Center:</label>
                    <input type="text" class="form-control" id="center"
th:field="${vaccineCenter.center}" required />
                </div>
                <div class="form-group">
                    <label for="city">City:</label>
                    <input type="text" class="form-control" id="city" th:field="${vaccineCenter.city}"
required />
                </div>
            </form>
        </div>
    </body>
</html>

```

```
        <!-- Add other fields as needed -->

        <button type="submit" class="btn btn-primary">Update Vaccine Center</button>

    </form>

</div>

<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>

<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"
"></script>

</body>

</html>
```

ErrorPage

```
<!DOCTYPE html>

<html>

    <head>

        <title>Error - Vaccination Project</title>

        <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" />

    </head>

    <body>

        <div class="container">

            <h1 class="mt-4">Error</h1>

            <div class="alert alert-danger" role="alert">

                Invalid username or password. Please try again or register.

            </div>

            <a href="/register" class="btn btn-primary">Register</a>

            <a href="/" class="btn btn-secondary">Try Again</a>

        </div>

        <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>

        <script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
```

```
</body>
</html>
```

Login

```
<!DOCTYPE html>
<html>
  <head>
    <title>Login - Vaccination Project</title>
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" />
  </head>
  <body>
    <div class="container">
      <h1 class="mt-4">Login</h1>
      <form action="/login" method="POST">
        <div class="form-group">
          <label for="username">Username:</label>
          <input type="text" class="form-control" id="username" name="username" required
/>
        </div>
        <div class="form-group">
          <label for="password">Password:</label>
          <input type="password" class="form-control" id="password" name="password"
required />
        </div>
        <button type="submit" class="btn btn-primary">Login</button>
      </form>
      <p class="mt-3">Don't have an account? <a href="/register">Register here</a>.</p>
    </div>
    <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
```

```
    <script
src="https://cdn.jsdelivr.net/npm/@popperjs/core@1.16.1/dist/umd/popper.min.js"></script>

    <script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>

    </body>
</html>
```

Register

```
<!DOCTYPE html>

<html xmlns:th="http://www.thymeleaf.org">

    <head>

        <title>Register - Vaccination Project</title>

        <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" />

    </head>

    <body>

        <div class="container">

            <h1 class="mt-4">Register</h1>

            <form action="@{/register}" method="POST">

                <div class="form-group">

                    <label for="username">Username:</label>

                    <input type="text" class="form-control" id="username" name="username" required

/>

                </div>

                <div class="form-group">

                    <label for="email">Email:</label>

                    <input type="email" class="form-control" id="email" name="email" required />

                </div>

                <div class="form-group">

                    <label for="password">Password:</label>
```

```

        <input type="password" class="form-control" id="password" name="password"
required />
    </div>
    <button type="submit" class="btn btn-primary">Register</button>
</form>
<p class="mt-3">Already have an account? <a href="/">Login here</a>.</p>
</div>
<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
<script
src="https://cdn.jsdelivr.net/npm/@popperjs/core@1.16.1/dist/umd/popper.min.js"></script>
<script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
</body>
</html>

```

Vaccine-Center-Details

```

<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
    <head>
        <title>Vaccination Center Details</title>
        <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min. css" />
        <style>
            /* Styles here */
        </style>
    </head>
    <body>
        <div class="container">
            <h1 class="mt-4">Welcome to Vaccination Center</h1>
            <ul>
                <li><a href="/citizens" style="color: #007bff;">Go to Citizen Details</a></li>

```

```

        <li><a href="/" style="color: #dc3545;">Logout</a></li>
    </ul>

    <hr />

    <h2 class="mt-4">Vaccination Center Details</h2>

    <a th:href="@{/vaccine-centers/add}" class="btn btn-primary mb-4">Add Vaccination
Center</a>

    <table class="table">
        <thead>
            <tr>
                <th>ID</th>
                <th>Center</th>
                <th>City</th>
                <th>Actions</th>
            </tr>
        </thead>
        <tbody id="vaccineCenterTableBody">
            <!-- Loop through the vaccination centers and display their details -->
            <tr th:each="vaccineCenter : ${vaccineCenters}">
                <td th:text="${vaccineCenter.id}"></td>
                <td th:text="${vaccineCenter.center}"></td>
                <td th:text="${vaccineCenter.city}"></td>
                <td>
                    <a th:href="@{/vaccine-centers/{id}(id=${vaccineCenter.id})}" class="btn btn
primary">View</a>
                    <a th:href="@{/vaccine-centers/{id}/edit(id=${vaccineCenter.id})}" class="btn
btn secondary">Edit</a>
                    <a th:href="@{/vaccine-centers/{id}/delete(id=${vaccineCenter.id})}"
class="btn btn danger">Delete</a>
                </td>
            </tr>
        </tbody>
    </table>

```

```

        </tbody>
    </table>

    <h6 style="text-align: left;"><strong>Total </strong><strong
th:text="{#lists.size(vaccineCenters)} "></strong><strong> Vaccination Center
found</strong></h6>

</div>

<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>

<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js
"></script>

</body>
</html>

```

Vaccine-Center-View

```

<!DOCTYPE html>

<html xmlns:th="http://www.thymeleaf.org">

    <head>

        <title>Vaccination Center Details</title>

        <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min. css" />

    </head>

    <body>

        <div class="container">

            <h1 class="mt-4">Vaccination Center Details</h1>

            <hr />

            <h2 class="mt-4">Vaccination Center</h2>

            <table class="table">

                <thead>

                    <tr>

                        <th>ID</th>

                        <th>Center</th>


```

```

        <th>City</th>
    </tr>
</thead>
<tbody>
    <tr>
        <td th:text="{vaccineCenter.id}"></td>
        <td th:text="{vaccineCenter.center}"></td>
        <td th:text="{vaccineCenter.city}"></td>
    </tr>
</tbody>
</table>

```

<h2 class="mt-4">Associated Citizens</h2>

```

<table class="table">
    <thead>
        <tr>
            <th>ID</th>
            <th>Name</th>
            <th>City</th>
            <th>No. of Doses</th>
            <th>Vaccination Status</th>
        </tr>
    </thead>
    <tbody>
        <tr th:each="citizen : {citizens}">
            <td th:text="{citizen.id}"></td>
            <td th:text="{citizen.name}"></td>
            <td th:text="{citizen.city}"></td>
            <td th:text="{citizen.noOfDoses}"></td>
            <td th:text="{citizen.vaccinationStatus}"></td>
        </tr>
    </tbody>
</table>

```



```

        </tr>
    </tbody>
</table>

<a href="/vaccine-centers" class="btn btn-secondary">Back to Vaccination Centers</a>
</div>

<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"
"></script>
</body>
</html>

```

View-Citizen

```

<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
    <head>
        <title>View Citizen</title>
        <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min. css" />
    </head>
    <body>
        <div class="container">
            <h1 class="mt-4">Citizen Details</h1>
            <div class="card">
                <div class="card-body">
                    <h5 class="card-title">Name: <span th:text="${citizen.name}"></span></h5>
                    <h6 class="card-subtitle mb-2 text-muted">City: <span
th:text="${citizen.city}"></span></h6>
                    <p class="card-text">No. of Doses: <span
th:text="${citizen.noOfDoses}"></span></p>

```

```

        <p class="card-text">Vaccination Status: <span
th:text="${citizen.vaccinationStatus}"></span></p>

        <p class="card-text">Vaccine Center: <span
th:text="${citizen.vaccinationCenter.center}"></span></p>

    </div>

</div>

<a href="/citizens" class="btn btn-primary mt-4">Back</a>

</div>

<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>

<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js
"></script>

</body>

</html>

```

POM

CitizenService

```

<?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>3.1.6</version>

        <relativePath/> <!-- lookup parent from repository -->

    </parent>

    <groupId>com.example</groupId>

    <artifactId>SPRING_VACCINE_CERTIFICATE</artifactId>

```

```
<version>0.0.1-SNAPSHOT</version>
<packaging>war</packaging>
<name>SPRING_VACCINE_CERTIFICATE</name>
<description>SPRING_VACCINE_CERTIFICATE</description>
<properties>
    <java.version>17</java.version>
</properties>
<dependencies>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-data-jpa</artifactId>
    </dependency>

    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-thymeleaf</artifactId>
    </dependency>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-web</artifactId>
    </dependency>
    <dependency>
        <groupId>org.thymeleaf.extras</groupId>
        <artifactId>thymeleaf-extras-springsecurity6</artifactId>
    </dependency>

    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-devtools</artifactId>
```

```
        <scope>runtime</scope>
        <optional>true</optional>
    </dependency>
    <dependency>
        <groupId>com.mysql</groupId>
        <artifactId>mysql-connector-j</artifactId>
        <scope>runtime</scope>
    </dependency>
    <dependency>
        <groupId>org.projectlombok</groupId>
        <artifactId>lombok</artifactId>
        <optional>true</optional>
    </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-test</artifactId>
        <scope>test</scope>
    </dependency>

</dependencies>

<build>
    <plugins>
        <plugin>
```

```

        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-maven-plugin</artifactId>
        <configuration>
            <image>
                <builder>paketobuildpacks/builder-jammy-
base:latest</builder>
            </image>
            <excludes>
                <exclude>
                    <groupId>org.projectlombok</groupId>
                    <artifactId>lombok</artifactId>
                </exclude>
            </excludes>
        </configuration>
    </plugin>
</plugins>
</build>

</project>
```