**MINI PROJECT**

**VACCINATION CENTER**

File Structure:

VaccinationCenter

**├── src/**

**│   ├── main/**

**│   │   ├── java/**

**│   │   │   └── com/**

**│   │   │       └── example/**

**│   │   │           └── vaccination/**

**│   │   │               ├── VaccinationProjectApplication.java**

**│   │   │               ├── config/**

**│   │   │               │   └── SwaggerConfig.java**

**│   │   │               ├── controller/**

**│   │   │               │   ├── CitizenController.java**

**│   │   │               │   └── VaccinationCenterController.java**

**│   │   │               ├── entity/**

**│   │   │               │   ├── CitizenBean.java**

**│   │   │               │   └── VaccineCenterBean.java**

**│   │   │               ├── repository/**

**│   │   │               │   ├── CitizenRepositoryDao.java**

**│   │   │               │   └── VaccineCenterRepositoryDao.java**

**│   │   │               ├── service/**

**│   │   │               │   ├── CitizenService.java**

**│   │   │               │   ├── CitizenServiceImpl.java**

**│   │   │               │   ├── VaccinationCenterService.java**

**│   │   │               │   └── VaccinationCenterServiceImpl.java**

**│   ├── resources/**

**│   │   ├── application.properties**

**│   │   └── static/**

**│   │       └── CSS  
| | |-- styles.css**

**│   │   └── templates/**

**│   │       └── index.html  
| | |-- citizens.html  
| | |--centers.html**

**├── pom.xml**

**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>3.5.3</version>

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

</parent>

<groupId>com.example</groupId>

<artifactId>VaccinationCenter</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>VaccinationCenter</name>

<description>Demo project for Spring Boot</description>

<url/>

<licenses>

<license/>

</licenses>

<developers>

<developer/>

</developers>

<scm>

<connection/>

<developerConnection/>

<tag/>

<url/>

</scm>

<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-web</artifactId>

</dependency>

<dependency>

<groupId>com.mysql</groupId>

<artifactId>mysql-connector-j</artifactId>

<scope>runtime</scope>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-test</artifactId>

<scope>test</scope>

</dependency>

<dependency>

   <groupId>org.springdoc</groupId>

   <artifactId>springdoc-openapi-starter-webmvc-ui</artifactId>

   <version>2.0.2</version>

</dependency>

    <dependency>

    <groupId>io.springfox</groupId>

    <artifactId>springfox-swagger-ui</artifactId>

    <version>2.9.2</version>

</dependency>

<!--<dependency>

    <groupId>javax.validation</groupId>

    <artifactId>validation-api</artifactId>

    <version>2.0.1.Final</version>

</dependency>-->

<dependency>

    <groupId>org.springframework.boot</groupId>

    <artifactId>spring-boot-starter-validation</artifactId>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-maven-plugin</artifactId>

</plugin>

</plugins>

</build>

</project>

**Application properties:**

spring.application.name=VaccinationCenter

server.port=8754

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

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

spring.datasource.username=root

spring.datasource.password=Password@12

spring.jpa.hibernate.ddl-auto=update

spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL8Dialect

spring.jpa.show-sql=true

#spring.devtools.livereload.enabled=true

**com.example.vaccinationCenter**

**VaccinationCenterApplication.java**package com.example.VaccinationCenter;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class VaccinationCenterApplication {

public static void main(String[] args) {

SpringApplication.run(VaccinationCenterApplication.class, args);

}

}

**com.example.VaccinationCenter.bean**

**CitizenBean.java**

**package** com.example.VaccinationCenter.bean;

**import** jakarta.persistence.\*;

@Entity

@Table(name="citizen")

**public** **class** CitizenBean {

@Id

@GeneratedValue

**private** **int** citizenid;

@Column

**private** String citizenname;

@Column

**private** **int** citizenage;

@Column

**private** String vaccinationstatus;

@ManyToOne

@JoinColumn(name="vaccinationid")

**private** VaccinationCenterBean vaccination;

**public** VaccinationCenterBean getVaccination() {

**return** vaccination;

}

**public** **void** setVaccination(VaccinationCenterBean vaccination) {

**this**.vaccination = vaccination;

}

**public** **int** getCitizenid() {

**return** citizenid;

}

**public** **void** setCitizenid(**int** citizenid) {

**this**.citizenid = citizenid;

}

**public** String getCitizenname() {

**return** citizenname;

}

**public** **void** setCitizenname(String citizenname) {

**this**.citizenname = citizenname;

}

**public** **int** getCitizenage() {

**return** citizenage;

}

**public** **void** setCitizenage(**int** citizenage) {

**this**.citizenage = citizenage;

}

**public** String getVaccinationstatus() {

**return** vaccinationstatus;

}

**public** **void** setVaccinationstatus(String vaccinationstatus) {

**this**.vaccinationstatus = vaccinationstatus;

}

@Override

public String toString() {

return "Citizen [id=" + id + ", name=" + name + ", email=" + email + ", phone=" + phone + ", address=" + address

+ "]";

}

}

**com.example.VaccinationCenter.bean**

**VaccinationCenterBean.java**

package com.example.VaccinationCenter.bean;

import java.util.List;

import jakarta.persistence.\*;

@Entity

@Table(name="vaccination\_center")

public class VaccinationCenterBean {

@Id

@GeneratedValue

private int vaccinationid;

@Column

private String vaccinationname;

@Column

private String location;

@Column

private int capacity;

@OneToMany(mappedBy = "vaccination")

List<CitizenBean> citizens;

public List<CitizenBean> getCitizens() {

return citizens;

}

public void setCitizens(List<CitizenBean> citizens) {

this.citizens = citizens;

}

public int getVaccinationid() {

return vaccinationid;

}

public void setVaccinationid(int vaccinationid) {

this.vaccinationid = vaccinationid;

}

public String getVaccinationname() {

return vaccinationname;

}

public void setVaccinationname(String vaccinationname) {

this.vaccinationname = vaccinationname;

}

public String getLocation() {

return location;

}

public void setLocation(String location) {

this.location = location;

}

public int getCapacity() {

return capacity;

}

public void setCapacity(int capacity) {

this.capacity = capacity;

}

}

**com.example.VaccinationCenter.controller**

**CitizenController.java**

package com.example.VaccinationCenter.controller;

import com.example.entity.Citizen;

import com.example.service.CitizenService;

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

import org.springframework.http.ResponseEntity;

import org.springframework.web.bind.annotation.\*;

import java.util.List;

@RestController

@RequestMapping("/api/citizens")

public class CitizenController {

    @Autowired

    private CitizenService citizenService;

    @GetMapping

    public List<Citizen> getAllCitizens() {

        return citizenService.getAllCitizens();

    }

    @GetMapping("/{id}")

    public ResponseEntity<Citizen> getCitizenById(@PathVariable Long id) {

        Citizen citizen = citizenService.getCitizenById(id);

        if (citizen != null) {

            return ResponseEntity.*ok*(citizen);

        }

        return ResponseEntity.*notFound*().build();

    }

    @PostMapping("/addCitizen")

    public Citizen createCitizen(@RequestBody Citizen citizen) {

System.*out*.println(citizen);

        return citizenService.createCitizen(citizen);

    }

    @PutMapping("/{id}")

    public ResponseEntity<Citizen> updateCitizen(@PathVariable Long id, @RequestBody Citizen citizen) {

        Citizen updatedCitizen = citizenService.updateCitizen(id, citizen);

        if (updatedCitizen != null) {

            return ResponseEntity.*ok*(updatedCitizen);

        }

        return ResponseEntity.*notFound*().build();

    }

    @DeleteMapping("/{id}")

    public ResponseEntity<Void> deleteCitizen(@PathVariable Long id) {

        citizenService.deleteCitizen(id);

        return ResponseEntity.*noContent*().build();

    }

}

**com.example.VaccinationCenter.controller**

**VaccinationCenterController.java**

package com.example.VaccinationCenter.controller;

import com.example.entity.VaccineCenter;

import com.example.service.VaccinationCenterService;

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

import org.springframework.http.ResponseEntity;

import org.springframework.web.bind.annotation.\*;

import java.util.List;

@RestController

@RequestMapping("/api/centers")

public class VaccinationCenterController {

    @Autowired

    private VaccinationCenterService centerService;

    @GetMapping

    public List<VaccineCenter> getAllCenters() {

        return centerService.getAllCenters();

    }

    @GetMapping("/{id}")

    public ResponseEntity<VaccineCenter> getCenterById(@PathVariable Long id) {

        VaccineCenter center = centerService.getCenterById(id);

        if (center != null) {

            return ResponseEntity.*ok*(center);

        }

        return ResponseEntity.*notFound*().build();

    }

    @PostMapping

    public VaccineCenter createCenter(@RequestBody VaccineCenter center) {

        return centerService.createCenter(center);

    }

    @PutMapping("/{id}")

    public ResponseEntity<VaccineCenter> updateCenter(@PathVariable Long id, @RequestBody VaccineCenter center) {

        VaccineCenter updatedCenter = centerService.updateCenter(id, center);

        if (updatedCenter != null) {

            return ResponseEntity.*ok*(updatedCenter);

        }

        return ResponseEntity.*notFound*().build();

    }

    @DeleteMapping("/{id}")

    public ResponseEntity<Void> deleteCenter(@PathVariable Long id) {

        centerService.deleteCenter(id);

        return ResponseEntity.*noContent*().build();

    }

}

**com.example.VaccinationCenter.dao**

**CitizenDao.java**

package com.example.VaccinationCenter.dao;

import java.util.List;

import java.util.Optional;

import java.util.function.Function;

import org.springframework.data.domain.Example;

import org.springframework.data.domain.Page;

import org.springframework.data.domain.Pageable;

import org.springframework.data.domain.Sort;

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

import org.springframework.data.repository.query.FluentQuery.FetchableFluentQuery;

import org.springframework.stereotype.Repository;

import com.example.VaccinationCenter.bean.CitizenBean;

@Repository

public interface CitizenDao extends JpaRepository<CitizenBean, Integer>{

}

**com.example.VaccinationCenter.dao**

**VaccinationCenterDao.java**

package com.example.VaccinationCenter.dao;

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

import org.springframework.stereotype.Repository;

import com.example.VaccinationCenter.bean.VaccinationCenterBean;

@Repository

public interface VaccinationCenterDao extends JpaRepository<VaccinationCenterBean, Integer>{

}

**com.example.VaccinationCenter.service**

**CitizenService.java**

package com.example.VaccinationCenter.service;

import com.example.entity.Citizen;

import java.util.List;

public interface CitizenService {

    List<Citizen> getAllCitizens();

    Citizen getCitizenById(Long id);

    Citizen createCitizen(Citizen citizen);

    Citizen updateCitizen(Long id, Citizen citizen);

    void deleteCitizen(Long id);

}

**com.example.VaccinationCenter.service**

**CitizenServiceImpl.java**

package com.example.VaccinationCenter.service;

import com.example.entity.Citizen;

import com.example.repository.CitizenRepository;

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

import org.springframework.stereotype.Service;

import java.util.List;

@Service

public class CitizenServiceImpl implements CitizenService {

    @Autowired

    private CitizenRepository citizenRepository;

    @Override

    public List<Citizen> getAllCitizens() {

        return citizenRepository.findAll();

    }

    @Override

    public Citizen getCitizenById(Long id) {

        return citizenRepository.findById(id).orElse(null);

    }

    @Override

    public Citizen createCitizen(Citizen citizen) {

        return citizenRepository.save(citizen);

    }

    @Override

    public Citizen updateCitizen(Long id, Citizen citizen) {

        Citizen existingCitizen = getCitizenById(id);

        if (existingCitizen != null) {

            existingCitizen.setName(citizen.getName());

            existingCitizen.setEmail(citizen.getEmail());

            existingCitizen.setPhone(citizen.getPhone());

            existingCitizen.setAddress(citizen.getAddress());

            return citizenRepository.save(existingCitizen);

        }

        return null;

    }

    @Override

    public void deleteCitizen(Long id) {

        citizenRepository.deleteById(id);

    }

}

**com.example.VaccinationCenter.service**

**VaccinationCenterService.java**

package com.example.VaccinationCenter.service;

import com.example.entity.VaccineCenter;

import java.util.List;

public interface VaccinationCenterService {

    List<VaccineCenter> getAllCenters();

    VaccineCenter getCenterById(Long id);

    VaccineCenter createCenter(VaccineCenter center);

    VaccineCenter updateCenter(Long id, VaccineCenter center);

    void deleteCenter(Long id);

}

**com.example.VaccinationCenter.service**

**VaccinationCenterServiceImpl.java**

package com.example.VaccinationCenter.service;

import com.example.entity.VaccineCenter;

import com.example.repository.VaccineCenterRepository;

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

import org.springframework.stereotype.Service;

import java.util.List;

@Service

public class VaccinationCenterServiceImpl implements VaccinationCenterService {

    @Autowired

    private VaccineCenterRepository centerRepository;

    @Override

    public List<VaccineCenter> getAllCenters() {

        return centerRepository.findAll();

    }

    @Override

    public VaccineCenter getCenterById(Long id) {

        return centerRepository.findById(id).orElse(null);

    }

    @Override

    public VaccineCenter createCenter(VaccineCenter center) {

        return centerRepository.save(center);

    }

    @Override

    public VaccineCenter updateCenter(Long id, VaccineCenter center) {

        VaccineCenter existingCenter = getCenterById(id);

        if (existingCenter != null) {

            existingCenter.setName(center.getName());

            existingCenter.setLocation(center.getLocation());

            return centerRepository.save(existingCenter);

        }

        return null;

    }

    @Override

    public void deleteCenter(Long id) {

        centerRepository.deleteById(id);

    }

}

**src/main/resources/static/css**

**styles.css**

**body** {

font-family: *Arial,* *sans-serif*;

margin: *20px*;

}

**h1,** **h2,** **h3** {

text-align: *center*;

}

**nav** {

text-align: *center*;

margin-bottom: *20px*;

}

**nav** **a** {

margin: *0* *10px*;

text-decoration: *none*;

color: *#333*;

}

**ul** {

list-style-type: *none*;

padding: *0*;

}

**li** {

margin: *5px* *0*;

}

**form** {

text-align: *center*;

}

**input** {

margin: *5px*;

}

**button** {

margin: *5px*;

}

**src/main/resources/templates**

**index.html**

<!DOCTYPE html>

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

<head>

    <meta charset=*"UTF-8"*>

    <title>Vaccination Center Management</title>

    <link rel=*"stylesheet"* th:href=*"@{/css/styles.css}"*>

</head>

<body>

    <h1>Vaccination Center Management</h1>

    <nav>

        <a th:href=*"@{/citizens}"*>Citizens</a>

        <a th:href=*"@{/centers}"*>Vaccination Centers</a>

    </nav>

</body>

</html>

**src/main/resources/templates**

**citizens.html**

<!DOCTYPE html>

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

<head>

    <meta charset=*"UTF-8"*>

    <title>Citizens</title>

    <link rel=*"stylesheet"* th:href=*"@{/css/styles.css}"*>

</head>

<body>

    <h2>Citizens</h2>

    <ul>

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

            <span th:text=*"${citizen.name}"*>Name</span> - <span th:text=*"${citizen.vaccinationStatus}"*>Status</span>

        </li>

    </ul>

    <h3>Add Citizen</h3>

    <form th:action=*"@{/citizens}"* method=*"post"*>

        <input type=*"text"* name=*"name"* placeholder=*"Name"*>

        <input type=*"number"* name=*"age"* placeholder=*"Age"*>

        <input type=*"text"* name=*"vaccinationStatus"* placeholder=*"Vaccination Status"*>

        <input type=*"number"* name=*"vaccinationCenterId"* placeholder=*"Vaccination Center ID"*>

        <button type=*"submit"*>Add Citizen</button>

    </form>

</body>

</html>

**src/main/resources/templates**

**centers.html**

<!DOCTYPE html>

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

<head>

    <meta charset=*"UTF-8"*>

    <title>Vaccination Centers</title>

    <link rel=*"stylesheet"* th:href=*"@{/css/styles.css}"*>

</head>

<body>

    <h2>Vaccination Centers</h2>

    <ul>

        <li th:each=*"center : ${centers}"*>

            <span th:text=*"${center.name}"*>Name</span> - <span th:text=*"${center.city}"*>City</span>

        </li>

    </ul>

    <h3>Add Vaccination Center</h3>

    <form th:action=*"@{/centers}"* method=*"post"*>

        <input type=*"text"* name=*"name"* placeholder=*"Name"*>

        <input type=*"text"* name=*"city"* placeholder=*"City"*>

        <button type=*"submit"*>Add Center</button>

    </form>

</body>

</html>

**MySQL Database Connection**

CREATE DATABASE vaccinationcenter;

USE vaccinationcenter;

-- Table for Citizens

CREATE TABLE citizens (

   citizenid BIGINT AUTO\_INCREMENT PRIMARY KEY,

    citizenname VARCHAR(100) NOT NULL,

    citizenage VARCHAR(100) UNIQUE NOT NULL,

    vaccinationstatus VARCHAR(20),

    vaccinationstatusid INT

);

-- Table for Vaccine Centers

CREATE TABLE vaccination\_center (

    vaccinationid BIGINT AUTO\_INCREMENT PRIMARY KEY,

    vaccinationname VARCHAR(100) NOT NULL,

    location VARCHAR(255)

capacity

);