**2. Displaying User Feedback**

* **Database Code**

create database userdb;

USE userdb;

CREATE TABLE feedback (

id INT AUTO\_INCREMENT PRIMARY KEY, name VARCHAR(255) NOT NULL,

email VARCHAR(255) NOT NULL,

rating INT NOT NULL,

comment VARCHAR(1000)

);

select \* from feedback;

# application.properties

spring.datasource.url=jdbc:mysql://localhost:3306/userdb?useSSL=false&se rverTimezone=UTC

spring.datasource.username=root spring.datasource.password=Root

spring.jpa.show-sql=true

spring.jpa.hibernate.ddl-auto=update

# logging.level.org.springframework=DEBUG

# FeedbackController.java

**package** com.ecommerce.FeedbackProject.controllers;

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

**import** org.springframework.http.MediaType;

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

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

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

**import** com.ecommerce.FeedbackProject.entity.Feedback;

**import** com.ecommerce.FeedbackProject.repository.FeedbackRepository; @RestController

**public class** FeedbackController {

**private final** FeedbackRepository feedbackRepository;

@Autowired

**public** FeedbackController(FeedbackRepository feedbackRepository) {

**this**.feedbackRepository = feedbackRepository;

}

@PostMapping(value = "/api/feedback", consumes = MediaType.***APPLICATION\_FORM\_URLENCODED\_VALUE***)

**public** String createFeedback(@RequestParam("comment") String comment, @RequestParam("email") String email,

@RequestParam("name") String name, @RequestParam("rating") **int** rating) {

Feedback feedback = **new** Feedback(name, email, rating,

comment);

}

feedbackRepository.save(feedback);

**return** "Feedback submitted successfully";

}

# FeedbackRepository.java

package com.ecommerce.FeedbackProject.repository;

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

import com.ecommerce.FeedbackProject.entity.Feedback;

@Repository

public interface FeedbackRepository extends JpaRepository<Feedback, Long> {

}

# Feedback.java

**package** com.ecommerce.FeedbackProject.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;

@Entity

@Table(name = "feedback")

**public class** Feedback { @Id

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

**private** Long id; @Column(nullable = **false**) **private** String name; @Column(nullable = **false**) **private** String email; @Column(nullable = **false**) **private int** rating; @Column

**private** String comment;

**public** Feedback() {

}

**public** Feedback(String name, String email, **int** rating, String comment) {

**this**.name = name; **this**.email = email; **this**.rating = rating; **this**.comment = comment;

}

**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 int** getRating() {

**return** rating;

}

**public void** setRating(**int** rating) {

**this**.rating = rating;

}

**public** String getComment() {

**return** comment;

}

**public void** setComment(String comment) {

**this**.comment = comment;

}

}

# FeedbackProjectApplication.java

**package** com.ecommerce.FeedbackProject;

**import** org.springframework.boot.SpringApplication;

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

**import** org.springframework.context.annotation.ComponentScan;

@SpringBootApplication

**public class** FeedbackProjectApplication {

**public static void** main(String[] args) {

SpringApplication.*run*(FeedbackProjectApplication.**class**, args);

}

}

# index.html

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<title>Feedback Form</title>

</head>

<body>

<h1>Feedback Form</h1>

<form id=*"feedbackForm"* method=*"POST"* action=*"/api/feedback"* enctype=*"application/json"*>

<label for=*"name"*>Name:</label> <input type=*"text"* id=*"name"* name=*"name"* required><br>

<br> <label for=*"email"*>Email:</label> <input type=*"email"* id=*"email"* name=*"email"* required><br>

<br> <label for=*"rating"*>Rating:</label> <input type=*"number"* id=*"rating"* name=*"rating"* min=*"1"* max=*"5"* required><br>

<br> <label for=*"comment"*>Comment:</label><br>

<textarea id=*"comment"* name=*"comment"* rows=*"4"* cols=*"50"*></textarea>

<br>

<br>

<button type=*"submit"*>Submit Feedback</button>

</form>

</body>

</html>

# 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.ecommerce</groupId>

<artifactId>FeedbackProject</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>FeedbackProject</name>

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

<properties>

<java.version>20</java.version>

</properties>

<dependencies>

<dependency>

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

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

</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-data-jpa</artifactId>

</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>