**User Feedback-Source Code:**

**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>UerFeedback</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>UerFeedback</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>

<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>

</dependencies>

<build>

<plugins>

<plugin>

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

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

</plugin>

</plugins>

</build>

</project>

**Feedback.java:**

**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;

@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;

// Constructors, getters, and setters

**public** Feedback() {

}

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

**this**.name = name;

**this**.email = email;

**this**.rating = rating;

**this**.comment = comment;

}

// Getters and setters

**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;

}

}

**FeedbackController:**

**package** com.ecommerce.controller;

**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.entity.Feedback;

**import** com.ecommerce.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:**

**package** com.ecommerce.repository;

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

**import** org.springframework.stereotype.Repository;

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

@Repository

**public** **interface** FeedbackRepository **extends** JpaRepository<Feedback, Long> {

}

**index.html**

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>User Feedback</title>

<style>

body {

font-family: Arial, sans-serif;

margin: 20px;

}

h1 {

text-align: center;

}

form {

max-width: 400px;

margin: 0 auto;

}

label {

display: block;

margin-top: 10px;

}

input[type="text"],

input[type="email"],

textarea {

width: 100%;

padding: 5px;

font-size: 16px;

}

button[type="submit"] {

display: block;

margin-top: 20px;

padding: 10px 20px;

background-color: #4CAF50;

color: #fff;

border: none;

cursor: pointer;

}

button[type="submit"]:hover {

background-color: #45a049;

}

</style>

</head>

<body>

<h1>User Feedback</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>

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

<input type="email" id="email" name="email" required>

<label for="rating">Rating:</label>

<input type="number" id="rating" name="rating" min="1" max="5" required>

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

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

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

</form>

</body>

</html>

**application.properties:**

spring.datasource.url=jdbc:mysql://localhost:3306/ecommerce?useSSL=false&serverTimezone=UTC

spring.datasource.username=root

spring.datasource.password=Ajitdk@26042000

spring.jpa.show-sql=true

spring.jpa.hibernate.ddl-auto=update

**SQl Database:**

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)

);