**PHASE3-SECONDPROJECT-WRITEUP**

**Displaying User Feedback**

**Project objective:**

Create a Spring Boot project that will capture user feedback using a REST endpoint. The REST resource will take in parameters using HTTP POST. The feedback data will be then added to a database table.

**Background of the problem statement:**

As a part of developing an ecommerce web application, a REST resource is needed to capture user feedback. Feedback data will be received from third-party apps and websites. The data will be sent to the REST API which will collect feedback from various sources.

**You must use the following:**

● Eclipse as the IDE  
● Apache Tomcat as the web server  
● Spring Boot with Hibernate

**Following requirements should be met:**

● Create a MySQL table named feedback for storing feedback data  
● An entity class Feedback should be made with annotations to link it with the feedback table  
● A repository class should then map the entity class to the CrudRepository interface  
● Create a REST controller class to create the REST endpoint. It should take in parameters using the POST protocol  
● Data received in the REST controller will be then saved into the database  
● Create a test form in HTML to submit data to the REST endpoint to ensure it’s working  
● The step-by-step process involved in completing this task should be documented

This project has following subsections, namely:

1. Creating a Spring Boot Starter Project which is web enabled
2. Building the project
3. Publishing and starting the project
4. Pushing the code to your GitHub repositories

**1. Creating a Spring Boot Starter Project which is web enabled**

* Open Eclipse
* Go to the **File** menu. Choose **New->Other**
* In the **Wizards** list, select **Spring Boot->Spring Starter Project**
* In **Name,** enter UserFeedback, **Type** as Maven, **Packaging** as Jar, **Group** as com. UserFeedback, and **Package** as com. UserFeedbackClick on **Next**
* In the list of **Available** dependencies, scroll down to select mysqldatabase, starter-web, starter-data-jpa, Lombo, jstl, jasper dependencies
* Click on **Next**
* Click on **Finish**
* This will create the project files in the Project Explorer

**Class:UserFeedbackApplication**

package com.example.UserFeedback;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class UserFeedbackApplication {

public static void main(String[] args) {

SpringApplication.run(UserFeedbackApplication.class, args);

}

}

ServletInitializer

package com.example.UserFeedback;

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(UserFeedbackApplication.class);

}

}

FeedbackController

package com.example.UserFeedback.controllers;

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

import org.springframework.ui.ModelMap;

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

import org.springframework.web.bind.annotation.ResponseBody;

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

import com.example.UserFeedback.entities.Feedback;

import com.example.UserFeedback.repositories.FeedbackRepository;

import com.example.UserFeedback.services.FeedbackService;

@RestController

public class FeedbackController {

@Autowired

FeedbackService feedbackService;

@Autowired

FeedbackRepository feedbackRepository;

@GetMapping(value="/")

public String showIndexPage(ModelMap model){

return "<html>\n"

+ "<head>\n"

+ " <style>\n"

+ " .center {\n"

+ " text-align: center;\n"

+ " }\n"

+ " \n"

+ " </style>\n"

+ "</head>\n"

+ "<body style=\"background-color:lightblue;\">\n"

+ " <div class=\"center\">\n"

+ " <h1>User Feedback Page</h1>\n"

+ " \n"

+ " <h2 class=\"hello-title\">View and Send User Feedback</h2>\n"

+ " \n"

+ " <a href=\"/feedback\">View all feedback</a>\n"

+ " <br><br>\n"

+ " <form method=\"get\" action=\"update\">\n"

+ " <br><h3>Enter your feedback below:</h3>\n"

+ " <input type=\"text\" id=\"comment\" name=\"comment\" placeholder=\"Comment Here\" required>\n"

+ " <input type=\"number\" id=\"rating\" name=\"rating\" placeholder=\"Rating Here\" required>\n"

+ " <input type=\"text\" id=\"name\" name=\"name\" placeholder=\"Name Here\" required> \n"

+ " <input type=\"submit\" value=\"Enter\" />\n"

+ " </form>"

+ " </div>\n"

+ "</body>\n"

+ "</html>";

}

@GetMapping("/feedback")

public @ResponseBody String getAllFeedbacks() {

// This returns a JSON or XML with the Feedbacks

Iterable<Feedback> allFB = feedbackRepository.findAll();

return "<html>\n"

+ "<head>\n"

+ " <style>\n"

+ " .center {\n"

+ " text-align: center;\n"

+ " }\n"

+ " \n"

+ " </style>\n"

+ "</head>\n"

+ "<body style=\"background-color:lightblue;\">\n"

+ " <div class=\"center\">\n"

+ "<h1>Feedback Table</h1>\n"

+ allFB.toString()

+ " </div>\n"

+ "</body>\n"

+ "</html>";

}

}

RestUpdateController

package com.example.UserFeedback.controllers;

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

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

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

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

import com.example.UserFeedback.entities.Feedback;

import com.example.UserFeedback.repositories.FeedbackRepository;

import com.example.UserFeedback.services.FeedbackService;

@RestController

public class RestUpdateController

{

@Autowired

FeedbackService feedbackService;

@Autowired

FeedbackRepository feedbackRepository;

@GetMapping("/update")

public String getEmployeeByName(@RequestParam("comment") String comment, @RequestParam("rating") int rating, @RequestParam("name") String name) {

Feedback f = new Feedback(count()+1, comment, rating, name);

feedbackRepository.save(f);

return "<html>\n"

+ "<head>\n"

+ " <style>\n"

+ " .center {\n"

+ " text-align: center;\n"

+ " }\n"

+ " \n"

+ " </style>\n"

+ "</head>\n"

+ "<body style=\"background-color:lightblue;\">\n"

+ " <div class=\"center\">\n"

+ " <h1>User Feedback Page</h1>\n"

+ " \n"

+ " <h2 class=\"hello-title\">Successfully Added Your Feedback</h2>\n"

+ " \n"

+ " <a href=\"/feedback\">Click here to view all feedback</a>\n"

+ " </div>\n"

+ "</body>\n"

+ "</html>";

}

public Integer count() {

int i = 1;

while(feedbackRepository.existsById(i))

i++;

return i;

}

}

Feedback

package com.example.UserFeedback.entities;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import lombok.Data;

@Data

@Entity

public class Feedback {

@Id

@GeneratedValue(strategy=GenerationType.AUTO)

private Integer id;

private String comments;

private int rating;

private String user;

public Feedback()

{

}

public Feedback(Integer id, String comments, int rating, String user)

{

this.id = id;

this.comments = comments;

this.rating = rating;

this.user = user;

}

@Override

public String toString()

{

return "<br><h3>" + user + " [" + id + "]" + " commented:</h3><h4>\"" + comments + "\" and rated: " + rating + "</h4><br>";

}

}

FeedbackRepository

package com.example.UserFeedback.repositories;

import org.springframework.data.repository.CrudRepository;

import com.example.UserFeedback.entities.Feedback;

public interface FeedbackRepository extends CrudRepository<Feedback, Integer>{

}

FeedbackService

package com.example.UserFeedback.services;

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

import org.springframework.stereotype.Service;

import com.example.UserFeedback.entities.Feedback;

import com.example.UserFeedback.repositories.FeedbackRepository;

@Service

public class FeedbackService {

@Autowired

private FeedbackRepository feedbackRepository;

public Iterable<Feedback> GetAllFeedback() {

return feedbackRepository.findAll();

}

}

Application.properties:

spring.jpa.hibernate.ddl-auto=update

spring.datasource.url=jdbc:mysql://${MYSQL\_HOST:localhost}:3306/springauth

spring.datasource.username=root

spring.datasource.password=root

server.port=8080

spring.mvc.view.prefix=/webapp/jsp/

spring.mvc.view.suffix=.jsp

index.jsp

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

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<style>

*.center* {

text-align: *center*;

}

</style>

</head>

<body style="background-color:*lightblue*;">

<div class=*"center"*>

<h1>User Manager</h1>

<h2 class=*"hello-title"*>Search for a User and Update Information</h2>

<a href=*"/feedback"*>View all feedback</a>

<br><br>

<form method=*"post"* action=*"update"*>

Enter an id from the user table: <input type=*"text"* id=*"id"* name=*"id"* placeholder=*"Type here"* required><input type=*"submit"* value=*"Enter"* />

</form>

</div>

</body>

</html>

UserFeedback/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>2.7.1</version>

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

</parent>

<groupId>com.example</groupId>

<artifactId>UserFeedback</artifactId>

<version>0.0.1-SNAPSHOT</version>

<packaging>war</packaging>

<name>UserFeedback</name>

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

<properties>

<java.version>1.8</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>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<scope>runtime</scope>

</dependency>

<dependency>

<groupId>org.projectlombok</groupId>

<artifactId>lombok</artifactId>

<optional>true</optional>

</dependency>

<dependency>

<groupId>javax.servlet</groupId>

<artifactId>jstl</artifactId>

</dependency>

<dependency>

<groupId>org.apache.tomcat.embed</groupId>

<artifactId>tomcat-embed-jasper</artifactId>

<scope>provided</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-test</artifactId>

<scope>test</scope>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

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

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

<configuration>

<excludes>

<exclude>

<groupId>org.projectlombok</groupId>

<artifactId>lombok</artifactId>

</exclude>

</excludes>

</configuration>

</plugin>

</plugins>

</build>

</project>

MYSQL TABLE

create database springauth;

use springauth;

CREATE TABLE `feedback` (

`id` int NOT NULL AUTO\_INCREMENT,

`comments` varchar(255) DEFAULT NULL,

`rating` int NOT NULL,

`user` varchar(255) DEFAULT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=4 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci ;

insert into feedback (comments, rating, user) VALUES ("Awesome", 10, "tim");

insert into feedback (comments, rating, user) VALUES ("Great", 9, "Sim");

insert into feedback (comments, rating, user) VALUES ("Nice", 7, "Bim");

select \* from feedback;

**2. Building the project**

* From the **Project** menu at the top, click on **Build**
* If any compile errors are shown, fix them as required

**3. Publishing and starting the project**

* In the Project Explorer, right click on **SpringBootStarter->Run As->Spring Boot App**
* Check in the Eclipse Console for the message **Started SpringBootStarterApplication**

**4. Pushing the code to your GitHub repositories:**

* Open your command prompt and navigate to the folder where you have created your files.

**cd <folder path>**

* Initialize your repository using the following command:

**git init**

* Add all the files to your git repository using the following command:

**git add .**

* Commit the changes using the following command:

**git commit . -m “Changes have been committed.”**

* Push the files to the folder you initially created using the following command:

**git push -u origin master**