

```
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);
```

```
    }
```

```
}
```

```
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.*;

@Service
public class FeedbackService {

    @Autowired
    private FeedbackRepository feedbackRepository;

    public Iterable<Feedback> GetAllFeedback() {
        return feedbackRepository.findAll();
    }

    public Feedback saver(Feedback feedback) {
        //isBlank()
        feedbackRepository.save(feedback);
        return feedback;
    }

}
```

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<h1>hey</h1>
<form action="save" method="post">
    <label for="comment">Comments:</label><br>
    <input type="text" id="coment" name="comment" value="${feed.comments}"><br>
    <label for="rating">Rating:</label><br>
    <input type="text" id="rating" name="rating" value="${feed.rating}"><br>
    <label for="user">User:</label><br>
    <input type="text" id="user" name="user" value="${feed.user}"><br>
    <input type="submit" value="save">
</form>
</body>
</html>
```

```

package com.example.UserFeedback.entities;

import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;

import lombok.Data;

@Entity
public class Feedback {

    @Id
    @GeneratedValue(strategy=GenerationType.IDENTITY)
    private Integer id;
    private String comments;
    private int rating;
    private String user;

    public Feedback() {
        super();
    }
    public Feedback(Integer id, String comments, int rating, String user) {
        super();
        this.id = id;
        this.comments = comments;
        this.rating = rating;
        this.user = user;
    }
    @Override
    public String toString() {
        return "Feedback [id=" + id + ", comments=" + comments + ", rating=" +
rating + ", user=" + user + "]";
    }
    public Integer getId() {
        return id;
    }
    public void setId(Integer id) {
        this.id = id;
    }
    public String getComments() {
        return comments;
    }
    public void setComments(String comments) {
        this.comments = comments;
    }
    public int getRating() {
        return rating;
    }
    public void setRating(int rating) {
        this.rating = rating;
    }
}

```

```
    public String getUser() {  
        return user;  
    }  
    public void setUser(String user) {  
        this.user = user;  
    }  
}
```

```

package com.example.UserFeedback.controllers;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.ui.ModelMap;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.RequestParam;

import com.example.UserFeedback.entities.Feedback;
import com.example.UserFeedback.exception.InvalidInputException;
import com.example.UserFeedback.services.FeedbackService;

@Controller
public class TestFormController {

    @Autowired
    private FeedbackService fbs;

    @GetMapping(value="/")
    public String showTestForm(ModelMap model) {
        //TODO: Create a JSP called "testform.jsp"
        Iterable<Feedback> feed = fbs.GetAllFeedback();
        model.addAttribute("feeds", feed);
        return "index";
    }

    @RequestMapping(value="/save", method=RequestMethod.POST)//url
    public String saveUser(@RequestParam("comment") String com,
    @RequestParam("rating") int rate, @RequestParam("user") String user, ModelMap mod) {

        if ((com == null)|| (rate == 0)|| (user == null)) {
            throw new InvalidInputException();
        }

        Feedback feed = new Feedback();
        feed.setComments(com);
        feed.setRating(rate);
        feed.setUser(user);

        fbs.saver(feed);

        //mod.addAttribute("users", uList);//attribute variable different
        from .jsp file

        return "index";//name of .jsp file
    }

    // @PostMapping("/testform")

```

```
    // TODO: implement form submission
    // TODO: add View JSP
    // TODO: Call RestTemplate and make POST json request to
    localhost:8090/feedback
}
```

```

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.PathVariable;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.ResponseBody;
import org.springframework.web.bind.annotation.RestController;

import com.example.UserFeedback.entities.Feedback;
import com.example.UserFeedback.services.FeedbackService;

@RestController
public class FeedbackController {

    @Autowired
    FeedbackService feedbackService;

    @GetMapping(value = "/feedback", produces = "application/json")
    public @ResponseBody Iterable<Feedback> getAllFeedbacks() {
        // This returns a JSON or XML with the Feedbacks
        return feedbackService.GetAllFeedback();
    }

    @PostMapping(value = "/feedback", produces = "application/json")
    public Feedback addNewFeedback(@RequestBody Feedback newFeedback) {
        Feedback addNewFeedback(@RequestBody Feedback newFeedback) {
            // TODO: Do something useful here.
            // Add the Feedback.
            return feedbackService.saver(newFeedback); // Change this.
            //return new feedback();
        }
    }
}

```



```
package com.example.UserFeedback.repositories;

import org.springframework.data.repository.CrudRepository;

import com.example.UserFeedback.entities.*;

public interface FeedbackRepository extends CrudRepository<Feedback, Integer>{

}
```

```
spring.jpa.hibernate.ddl-auto=update
spring.datasource.url=jdbc:mysql://${MYSQL_HOST:localhost}:3306/db_example
spring.datasource.username=user
spring.datasource.password=password
```

```
logging.level.org.springframework.web: DEBUG
spring.mvc.view.prefix=/
spring.mvc.view.suffix=.jsp
server.port=8090
```

```

<?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>2.4.3</version>
    <relativePath/> <!-- lookup parent from repository -->
  </parent>
  <groupId>com.example</groupId>
  <artifactId>UserFeedback</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <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>
      <version>1.2</version>
    </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-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>
```

```
package com.example.UserFeedback.controllers;

import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springframework.web.bind.annotation.ControllerAdvice;
import org.springframework.web.bind.annotation.ExceptionHandler;
import com.example.UserFeedback.exception.InvalidInputException;

@ControllerAdvice
public class ExceptionController {
    @ExceptionHandler(value = InvalidInputException.class)
    public ResponseEntity<Object> exception(InvalidInputException exception) {
        return new ResponseEntity<>("Invalid Input, please try again!",
        HttpStatus.UNPROCESSABLE_ENTITY);
    }
}
```

```
package com.example.UserFeedback.exception;

public class InvalidInputException extends RuntimeException {
    private static final long serialVersionUID = 1L;
}
```

