

CODE

FeedBack

```
package com.example.demo;

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;
        }
    }
}

```

FeedBackController

```

package com.example.demo;

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;

@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";
}
}
```

FeedBackFormApplication

```
package com.example.demo;

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

@SpringBootApplication

public class FeedBackFormApplication {

    public static void main(String[] args) {
        SpringApplication.run(FeedBackFormApplication.class, args);
    }

}
```

FeedBackProjectApplication

```
package com.example.demo;

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);  
    }  
}
```

FeedBackRepository

```
package com.example.demo;  
  
import org.springframework.data.jpa.repository.JpaRepository;  
import org.springframework.stereotype.Repository;  
  
@Repository  
public interface FeedbackRepository extends JpaRepository<Feedback, Long> {  
}
```

ServletInitializer

```
package com.example.demo;  
  
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(FeedBackFormApplication.class);  
    }  
}
```

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>
```

Application

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

spring.datasource.username=root

spring.datasource.password=nareshrao

spring.jpa.show-sql=true

spring.jpa.hibernate.ddl-auto=update