# PROJECT: DISPLAYING USER FEEDBACK

BY: ANUSHKA KAMATH

Git Repository Link: <https://github.com/anushkakamath/simplilearn_bnp_project2.git>

**DESCRIPTION:**

The developed Spring Boot project with Rest is a feedback system to record user feedback for a software product deployed. It accepts feedback from the user with respect to the following fields:

1. Quality of the product
2. Response time (speed of response time)
3. Satisfaction with the product
4. Overall experience
5. Likely to recommend the product to others.

Apart from this the user is also expected to enter his/her name into the record. The user is expected to give his feedback with respect to the above-mentioned fields in a numeric form from a scale of 0 to 5. The scale for each of the fields is 0 (worst) and 5 (excellent).

The responses of the user are stored into the Spring Boot embedded database called H2, where the id is generated automatically and user is supposed to enter the name and his feedback on a scale of 0 to 5 with respect to the 5 fields.

The operations available:

1. Create a new feedback (POST /feedback)

Creates a new feedback entry into the database with the data entered by user. This contains an exception handler and returns an error message in the header with a HTTP 404 error, in case the feedback entered for any of the fields is not numeric in the range 0 to 5.

1. Update an existing feedback (PUT /feedback)

Updates an existing feedback entry in the database. In case, an id is entered that does not exist in the database an error message is displayed in the header with an HTTP status code. This contains an exception handler and returns an error message in the header with a HTTP 404 error, in case the feedback entered for any of the fields is not numeric in the range 0 to 5.

1. Delete an existing feedback by id (DELETE /feedback/{id})
2. Get feedback by id (GET /feedback/{id})

This consists of an exception handler that gives a message in the header if an id less than 0 is entered or an id that doesn’t exist in the database table is entered

1. Get all the feedbacks in the database (GET /feedbacks)
2. Get all feedbacks by name (GET /feedbacks/name/{name})
3. Get feedbacks by satisfaction of the product (GET /feedbacks/{satisfactionOfProduct})
4. Get feedbacks by overall experience (GET /feedbacks/experience/{overallExperience})
5. Get feedbacks by likeness to recommend (GET /feedbacks/recommend/{likelyToRecommend)

**HOW TO RUN THE PROJECT**

The project is built in with Swagger dependency and can be easily run on Swagger UI to observe the results. These configurations are done in application.properties file and dependencies are installed from pom.xml

STEPS:

1. Run the spring boot project on the Spring Tool Suite
2. The project is running on server port 9100. Hence open the link on a browser: **localhost:9100/swagger-ui.html**
3. Click on feedback-controller to access all the operations provided in the project.
4. Note the feedback to be entered for each field should be on a scale of 0 to 5. 0 being the worst and 5 being the best.
5. To test POST or create a new feedback click on the post and then click on Example value which copies the feedback to the editable json script, remove the id as id is auto generated and add all the other fields and click on try this. This generates the result.
6. To test PUT or update an existing feedback, click on the post and then click on Example value which copies the feedback to the editable json script, enter id and all other fields, and click on try this. In case an invalid entry is entered it gives an message in the header section along with HTTP code, otherwise it successfully updates the entry.
7. Similarly, all other operations may be tested.

TO ACCESS THE DATABASE / VIEW H2 CONSOLE:

1. Open the link: **localhost:9100/h2**
2. Change the JDBC URL from what’s obtained on the console (jdbc:h2:mem:testdb)
3. Click on Connect. This opens the H2 database for viewing.

This project may also be run on Postman although, Swagger Ui is recommended.