

# Customer Sentiment Analysis Platform

## Problem Statement:

The platform will collect customer feedback from multiple channels (e.g., email, surveys, social media) and automatically analyze the sentiment using AI-powered Natural Language Processing (NLP). Based on the sentiment analysis, real-time reports will be generated, and actionable insights will be provided, such as notifications to the customer service team for negative feedback and recommendations for improvement.

## Objective:

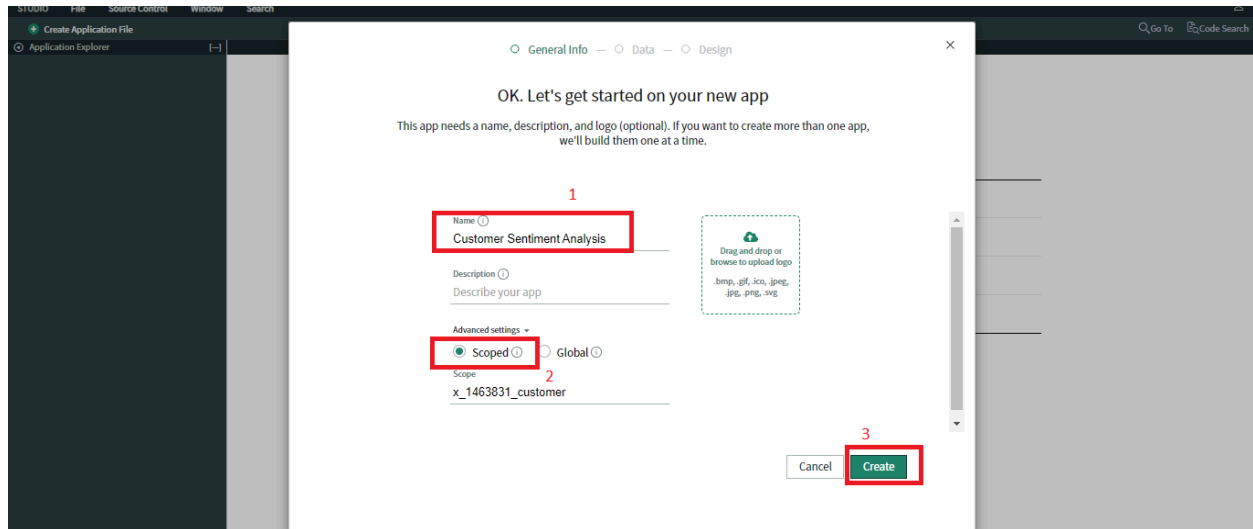
Automatically analyze customer feedback using AI to generate sentiment scores, notify the customer service team of negative feedback, and trigger follow-up actions to improve customer satisfaction.

## TASK INITIATION

### Milestone 1 : Studio

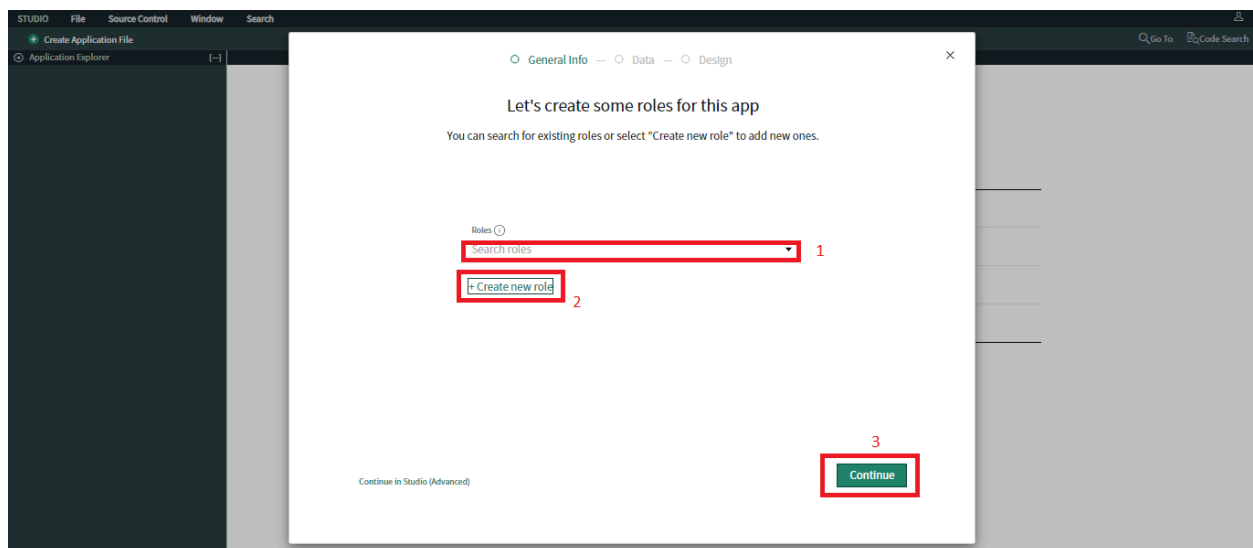
#### Activity 1: Create Application in Studio

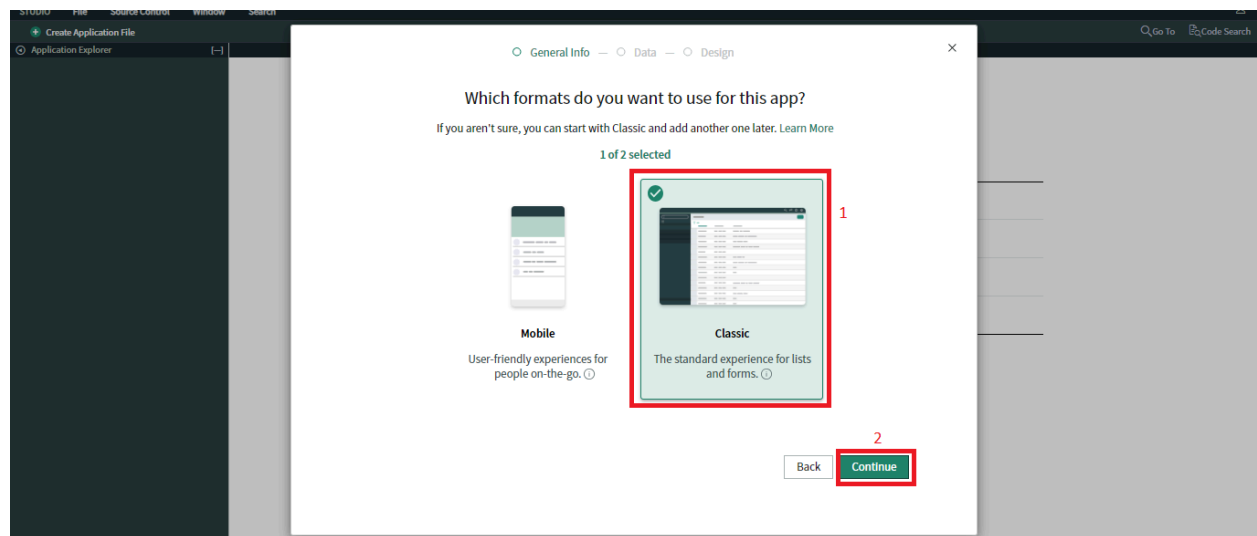
- 1) Open service now.
- 2) Click on All >> search for studio
- 3) Select studio under system applications
- 4) Click on create application
- 5) Fill in the details and click on create



## Activity 2 : Create Users

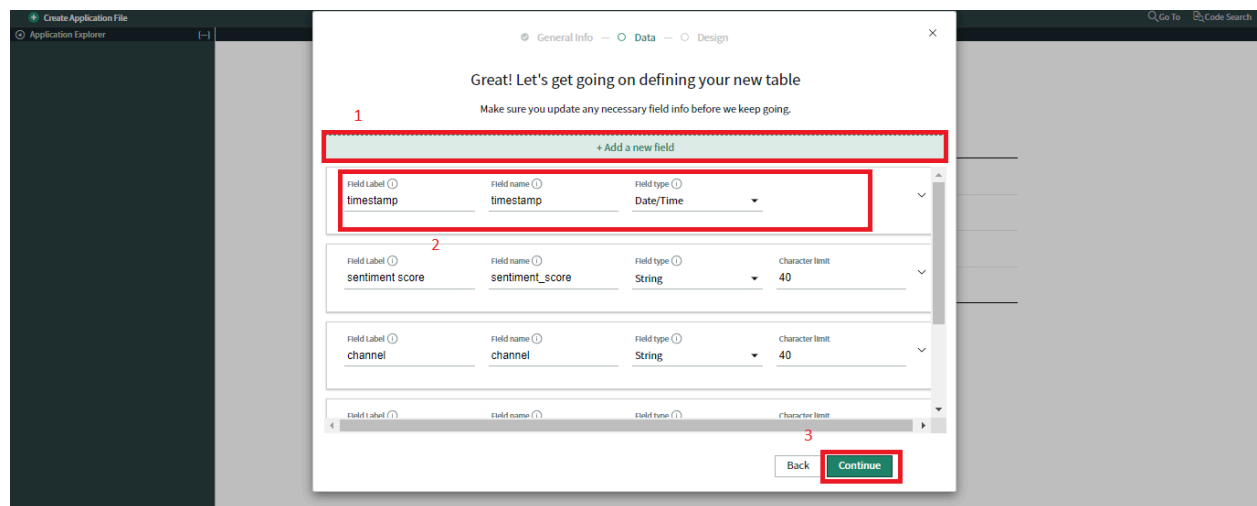
- 1) Click on create roles and create
- 2) Customer AI model trainer, customer sentiment analyst, customer feedback collector and customer platform administrator
- 3) Click on continue
- 4) Select Classic and click on continue

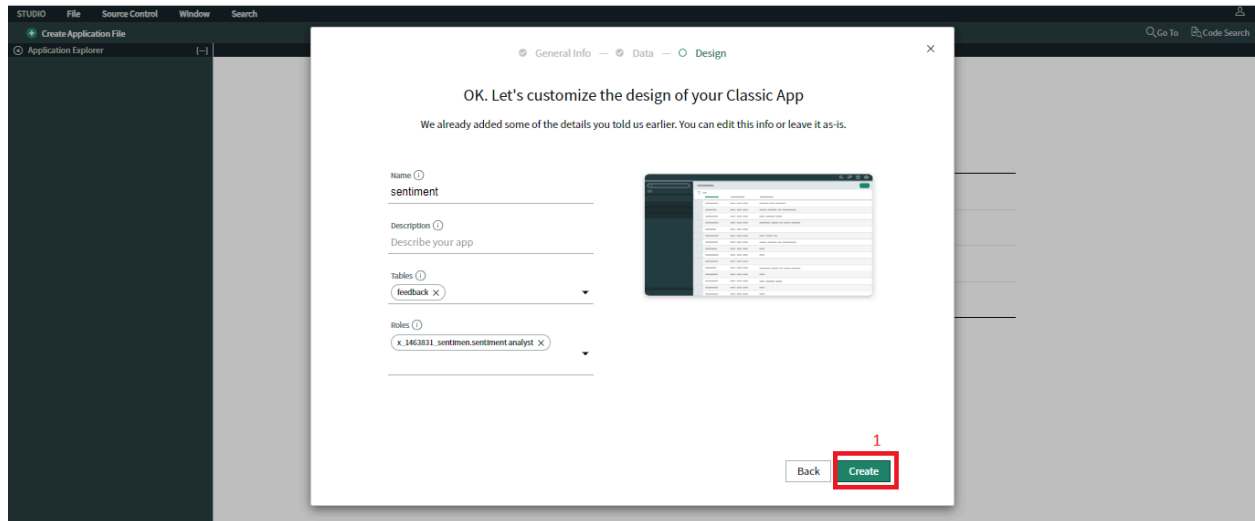




### Activity 3 : Create Tables

- 1) Click on create new table and continue
- 2) Click on create table from scratch and continue
- 3) Add new fields feedback id, feedback text , channel sentiment score, timestamp
- 4) Click on continue and done with tables
- 5) Give table name customer feedback and click continue , done with tables
- 6) Click on start and create

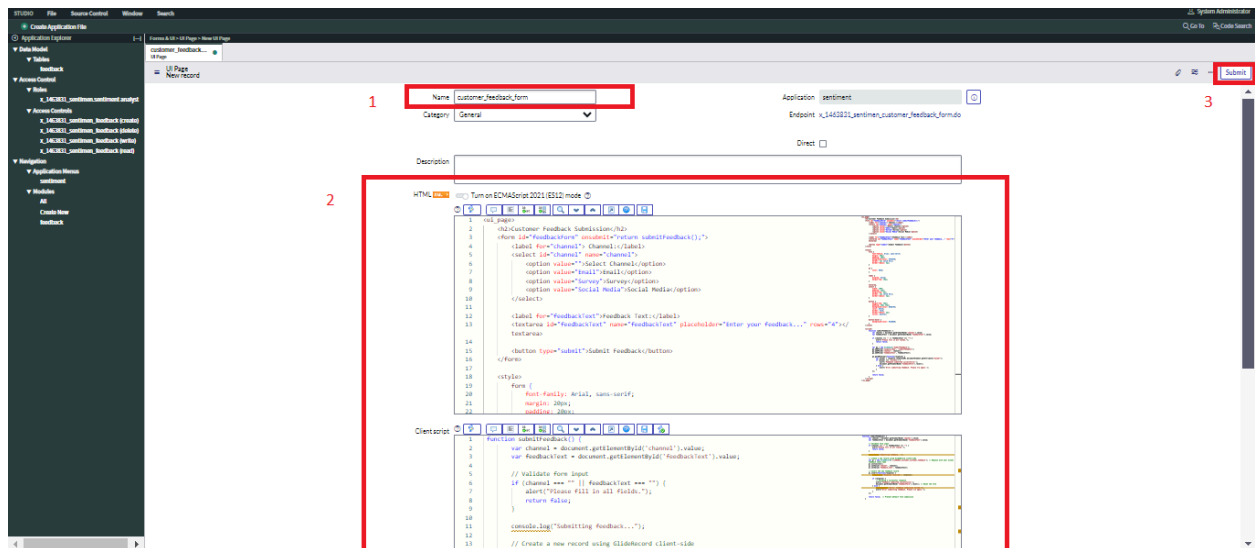




## Milestone 2 : Create UI Page

### Activity 1 : UI page

- 1) In the Studio ,click on create application file
- 2) Select UI page and click on create
- 3) Fill in the details ; Name : Customer feedback form , write html code and client script



## HTML CODE :

```
<ui_page>
  <h2>Customer Feedback Submission</h2>
  <form id="feedbackForm" onsubmit="return submitFeedback();">
    <label for="channel"> Channel:</label>
    <select id="channel" name="channel">
      <option value="">Select Channel</option>
      <option value="Email">Email</option>
      <option value="Survey">Survey</option>
      <option value="Social Media">Social Media</option>
    </select>

    <label for="feedbackText">Feedback Text:</label>
    <textarea id="feedbackText" name="feedbackText" placeholder="Enter your feedback..."
rows="4"></textarea>

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

  <style>
    form {
      font-family: Arial, sans-serif;
      margin: 20px;
      padding: 20px;
      background-color: #f9f9f9;
      border: 1px solid #ccc;
      border-radius: 5px;
    }

    h2 {
      color: #333;
```

```
label {

    display: block;
    margin-top: 10px;
}

textarea,
select {
    width: 100%;
    padding: 10px;
    margin-top: 5px;
    border: 1px solid #ccc;
    border-radius: 4px;
}

button {
    margin-top: 10px;
    padding: 10px 15px;
    background-color: #28a745;
    color: white;
    border: none;
    border-radius: 4px;
    cursor: pointer;
}

button:hover {
    background-color: #218838;
}

</style>

<script>
function submitFeedback() {
    var channel = document.getElementById('channel').value;
```

```
var feedbackText = document.getElementById('feedbackText').value;

if (channel === "" || feedbackText === "") {

    alert("Please fill in all fields.");
    return false;
}

var ga = new GlideAjax('SubmitFeedback');
ga.addParam('sysparm_name', 'submitFeedback');
ga.addParam('channel', channel);
ga.addParam('feedbackText', feedbackText);

ga.getXMLAnswer(function(response) {
    var answer = response.responseXML.documentElement.getAttribute("answer");
    if (answer === 'success') {
        alert('Feedback submitted successfully!');
        document.getElementById('feedbackForm').reset();
    } else {
        alert('Error submitting feedback. Please try again.');
```

CLIENT SCRIPT CODE :

```
function submitFeedback() {
    var channel = document.getElementById('channel').value;
    var feedbackText = document.getElementById('feedbackText').value;

    // Validate form input
```

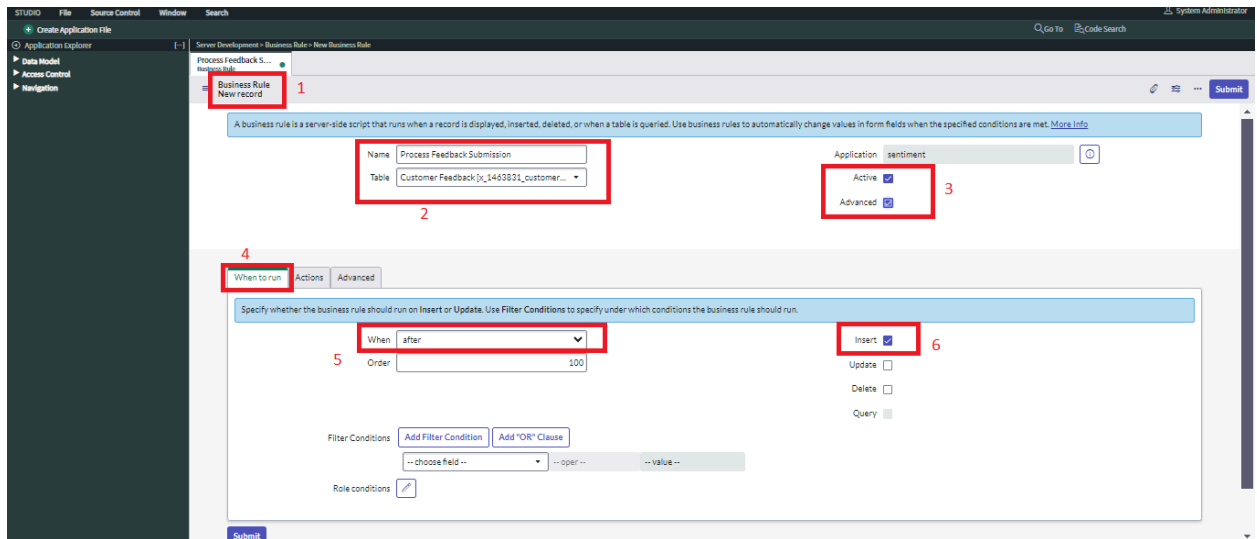
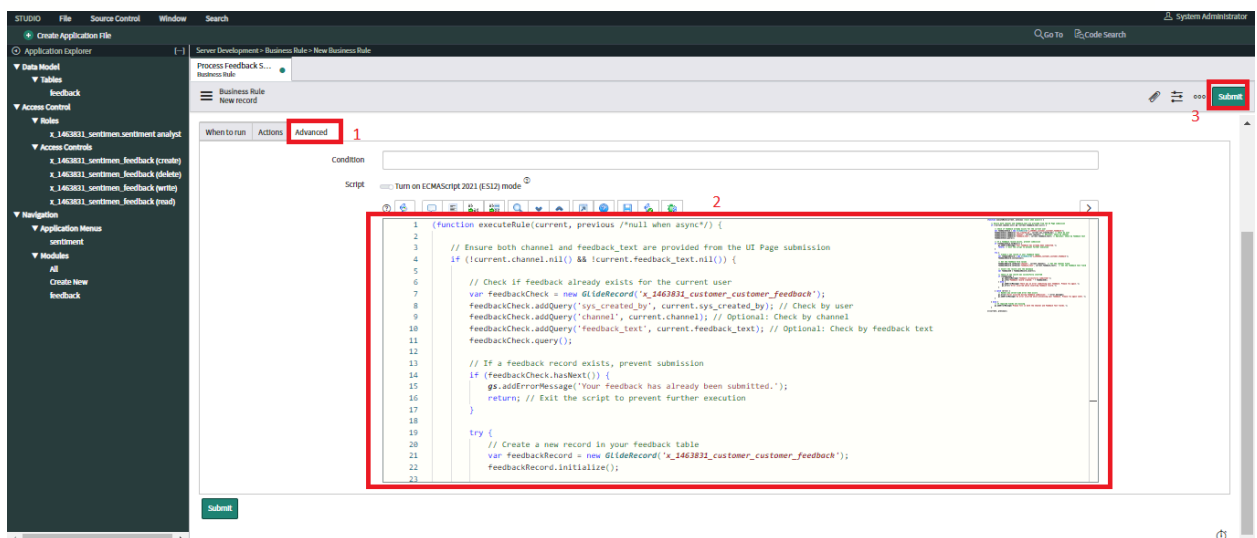
```
if (channel === "" || feedbackText === "") {  
    alert("Please fill in all fields.");  
    return false;  
  
}  
  
console.log("Submitting feedback...");  
  
// Create a new record using GlideRecord client-side  
var gr = new GlideRecord('x_1463831_customer_customer_feedback'); // Replace with your  
actual feedback table name  
gr.initialize();  
gr.setValue('channel', channel);  
gr.setValue('feedback_text', feedbackText);  
  
// Insert the new feedback record  
gr.insert(function(response) {  
    console.log("Response received:", response);  
  
    if (response) {  
        // Assuming a successful response  
        alert('Feedback submitted successfully!');  
        document.getElementById('feedbackForm').reset(); // Reset the form  
    } else {  
        console.error("Error: feedback submission failed.");  
        alert("Error submitting feedback. Please try again.");  
    }  
});  
  
return false; // Prevent default form submission  
}
```

## Activity 2 : Create business rule

- 1) Click on create application file ,



- 2) Select business rule and click on create
- 3) Fill in the details
- 4) Name : Process feedback form , Table : customer feedback
- 5) Check the active and advanced check box
- 6) Under when to run section select after and check the insert box
- 7) Under the advanced section ,write the script

```

1 (function executeRule(current, previous /*null when async*/) {
2
3 // Ensure both channel and feedback_text are provided from the UI Page submission
4 if (!current.channel.nil() && !current.feedback_text.nil()) {
5
6 // Check if feedback already exists for the current user
7 var feedbackCheck = new GlideRecord("x_1463831_customer_customer_feedback");
8 feedbackCheck.addQuery('sys_created_by', current.sys_created_by); // Check by user
9 feedbackCheck.addQuery('channel', current.channel); // Optional: Check by channel
10 feedbackCheck.addQuery('feedback_text', current.feedback_text); // Optional: Check by feedback text
11 feedbackCheck.query();
12
13 // If a feedback record exists, prevent submission
14 if (feedbackCheck.hasNext()) {
15   gs.addErrorMessage('Your feedback has already been submitted.');

```

SCRIPT :

```
(function executeRule(current, previous /*null when async*/) {
```

```
// Ensure both channel and feedback_text are provided from the UI Page submission
if (!current.channel.nil() && !current.feedback_text.nil()) {

    // Check if feedback already exists for the current user
    var feedbackCheck = new GlideRecord('x_1463831_customer_customer_feedback');
    feedbackCheck.addQuery('sys_created_by', current.sys_created_by); // Check by user
    feedbackCheck.addQuery('channel', current.channel); // Optional: Check by channel
    feedbackCheck.addQuery('feedback_text', current.feedback_text); // Optional: Check by
feedback text
    feedbackCheck.query();

    // If a feedback record exists, prevent submission
    if (feedbackCheck.hasNext()) {
        gs.addErrorMessage('Your feedback has already been submitted.');
```

```
        return; // Exit the script to prevent further execution
    }
}

try {
    // Create a new record in your feedback table
    var feedbackRecord = new GlideRecord('x_1463831_customer_customer_feedback');
    feedbackRecord.initialize();
    // Set the feedback form values
    feedbackRecord.setValue('channel', current.channel); // Set the channel field
    feedbackRecord.setValue('feedback_text', current.feedback_text); // Set the feedback
text field

    // Insert the record into the database
    var feedbackID = feedbackRecord.insert();

    // Check if the record was successfully inserted
    if (feedbackID) {
        gs.addInfoMessage('Feedback successfully submitted!');
        gs.info('Feedback record created: ' + feedbackID);
    } else {
```

```
gs.addErrorMessage('There was an error submitting your feedback. Please try  
again.');
```

```
gs.error('Error occurred while inserting feedback record.');
```

```
    }
```

```
  } catch (error) {  
    // Handle any server-side error that occurs  
    gs.error('Error occurred during feedback submission: ' + error.message);  
    gs.addErrorMessage('An error occurred while processing your feedback. Please try  
again later.');
```

```
  }
```

```
  } else {  
    // If required fields are missing  
    gs.addErrorMessage('Please fill in both the Channel and Feedback Text fields.');
```

```
  }
```

```
})(current, previous);
```

## **Milestone 3 : Business Rule**

### **Activity 1 : Business rule**

- 1) Click on create application file and select business rule and create
- 2) Fill in the details
- 3) Name : Analyze customer feedback sentiment , Table : customer feedback
- 4) Check the active and advanced box
- 5) Under the when to run section ,select before and check insert and update box
- 6) Under the advanced section ,write the script

Create Application File

Application Explorer

- Data Model
  - Table
    - Feedback
- Server Development
  - Business Rules
    - Process Feedback Submission
- Access Control
  - Rules
    - x\_1463831\_sentimen\_sentiment\_analyst
  - Access Controls
    - x\_1463831\_sentimen\_feedback (create)
    - x\_1463831\_sentimen\_feedback (delete)
    - x\_1463831\_sentimen\_feedback (write)
    - x\_1463831\_sentimen\_feedback (read)
- Navigation
  - Application Menus
    - sentimen
  - Modules
    - All
    - Create New
    - Feedback

Server Development - Business Rule - New Business Rule

Business Rule

1

A business rule is a server-side script that runs when a record is displayed, inserted, deleted, or when a table is queried. Use business rules to automatically change values in form fields when the specified conditions are met. [More Info](#)

Name: Analyze Customer Feedback Sentiment

Table: Customer Feedback[x\_1463831\_customer\_...]

Application: sentiment

Active: ☒

Advanced: ☒

2

3

4

When to run: When

Order: 100

5

Insert: ☒

Update: ☒

6

Delete: ☐

Query: ☐

Filter Conditions: Add Filter Condition Add "OR" Clause

Role conditions: [Add Role Condition](#)

Submit

Create Application File

Application Explorer

- Data Model
  - Tables
    - Feedback
- Server Development
  - Business Rules
    - Process Feedback Submission
- Access Control
  - Rules
    - x\_1463831\_sentimen\_sentiment\_analyst
  - Access Controls
    - x\_1463831\_sentimen\_feedback (create)
    - x\_1463831\_sentimen\_feedback (delete)
    - x\_1463831\_sentimen\_feedback (write)
    - x\_1463831\_sentimen\_feedback (read)
- Navigation
  - Application Menus
    - sentimen
  - Modules
    - All
    - Create New
    - Feedback

Server Development - Business Rule - New Business Rule

Business Rule

When to run: When

Order: 100

Advanced

1

Condition

Script

2

```

1 // Business Rule: Analyze Sentiment
2 // Table: Customer Feedback
3 // When: Before (Insert and Update)
4
5 if (current.feedback_text) {
6   // Initialize sentiment score
7   var sentimentScore = 0; // Default score
8
9   // Analyze sentiment based on the feedback text
10  if (current.feedback_text.includes("good") || current.feedback_text.includes("great")) {
11    sentimentScore = 1; // Positive sentiment
12  } else if (current.feedback_text.includes("bad") || current.feedback_text.includes("poor")) {
13    sentimentScore = -1; // Negative sentiment
14  } else {
15    sentimentScore = 0; // Neutral sentiment if no keywords match
16  }
17
18  // Store the sentiment score back in the record
19  current.sentiment_score = sentimentScore; // Ensure 'sentiment_score' field exists in your table
20
21

```

Submit

SCRIPT :

```

// Business Rule: Analyze Sentiment
// Table: Customer Feedback

```



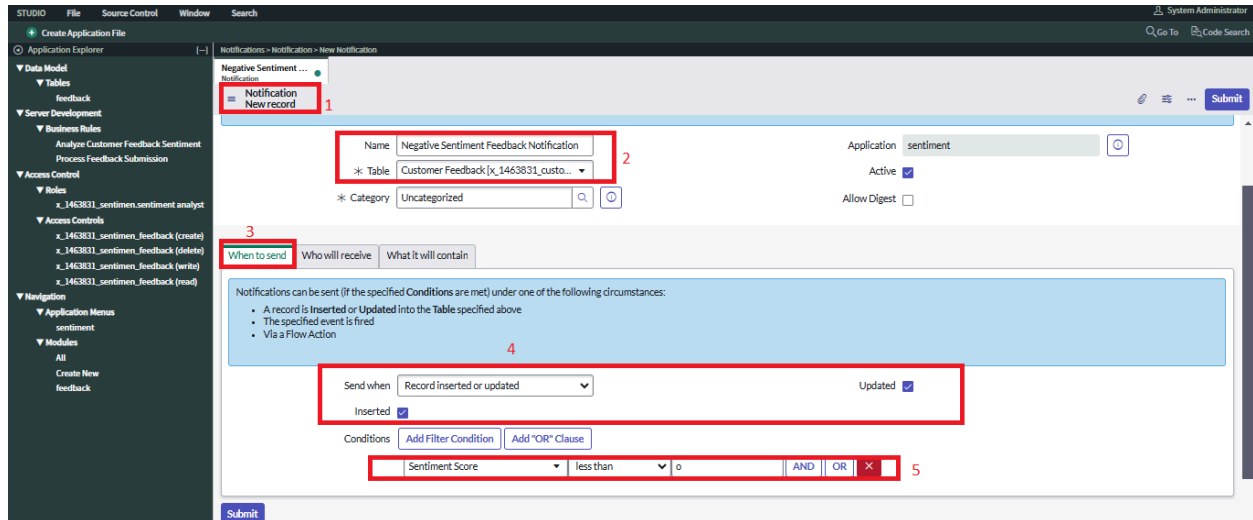
```
if (current.feedback_text) {  
    // Initialize sentiment score  
  
    var sentimentScore = 0; // Default score  
  
    // Analyze sentiment based on the feedback text  
    if (current.feedback_text.includes("good") ||  
current.feedback_text.includes("great")) {  
        sentimentScore = 1; // Positive sentiment  
    } else if (current.feedback_text.includes("bad") ||  
current.feedback_text.includes("poor")) {  
        sentimentScore = -1; // Negative sentiment  
    } else {  
        sentimentScore = 0; // Neutral sentiment if no keywords  
match  
    }  
  
    // Store the sentiment score back in the record  
    current.sentiment_score = sentimentScore; // Ensure  
'sentiment_score' field exists in your table  
}
```

## Milestone 4 : Notification

### Activity 1 : Create notification

- 1) Click on create application file and select notification and create
- 2) Fill in the details
- 3) Name : Negative Sentiment Feedback Notification , Table : customer feedback
- 4) Under 'when to send' section select record inserted or updated and give the condition as 'sentiment score less than 0'.
- 5) Under 'who will receive' section select few users

- 6) Under 'what will it contain' section give subject 'New negative feedback sentiment' and write html message.



**1** Notification New record

**2** Name: Negative Sentiment Feedback Notification

\* Table: Customer Feedback [x\_1463831\_custo...]

\* Category: Uncategorized

Application: sentiment

Active: ☒

Allow Digest: ☐

**3** When to send

Who will receive: [ ]

What it will contain: [ ]

Notifications can be sent (if the specified Conditions are met) under one of the following circumstances:

- A record is Inserted or Updated into the Table specified above
- The specified event is fired
- Via a Flow Action

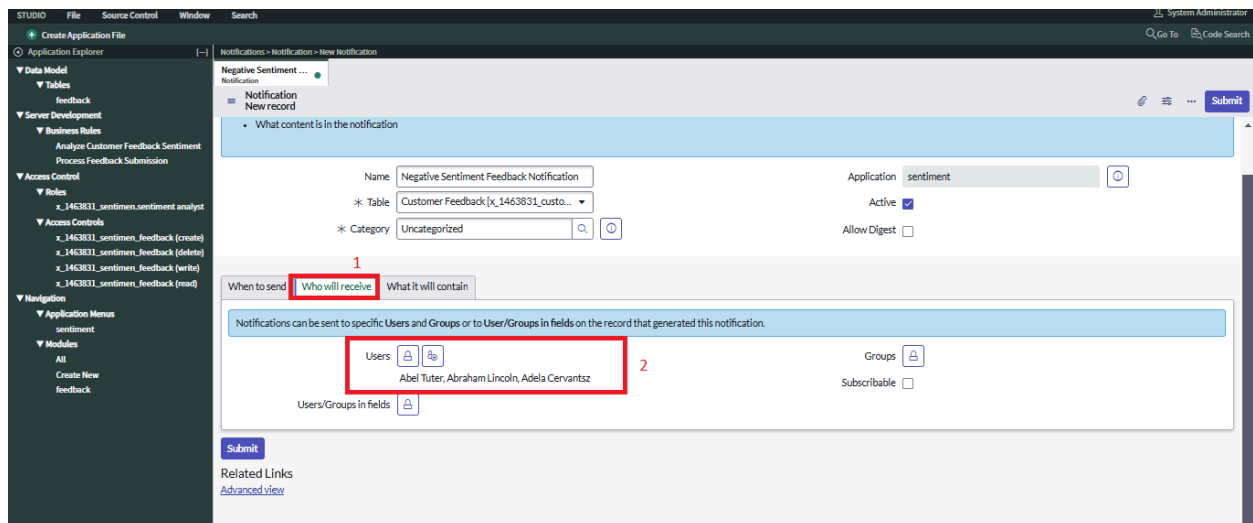
**4** Send when: Record inserted or updated

Updated: ☒

Inserted: ☒

Conditions: Add Filter Condition Add "OR" Clause

Sentiment Score less than 0 AND OR **5**



**1** Notification New record

What content is in the notification: [ ]

Name: Negative Sentiment Feedback Notification

\* Table: Customer Feedback [x\_1463831\_custo...]

\* Category: Uncategorized

Application: sentiment

Active: ☒

Allow Digest: ☐

**2** Who will receive

What it will contain: [ ]

Notifications can be sent to specific Users and Groups or to User/Groups in fields on the record that generated this notification.

Users: [ ]

Abel Tuter, Abraham Lincoln, Adela Cervantsz

Groups: [ ]

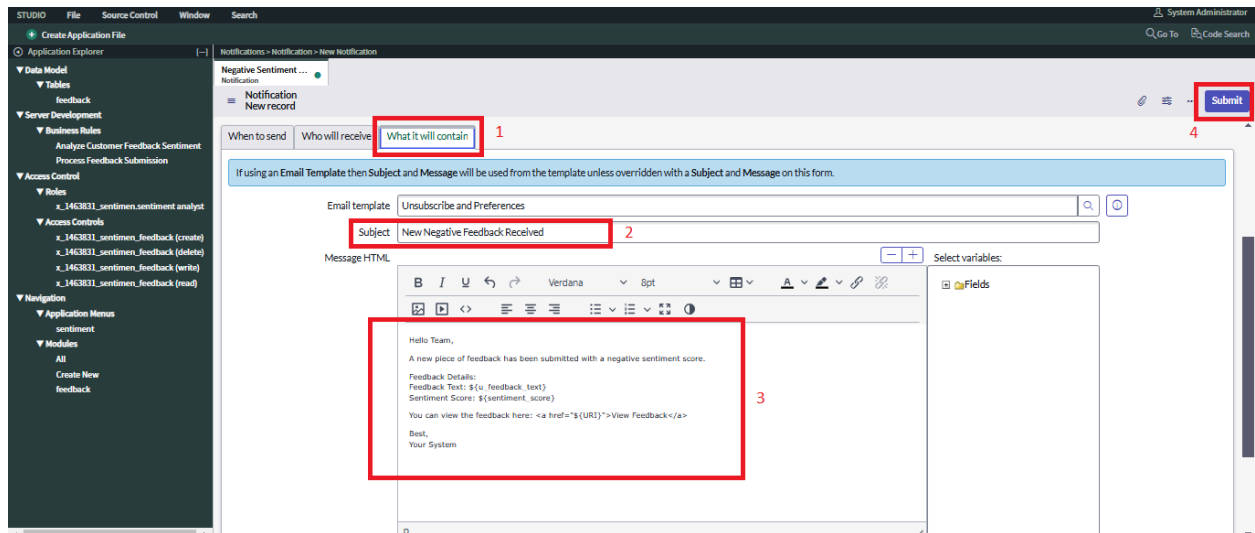
Subscribable: ☐

Users/Groups in fields: [ ]

Submit

Related Links

[Advanced view](#)



HTML MESSAGE :

Hello Team,

A new piece of feedback has been submitted with a negative sentiment score.

Feedback Details:

Feedback Text: `${u_feedback_text}`

Sentiment Score: `${sentiment_score}`

You can view the feedback here: `<a href="${URI}">View Feedback</a>`

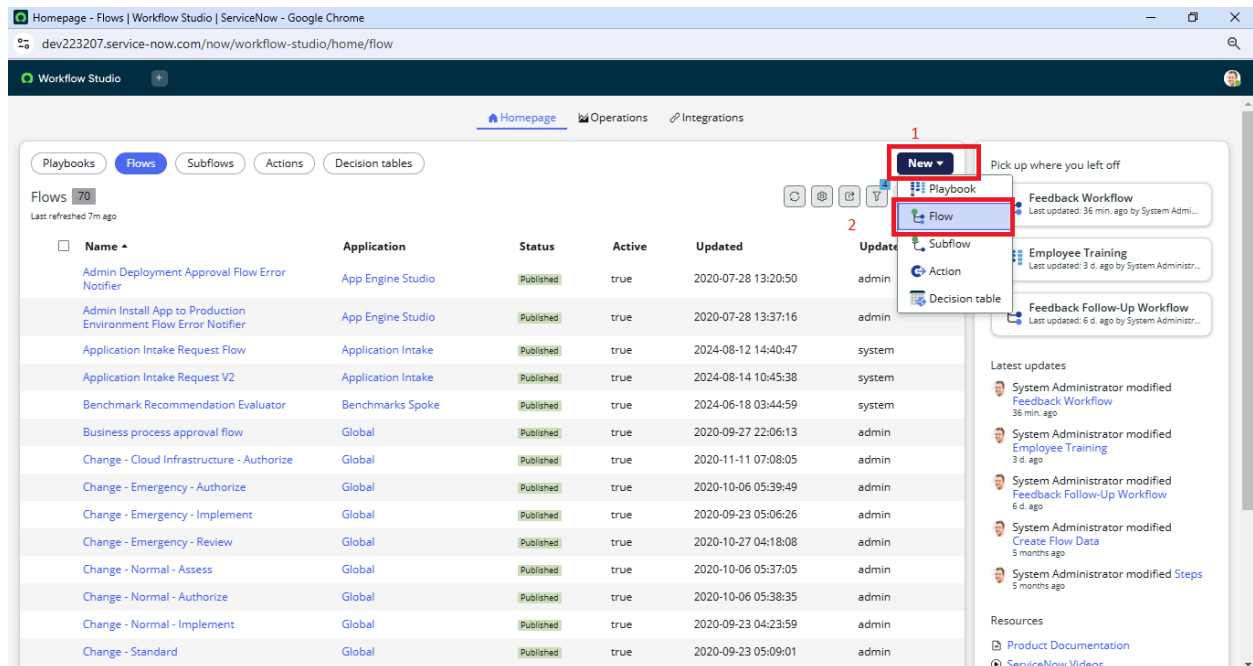
Best,

Your System

## Activity 2 : Create Flow

- 1) Click on create application file and select flow and create.

- 2) Click on 'New' and select 'Flow'
- 3) Give Name 'Customer feedback Flow' and click on build flow.
- 4) Under the Trigger select 'Created or updated', Table : 'customer feedback, Run trigger : for every update and click done.
- 5) Under Actions , Action : create task, Table : incident , Condition : ' short description is follow up on negative feedback' and 'assignment group is problem solving' and click on done
- 6) Click on save and activate
- 7) Then click on test



Homepage - Flows | Workflow Studio | ServiceNow - Google Chrome

dev223207.service-now.com/now/workflow-studio/home/flow

Workflow Studio

Homepage Operations Integrations

Playbooks **Flows** Subflows Actions Decision tables

Flows 70  
Last refreshed 7m ago

Name	Application	Status	Active	Updated	Updated by
Admin Deployment Approval Flow Error Notifier	App Engine Studio	Published	true	2020-07-28 13:20:50	admin
Admin Install App to Production Environment Flow Error Notifier	App Engine Studio	Published	true	2020-07-28 13:37:16	admin
Application Intake Request Flow	Application Intake	Published	true	2024-08-12 14:40:47	system
Application Intake Request V2	Application Intake	Published	true	2024-08-14 10:45:38	system
Benchmark Recommendation Evaluator	Benchmarks Spoke	Published	true	2024-06-18 03:44:59	system
Business process approval flow	Global	Published	true	2020-09-27 22:06:13	admin
Change - Cloud Infrastructure - Authorize	Global	Published	true	2020-11-11 07:08:05	admin
Change - Emergency - Authorize	Global	Published	true	2020-10-06 05:39:49	admin
Change - Emergency - Implement	Global	Published	true	2020-09-23 05:06:26	admin
Change - Emergency - Review	Global	Published	true	2020-10-27 04:18:08	admin
Change - Normal - Assess	Global	Published	true	2020-10-06 05:37:05	admin
Change - Normal - Authorize	Global	Published	true	2020-10-06 05:38:35	admin
Change - Normal - Implement	Global	Published	true	2020-09-23 04:23:59	admin
Change - Standard	Global	Published	true	2020-09-23 05:09:01	admin

Latest updates

- System Administrator modified [Feedback Workflow](#) 36 min. ago
- System Administrator modified [Employee Training](#) 3 d. ago
- System Administrator modified [Feedback Follow-Up Workflow](#) 6 d. ago
- System Administrator modified [Create Flow Data](#) 5 months ago
- System Administrator modified [Steps](#) 5 months ago

Resources

- [Product Documentation](#)
- [ServiceNow Videos](#)



New Flow | Workflow Studio | ServiceNow - Google Chrome

dev223207.service-now.com/now/workflow-studio/builder%3FtypeSysId%3D2d85e527439231106c4bb0117fb8f208%26sysId%3D-1

Workflow Studio

Feedback Workflow

New Flow

Let's get the details for your flow

1

Flow name \*

customer feedback flow

Description

Describe your flow.

Application \*

sentiment

> Show additional properties

2

Cancel Build flow

customer feedback flow | Workflow Studio | ServiceNow - Google Chrome

dev223207.service-now.com/now/workflow-studio/builder%3Ftable%3Dsys\_hub\_flow%26sysId%3Dfb91223b83595a107b3cec80cead3a2

Workflow Studio

Feedback Workflow

customer feedback flow

customer feedback flow

Trigger

Customer Feedback Created or Updated

Trigger: Created or Updated

\* Table: Customer Feedback [x\_1463831...

Condition: Add filters

Run Trigger: For every update

1

Advanced Options

Delete Cancel Done

2

ACTIONS

Select multiple

1

Create Task

Add an Action, Flow Logic, or Subflow

ERROR HANDLER

If an error occurs in your flow, the actions you add here will run.

Data

Collapse All

Flow Variables

Trigger - Record Created or Updated

Customer Feedback Record

Changed Fields

Customer Feedback Table

Run Start Time UTC

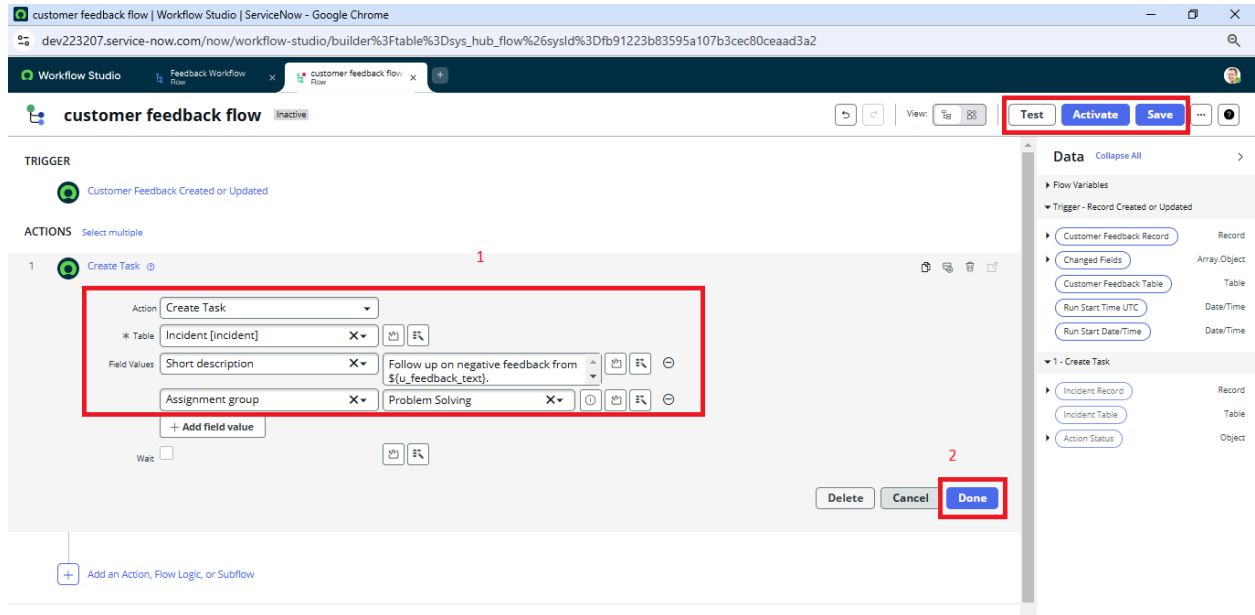
Run Start Date/Time

1 - Create Task

Incident Record

Incident Table

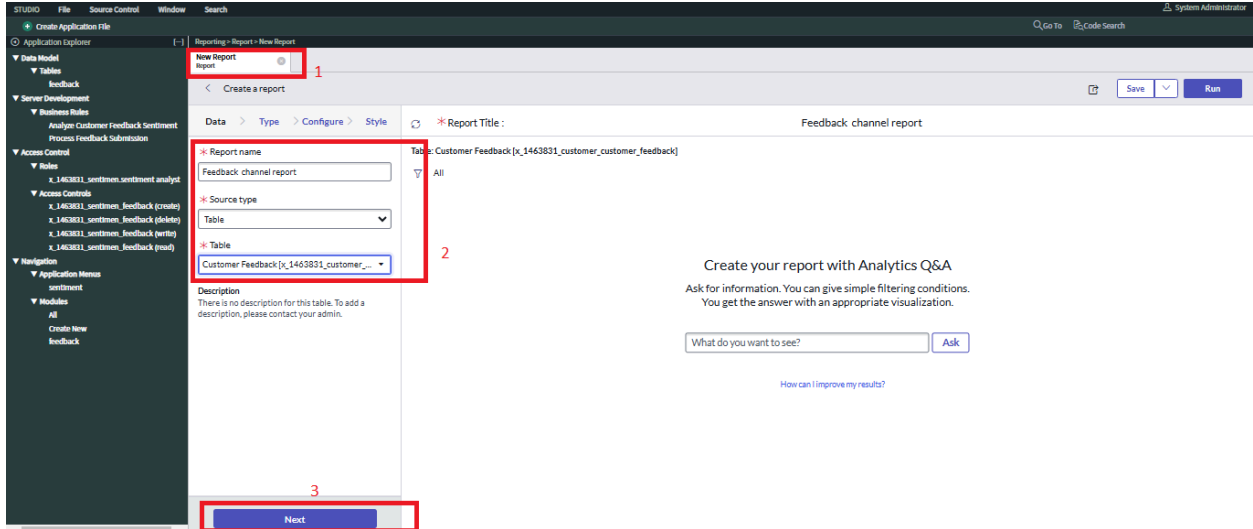
Action Status



## Milestone 5 : Reports

### Activity 1 : Create Reports

- 1) Click on create application file and select report and create
- 2) Fill the details
- 3) Under Data ; Name : Feedback channel report , Data source : table , Table : customer feedback and click on next
- 4) Under Type ; Select bars and click on next
- 5) Under configure ; Group by : feedback text , Stack by : channel , Aggregation : average and select sentiment score and then click on next
- 6) Click on Save and Run



STUDIO File Source Control Window Search

Application Explorer

Data Model

Tables

Feedback

Server Development

Business Rules

Analyze Customer Feedback Sentiment

Process Feedback Submission

Access Control

Roles

x\_1463831\_sentimen\_sentiment analyst

Access Controls

x\_1463831\_sentimen\_feedback (create)

x\_1463831\_sentimen\_feedback (delete)

x\_1463831\_sentimen\_feedback (write)

x\_1463831\_sentimen\_feedback (read)

Navigation

Application Menu

sentiment

Modules

All

Create New

Feedback

Reporting > Report > New Report

New Report Report

Create a report

Report name: Feedback channel report

Source type: Table

Table: Customer Feedback [x\_1463831\_customer\_customer\_feedback]

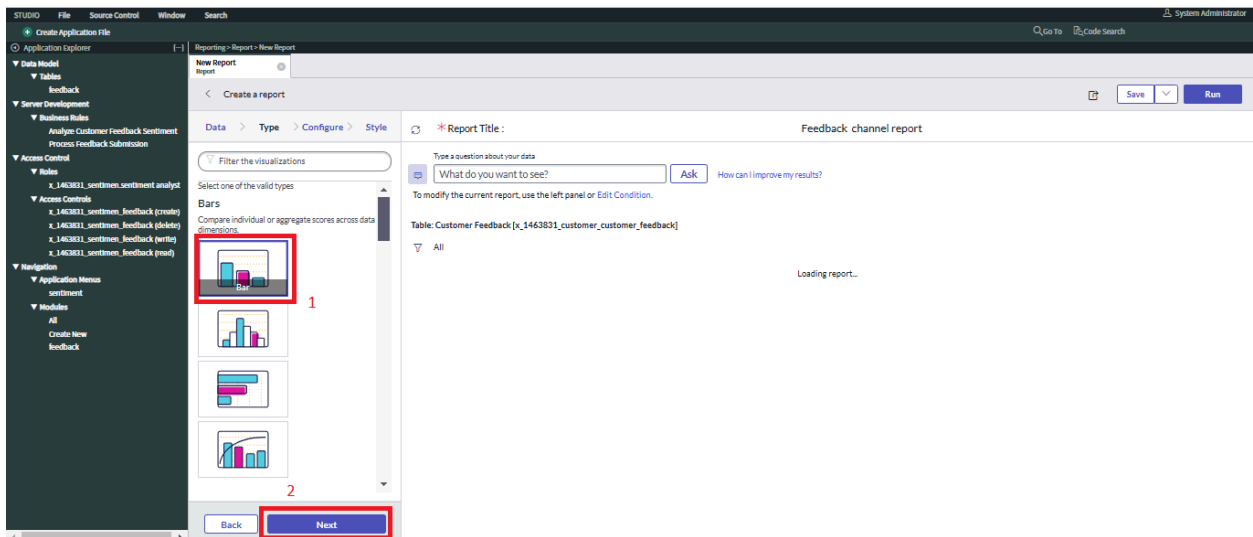
Report Title: Feedback channel report

Description: There is no description for this table. To add a description, please contact your admin.

What do you want to see? Ask

How can I improve my results?

Next



STUDIO File Source Control Window Search

Application Explorer

Data Model

Tables

Feedback

Server Development

Business Rules

Analyze Customer Feedback Sentiment

Process Feedback Submission

Access Control

Roles

x\_1463831\_sentimen\_sentiment analyst

Access Controls

x\_1463831\_sentimen\_feedback (create)

x\_1463831\_sentimen\_feedback (delete)

x\_1463831\_sentimen\_feedback (write)

x\_1463831\_sentimen\_feedback (read)

Navigation

Application Menu

sentiment

Modules

All

Create New

Feedback

Reporting > Report > New Report

New Report Report

Create a report

Report name: Feedback channel report

Source type: Table

Table: Customer Feedback [x\_1463831\_customer\_customer\_feedback]

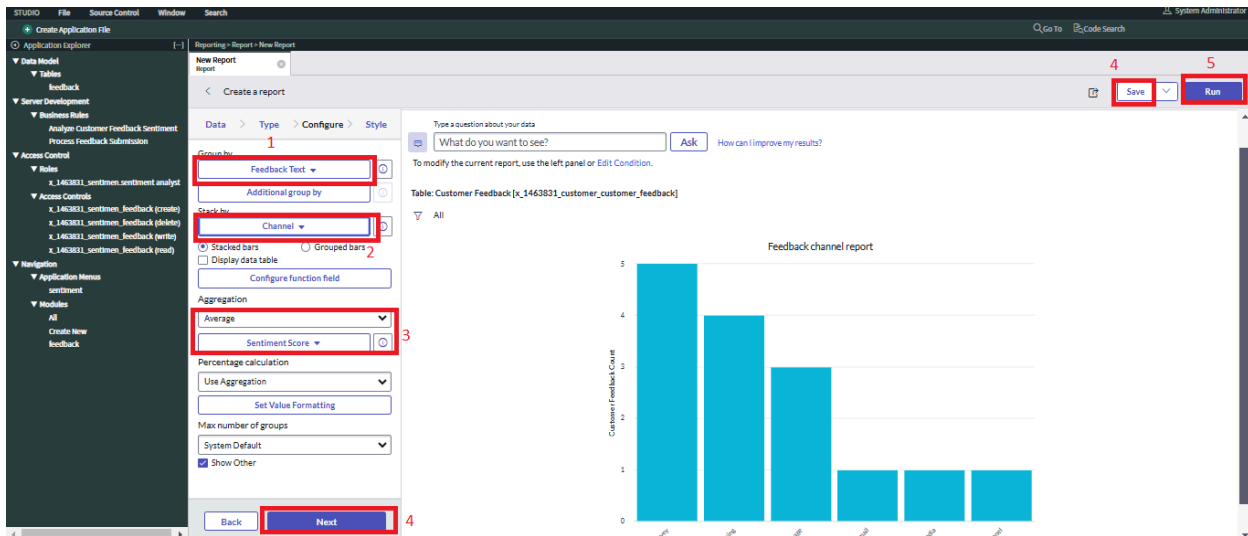
Report Title: Feedback channel report

Type: Bars

What do you want to see? Ask

How can I improve my results?

Back Next



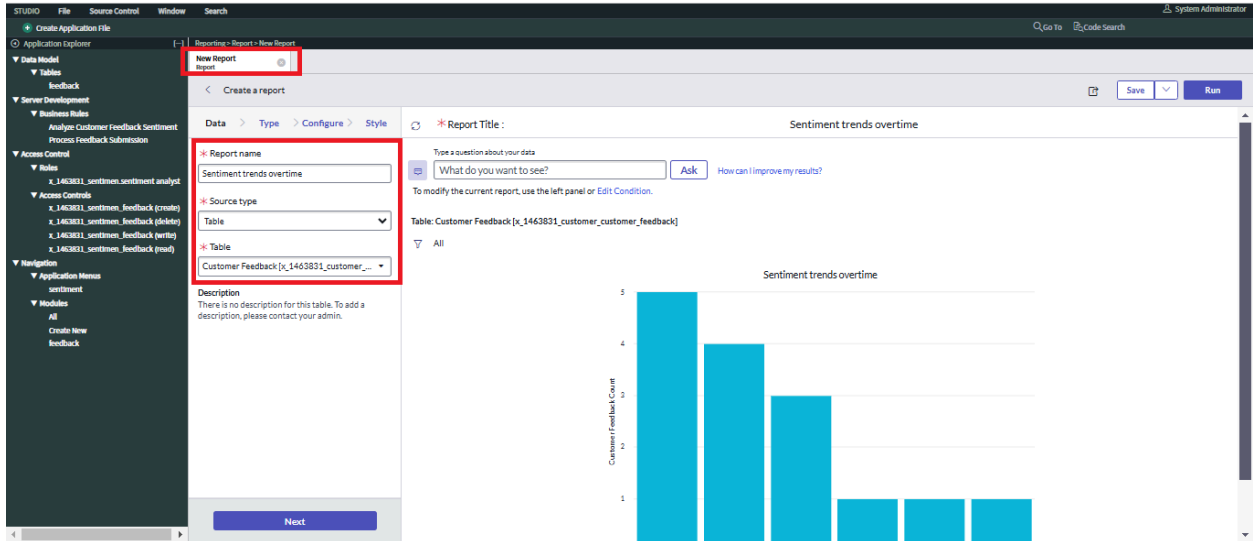
The screenshot shows the 'Create a report' wizard in the Smart Internz Studio. The 'Configure' tab is selected, and the following settings are visible:

- Group by:** Feedback Text (highlighted with a red box and number 1)
- Stack by:** Channel (highlighted with a red box and number 2)
- Aggregation:** Average (highlighted with a red box and number 3)
- Percentage calculation:** Use Aggregation
- Max number of groups:** System Default
- Show Other:** Checked

The 'Next' button is highlighted with a red box and a red number 4. The 'Save' and 'Run' buttons are also highlighted with red boxes and red numbers 4 and 5 respectively. A bar chart titled 'Feedback channel report' is displayed on the right, showing the distribution of sentiment scores across different channels.

## Activity 2 : Create another Report

- 1) Click on create application file and select report and create Fill the details
- 2) Under Data ; Name : Sentiment trends overtime , Data source : table , Table : customer feedback and click on next
- 3) Under Type ; Select bars and click on next
- 4) Under configure ; Group by : feedback text , Stack by : channel , Aggregation : average and select sentiment score and then click on next
- 5) Click on Save and Run



**Create a report**

**Report name:** Sentiment trends overtime

**Source type:** Table

**Table:** Customer Feedback [x\_1463831\_customer\_customer\_feedback]

**Description:** There is no description for this table. To add a description, please contact your admin.

**Report Title:** Sentiment trends overtime

Type a question about your data: What do you want to see? [Ask](#) [How can I improve my results?](#)

To modify the current report, use the left panel or [Edit Condition](#).

Table: Customer Feedback [x\_1463831\_customer\_customer\_feedback]

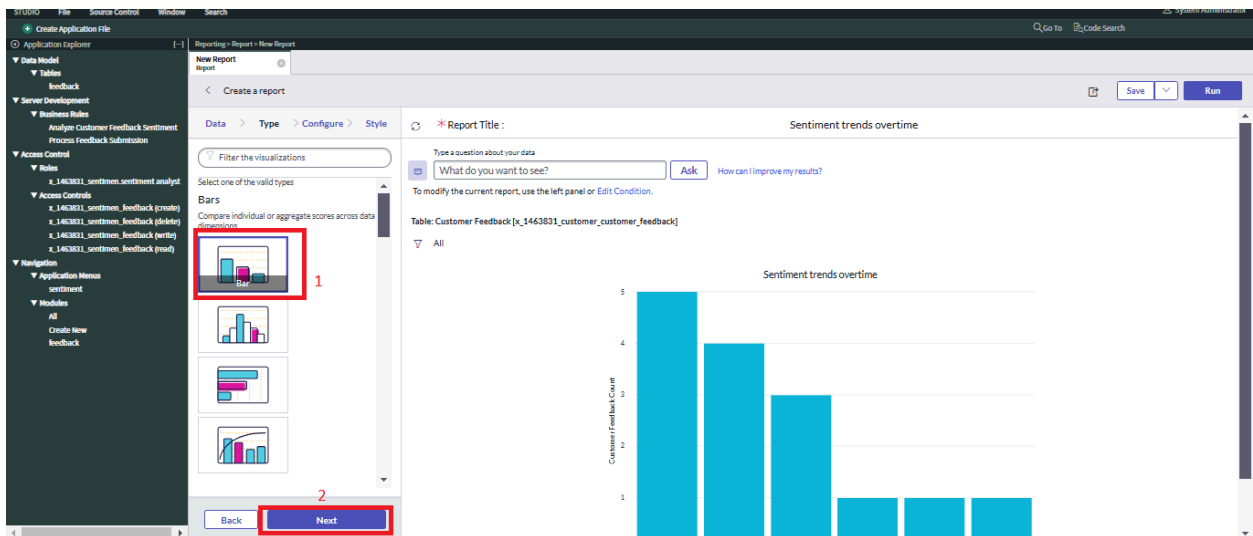
▽ All

**Sentiment trends overtime**

Customer Feedback Count

Category	Count
1	5
2	4
3	3
4	1
5	1
6	1

[Next](#)



**Create a report**

**Report name:** Sentiment trends overtime

**Source type:** Table

**Table:** Customer Feedback [x\_1463831\_customer\_customer\_feedback]

**Description:** There is no description for this table. To add a description, please contact your admin.

**Report Title:** Sentiment trends overtime

Type a question about your data: What do you want to see? [Ask](#) [How can I improve my results?](#)

To modify the current report, use the left panel or [Edit Condition](#).

Table: Customer Feedback [x\_1463831\_customer\_customer\_feedback]

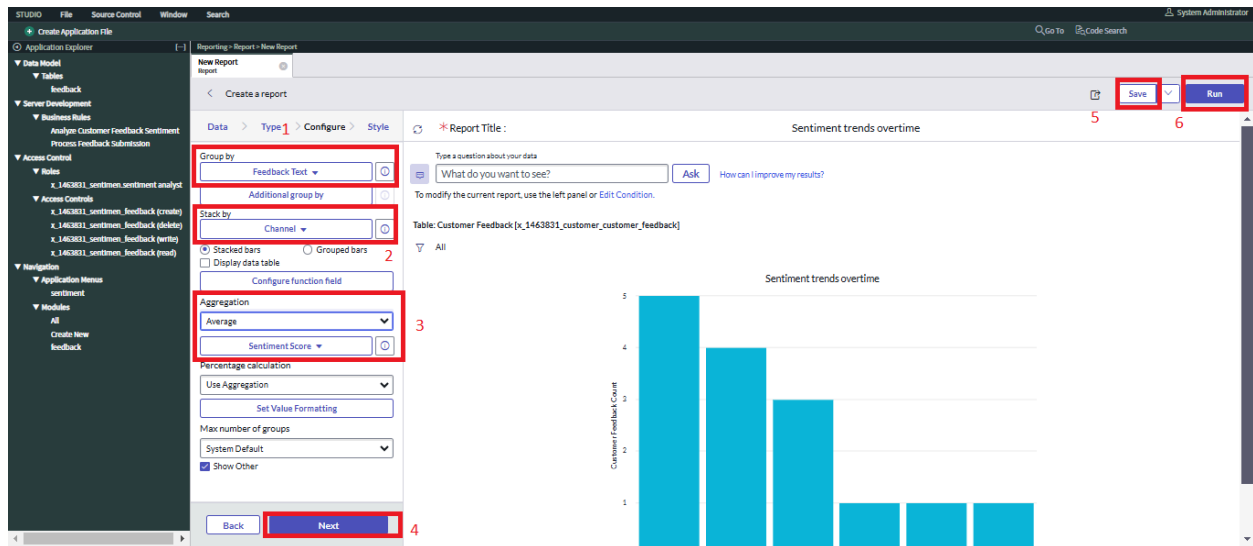
▽ All

**Sentiment trends overtime**

Customer Feedback Count

Category	Count
1	5
2	4
3	3
4	1
5	1
6	1

[Back](#) [Next](#)

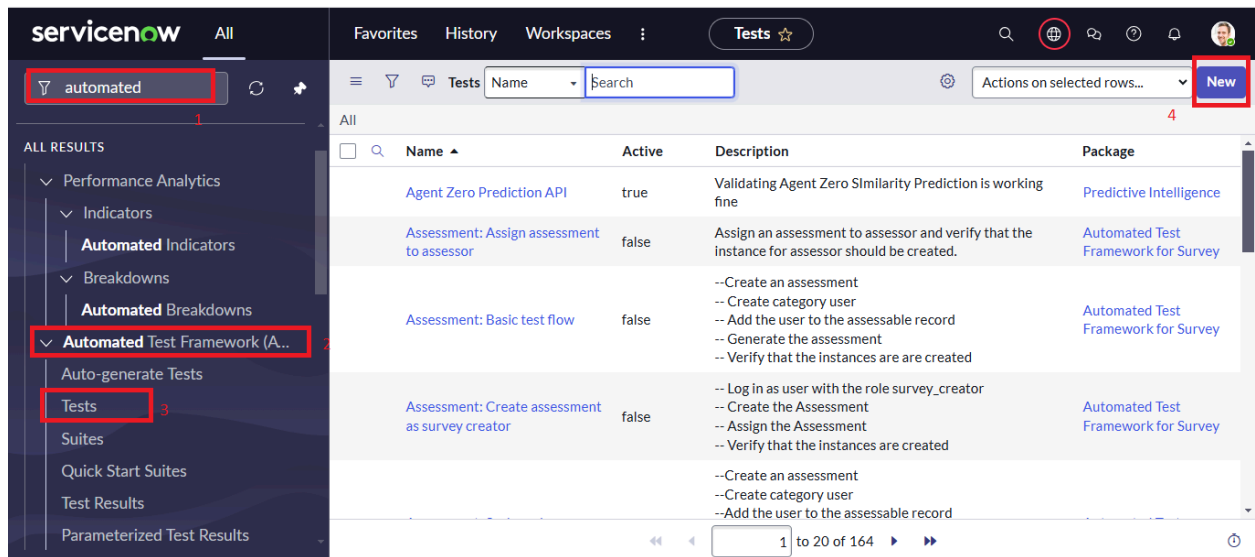


## Milestone 6 : Automated tested framework

### Activity : Tests

- 1) Go to application navigator and select tests under automated tested frameworks and click on new
- 2) Fill the details ; Name : Submit feedback via form verify insertion into customer feedback table
- 3) Click on save
- 4) Under the test steps create four tests
  - a) Open a new form ; select table 'customer feedback' and save
  - b) Set Field Values ; select table 'customer feedback', conditions as 'channel : email' 'feedback text : great service!' and save
  - c) click a UI action ; select table 'customer feedback' , UI action select 'submit' and save.
  - d) Record Validation ; select table 'customer feedback' ,under record select any record and conditions 'channel is email' 'Feedback text is bad' and click on save.

- 5) Click on run test
- 6) Under the test results section check the result.



**servicenow** All

automated 1

ALL RESULTS

- Performance Analytics
  - Indicators
    - Automated Indicators
  - Breakdowns
    - Automated Breakdowns
  - Automated Test Framework (A...
    - Auto-generate Tests
    - Tests 3
    - Suites
    - Quick Start Suites
    - Test Results
    - Parameterized Test Results

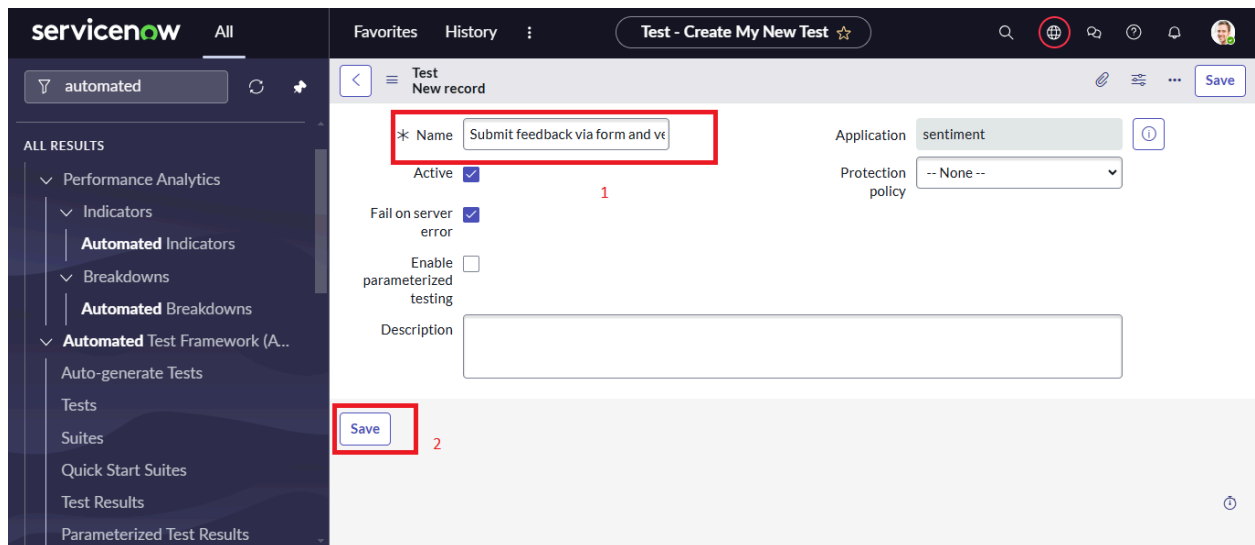
Tests

Search

Actions on selected rows... New 4

Name	Active	Description	Package
Agent Zero Prediction API	true	Validating Agent Zero Similarity Prediction is working fine	Predictive Intelligence
Assessment: Assign assessment to assessor	false	Assign an assessment to assessor and verify that the instance for assessor should be created.	Automated Test Framework for Survey
Assessment: Basic test flow	false	--Create an assessment -- Create category user -- Add the user to the assessable record -- Generate the assessment -- Verify that the instances are created	Automated Test Framework for Survey
Assessment: Create assessment as survey creator	false	-- Log in as user with the role survey_creator -- Create the Assessment -- Assign the Assessment -- Verify that the instances are created	Automated Test Framework for Survey

1 to 20 of 164



**servicenow** All

automated

ALL RESULTS

- Performance Analytics
  - Indicators
    - Automated Indicators
  - Breakdowns
    - Automated Breakdowns
  - Automated Test Framework (A...
    - Auto-generate Tests
    - Tests
    - Suites
    - Quick Start Suites
    - Test Results
    - Parameterized Test Results

Test - Create My New Test

Test New record

\* Name Submit feedback via form and ve 1

Active ☒

Fail on server error ☒

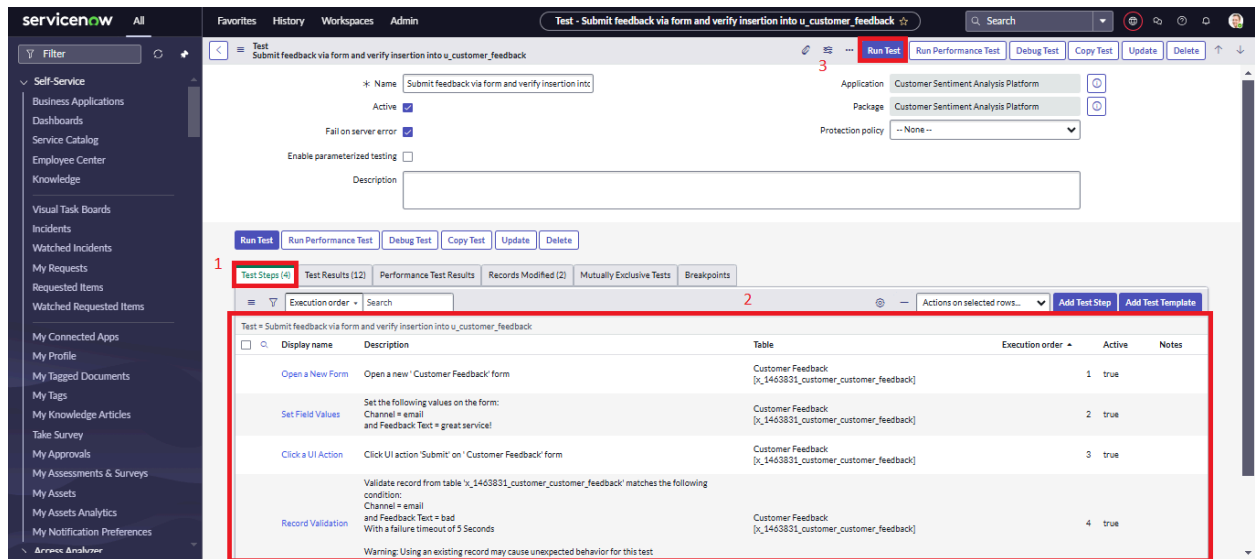
Enable parameterized testing ☐

Description

Application sentiment

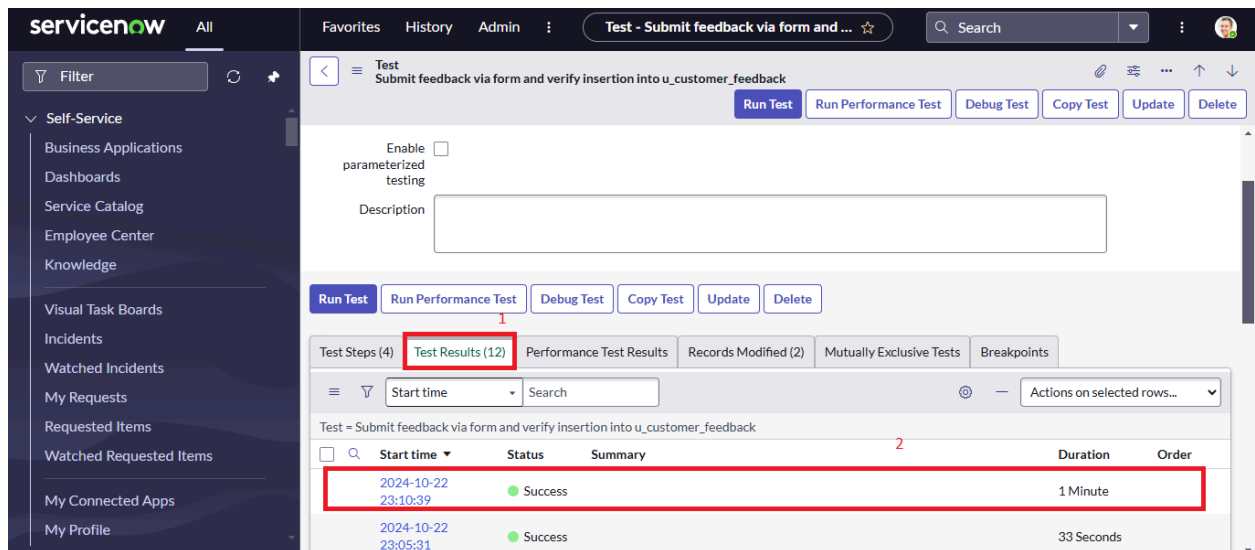
Protection policy -- None --

Save 2



The screenshot shows the ServiceNow Test Suite configuration page for a test named "Submit feedback via form and verify insertion into u\_customer\_feedback". The test is active and has a description field. The "Test Steps" tab is selected, showing a list of steps with their display names, descriptions, tables, execution orders, and active status.

Display name	Description	Table	Execution order	Active	Notes
Open a New Form	Open a new 'Customer Feedback' form	Customer Feedback [x_1463831_customer_customer_feedback]	1	true	
Set Field Values	Set the following values on the form: Channel = email and Feedback Text = great service!	Customer Feedback [x_1463831_customer_customer_feedback]	2	true	
Click a UI Action	Click UI action 'Submit' on 'Customer Feedback' form	Customer Feedback [x_1463831_customer_customer_feedback]	3	true	
Record Validation	Validate record from table 'x_1463831_customer_customer_feedback' matches the following condition: Channel = email and Feedback Text = bad With a failure timeout of 5 Seconds	Customer Feedback [x_1463831_customer_customer_feedback]	4	true	



The screenshot shows the ServiceNow Test Results page for the same test. The "Test Results" tab is selected, showing a list of test results with their start times, statuses, summaries, durations, and orders.

Start time	Status	Summary	Duration	Order
2024-10-22 23:10:39	Success		1 Minute	
2024-10-22 23:05:31	Success		33 Seconds	

## Activity 2 : Create second test

- 1) In application navigator open tests under Automated tested frameworks
- 2) Click on new



- 3) Fill the details ; Name : Check the sentiment analysis is performed after feedback submission
- 4) Click on save
- 5) Under the test steps create four tests

a) Open a new form ; select table ' customer feedback' and save

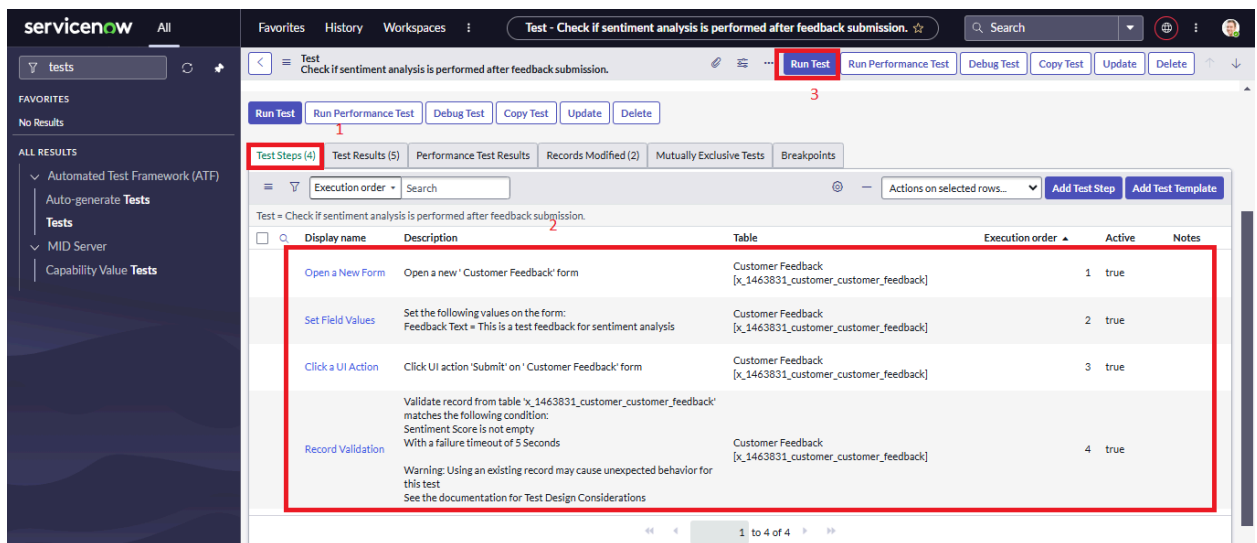
b) Set Field Values ; select table 'customer feedback', conditions as feedback text :

This the test feedback for sentiment analysis and save.

c) Click a UI action ; select table 'customer feedback' , UI action select 'submit' and save.

d) Record Validation ; select table 'customer feedback' ,under record select any record and condition is 'sentiment score is not empty' and click on save.

- 6) Click on run test
- 7) Under the test results section check the result.



The screenshot shows the ServiceNow Test Framework interface. The test name is "Check if sentiment analysis is performed after feedback submission." The test steps are listed in a table, and the "Run Test" button is highlighted. The test steps are:

Display name	Description	Table	Execution order	Active	Notes
Open a New Form	Open a new ' Customer Feedback' form	Customer Feedback [x_1463831_customer_customer_feedback]	1	true	
Set Field Values	Set the following values on the form: Feedback Text = This is a test feedback for sentiment analysis	Customer Feedback [x_1463831_customer_customer_feedback]	2	true	
Click a UI Action	Click UI action 'Submit' on ' Customer Feedback' form	Customer Feedback [x_1463831_customer_customer_feedback]	3	true	
Record Validation	Validate record from table 'x_1463831_customer_customer_feedback' matches the following condition: Sentiment Score is not empty With a failure timeout of 5 Seconds Warning: Using an existing record may cause unexpected behavior for this test See the documentation for Test Design Considerations	Customer Feedback [x_1463831_customer_customer_feedback]	4	true	

**servicenow** All

Favorites History Workspaces **Test - Create My New Test** Search

test

FAVORITES  
No Results

ALL RESULTS

- NLU Workbench
- NLU Advanced Features
  - Multi-model Batch **Testing**
- Automated **Test** Framework (A...)
  - Auto-generate **Tests**
  - Tests**
  - Suites
  - Quick Start Suites
  - Test** Results
  - Parameterized **Test** Results
  - Suite Results
  - Schedules

**Test**  
New record

\* Name **Check if sentiment analysis is perform** Application sentiment

Active ☒ 1 Protection policy -- None --

Fail on server error ☒

Enable parameterized testing ☐

Description

**Save** 2

**servicenow** All

Favorites History Workspaces **Test - Check if sentiment analysis is performed after feedback submission.** Search

tests

FAVORITES  
No Results

ALL RESULTS

- Automated Test Framework (ATF)
  - Auto-generate **Tests**
- MID Server
  - Capability Value **Tests**

**Test**  
Check if sentiment analysis is performed after feedback submission.

Application Customer Sentiment Analysis Platform

Package Customer Sentiment Analysis Platform

Protection policy -- None --

Run Test Run Performance Test Debug Test Copy Test Update Delete

Test Steps (4) **Test Results (5)** Performance Test Results Records Modified (2) Mutually Exclusive Tests Breakpoints

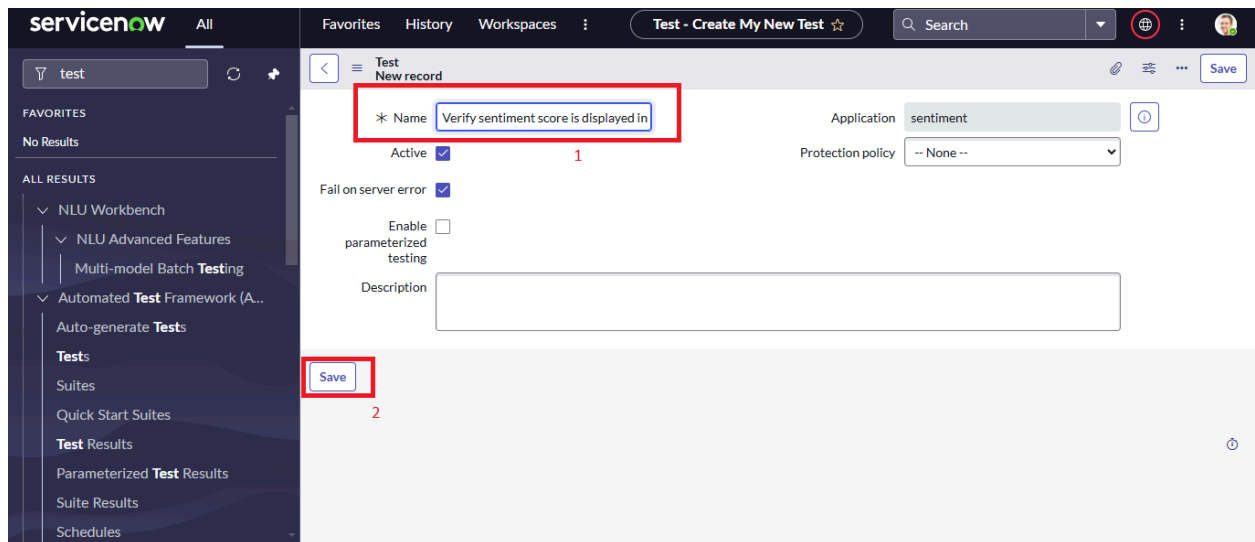
Start time Search

Test = Check if sentiment analysis is performed after feedback submission.

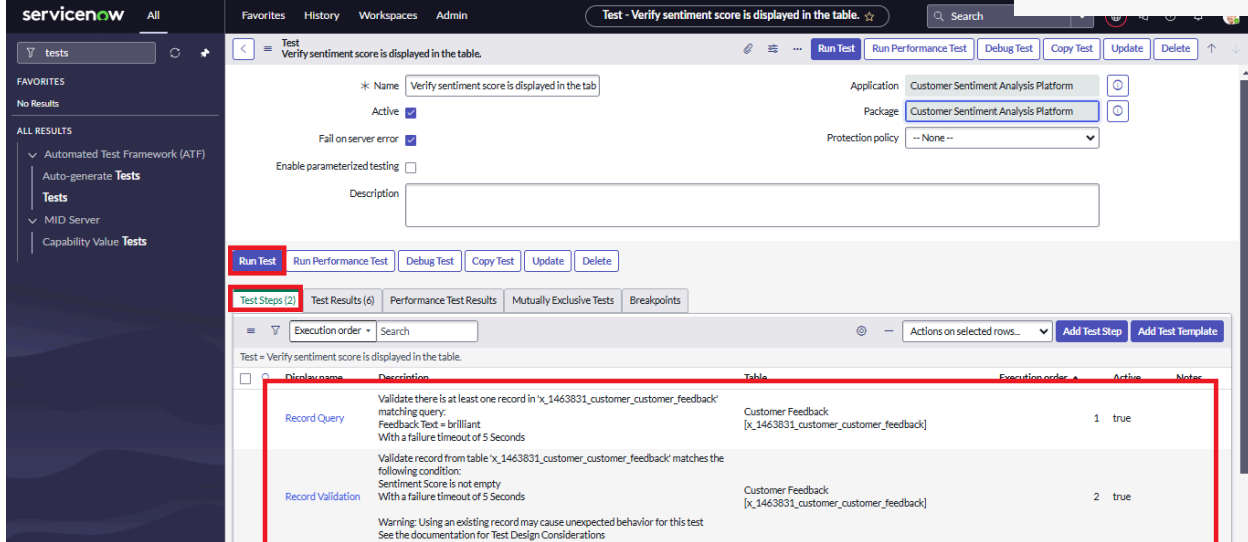
Start time	Status	Summary	Duration	Order
2024-10-22 23:07:14	Success		31 Seconds	
2024-10-18 03:10:20	Success		30 Seconds	
2024-10-18 03:07:24	Success		27 Seconds	
2024-10-18 03:05:29	Success		14 Seconds	
2024-10-18 03:03:47	Success		11 Seconds	

### Activity 3 : Create third test

- 1) Go to application navigator and select tests under automated tested frameworks and click on new
- 2) Fill the details ; Name : Verify sentiment score is displayed in the table.
- 3) Click on save
- 4) Under the test steps
  - a) Record Query ; select table 'customer feedback' and condition as 'feedback text is brilliant' and click on save.
  - b) Record Validation ; select table 'customer feedback, and condition as 'sentiment score is not empty' and click on save.
- 5) Click on run test
- 6) Under the test result check results



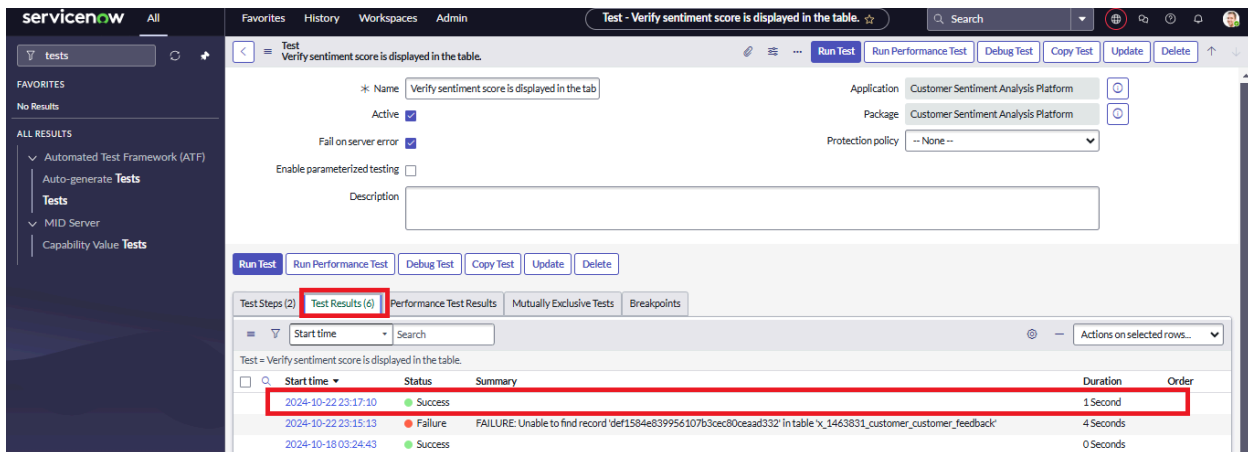
ServiceNow interface showing the 'Test - Create My New Test' form. The form includes fields for Name, Application, Active, Fall on server error, Enable parameterized testing, and Description. The 'Name' field is highlighted with a red box and contains the text 'Verify sentiment score is displayed in'. The 'Application' field is set to 'sentiment'. The 'Active' checkbox is checked. The 'Fall on server error' checkbox is checked. The 'Enable parameterized testing' checkbox is unchecked. The 'Description' field is empty. The 'Save' button is highlighted with a red box. The left sidebar shows the 'test' search bar and a list of results including 'NLU Workbench', 'NLU Advanced Features', 'Multi-model Batch Testing', 'Automated Test Framework (A...', 'Auto-generate Tests', 'Tests', 'Suites', 'Quick Start Suites', 'Test Results', 'Parameterized Test Results', 'Suite Results', and 'Schedules'.



The screenshot shows the ServiceNow Test Framework interface. The test name is "Verify sentiment score is displayed in the table". The application is "Customer Sentiment Analysis Platform" and the package is "Customer Sentiment Analysis Platform". The test is active and has a failure timeout of 5 seconds. The test steps are:

Step	Description	Table	Execution order	Action	Note
Record Query	Validate there is at least one record in 'x_1463831_customer_customer_feedback' matching query: Feedback Text = brilliant With a failure timeout of 5 Seconds	Customer Feedback [x_1463831_customer_customer_feedback]	1	true	
Record Validation	Validate record from table 'x_1463831_customer_customer_feedback' matches the following condition: Sentiment Score is not empty With a failure timeout of 5 Seconds	Customer Feedback [x_1463831_customer_customer_feedback]	2	true	

Warning: Using an existing record may cause unexpected behavior for this test. See the documentation for Test Design Considerations.



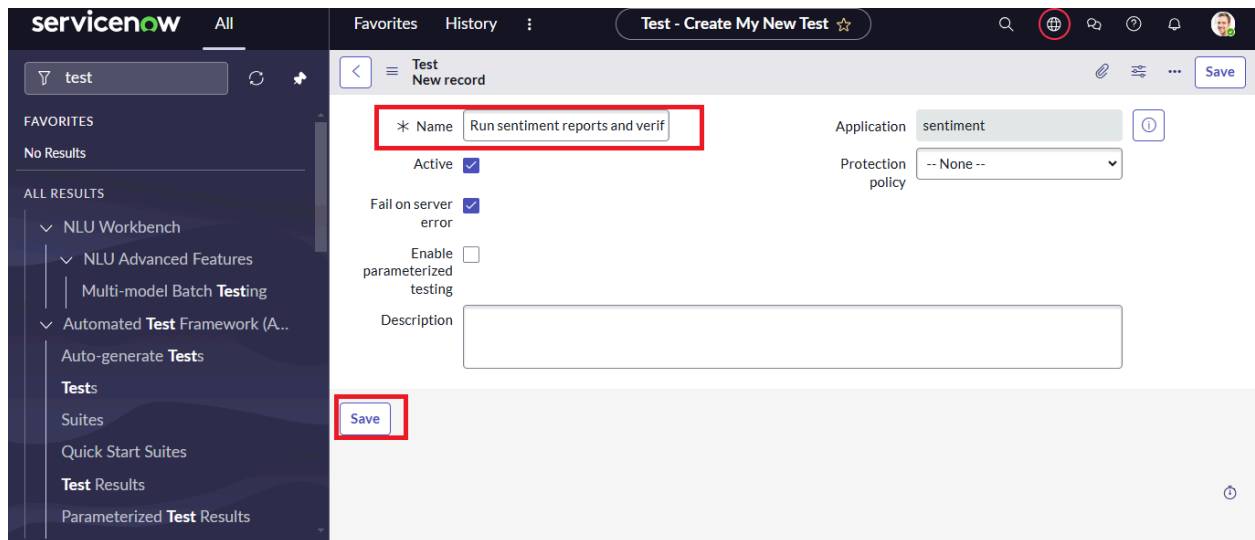
The screenshot shows the ServiceNow Test Framework interface with the test results tab selected. The test results are as follows:

Start time	Status	Summary	Duration	Order
2024-10-22 23:17:10	Success		1 Second	
2024-10-22 23:15:13	Failure	FAILURE: Unable to find record 'def1584e839956107b3cec80ccad332' in table 'x_1463831_customer_customer_feedback'	4 Seconds	
2024-10-18 03:24:43	Success		0 Seconds	

## Activity 4 : Create fourth Test

- 1) Go to application navigator and select tests under automated tested frameworks and click on new
- 2) Fill the details ; Name : Run sentiment reports and verify and
- 3) Click on save
- 4) Under test steps create steps
  - a) Report visibility ; select report 'sentiment score over time' and assert type as 'can view report' click on save

- 5) Click save and run
- 6) Under the test results check the result



servicenow All

Favorites History Test - Create My New Test

test

FAVORITES  
No Results

ALL RESULTS

- ✓ NLU Workbench
- ✓ NLU Advanced Features
  - Multi-model Batch **Testing**
- ✓ Automated **Test** Framework (A...
  - Auto-generate **Tests**
  - Tests**
  - Suites
  - Quick Start Suites
  - Test** Results
  - Parameterized **Test** Results

Test  
New record

\* Name Run sentiment reports and verif

Application sentiment

Active ☒

Fail on server error ☒

Enable parameterized testing ☐

Protection policy -- None --

Description

Save

**servicenow** All

Test - Run sentiment reports and verify correct data is pulled.

Test: Run sentiment reports and verify correct data is pulled.

Name: Run sentiment reports and verify correct d

Application: Customer Sentiment Analysis Platform

Active: ☒

Package: Customer Sentiment Analysis Platform

Fail on server error: ☒

Protection policy: -- None --

Enable parameterized testing: ☐

Description:

Run Test Debug Test Copy Test Update Delete

Test Steps (1) Test Results (2) Performance Test Results Mutually Exclusive Tests Breakpoints

Execution order Search Actions on selected rows... Add Test Step Add Test Template

Test = Run sentiment reports and verify correct data is pulled.

Display name	Description	Table	Execution order	Active	Notes
Report Visibility	Confirm report 'Sentiment Trends Over Time,' can be viewed by the test user		1	true	

1 to 1 of 1

**servicenow** All

Test - Run sentiment reports and verify correct data is pulled.

Test: Run sentiment reports and verify correct data is pulled.

Name: Run sentiment reports and verify correct d

Application: Customer Sentiment Analysis Platform

Active: ☒

Package: Customer Sentiment Analysis Platform

Fail on server error: ☒

Protection policy: -- None --

Enable parameterized testing: ☐

Description:

Run Test Debug Test Copy Test Update Delete

Test Steps (1) Test Results (2) Performance Test Results Mutually Exclusive Tests Breakpoints

Start time Search Actions on selected rows...

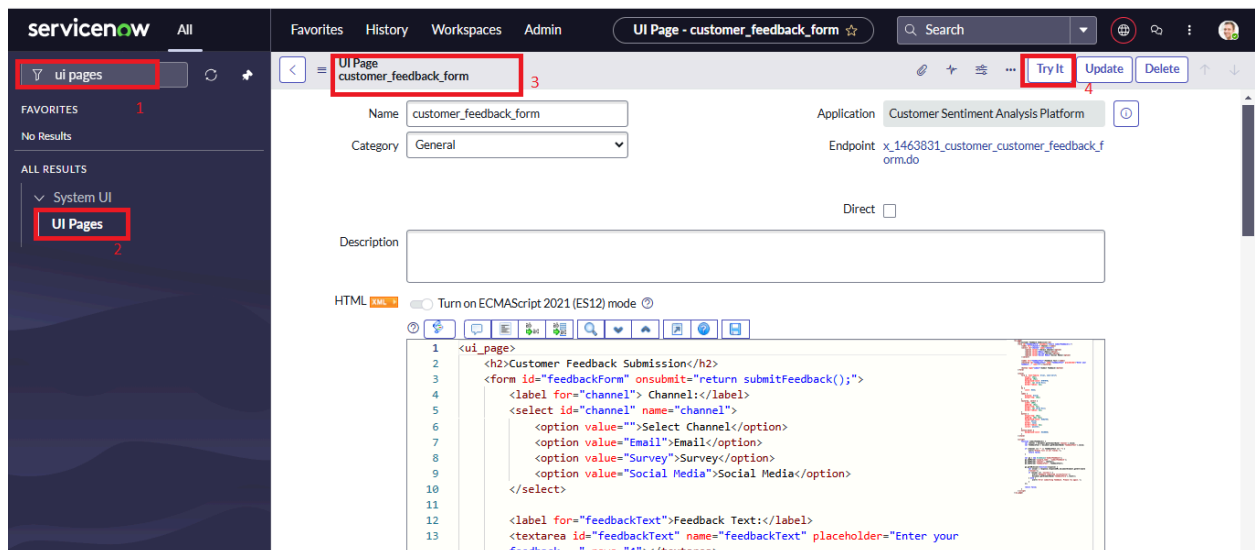
Test = Run sentiment reports and verify correct data is pulled.

Start time	Status	Summary	Duration	Order
2024-10-22 23:18:22	Success		0 Seconds	
2024-10-18 03:30:33	Success		1 Second	

## Milestone 7 : Manual testing

### Activity 1 : Test Ui page form

- 1) In the application navigator open UI pages
- 2) Open 'customer feedback form'
- 3) Click on try it
- 4) On the form Select Channel as 'survey' and feedback text as 'excellent service!'
- 5) Click on submit
- 6) In the application navigator open 'customer feedback' table
- 7) See the record that was created through feedback form



The screenshot shows the ServiceNow UI Page editor interface. On the left sidebar, under 'System UI', the 'UI Pages' link is highlighted with a red box and labeled '2'. The main area displays the configuration for the 'UI Page - customer\_feedback\_form', which is also highlighted with a red box and labeled '3'. The 'Try It' button is highlighted with a red box and labeled '4'. The configuration includes the name 'customer\_feedback\_form', category 'General', application 'Customer Sentiment Analysis Platform', and endpoint 'x\_1463831\_customer\_customer\_feedback\_form.do'. The HTML editor at the bottom shows the following code:

```
1 <ui_page>
2 <h2>Customer Feedback Submission</h2>
3 <form id="feedbackForm" onsubmit="return submitFeedback();">
4 <label for="channel"> Channel:</label>
5 <select id="channel" name="channel">
6 <option value="">Select Channel</option>
7 <option value="Email">Email</option>
8 <option value="Survey">Survey</option>
9 <option value="Social Media">Social Media</option>
10 </select>
11
12 <label for="feedbackText">Feedback Text:</label>
13 <textarea id="feedbackText" name="feedbackText" placeholder="Enter your feedback..." rows="4"></textarea>
```

customer

1

No Results

ALL RESULTS

Customer Sentiment Analysis PL...

Customer Feedback

Create New

All

Sentiment Reports

Create New

All

System Security

Security Center

Customer Actions

Favorites

History

Workspaces

Admin

Customer Feedbacks

Search

Actions on selected rows...

New

Channel

Search

All

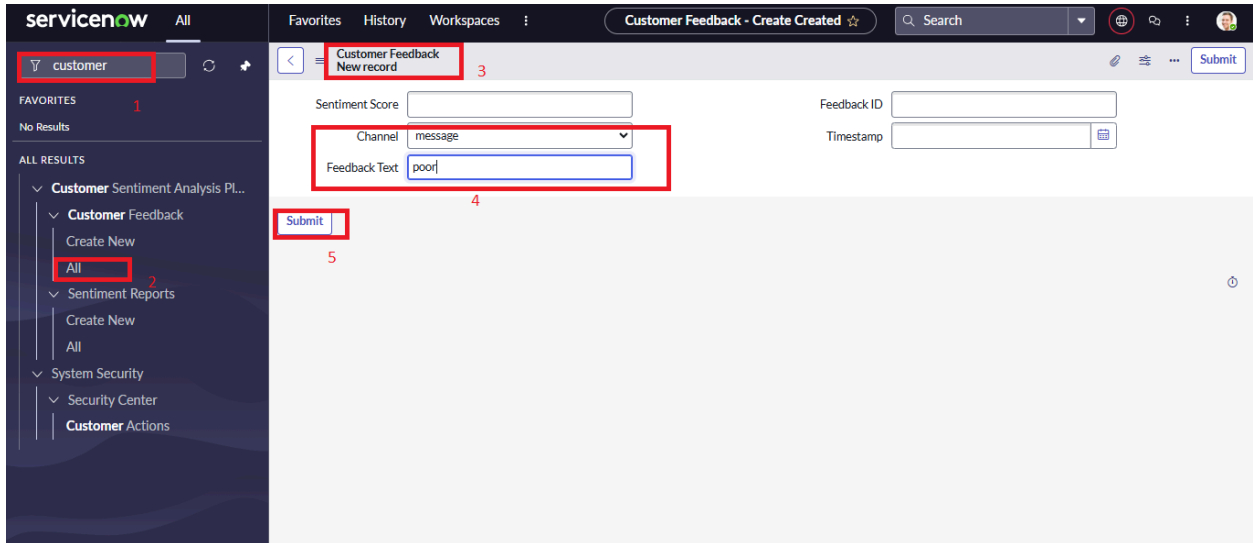
<input type="checkbox"/>	Channel	Feedback ID	Feedback Text	Sentiment Score	Timestamp
<input type="checkbox"/>	message	123	poor	-1	2024-10-30 22:22:01
<input type="checkbox"/>	message		bad	-1 (empty)	
<input type="checkbox"/>	message		brilliant	0 (empty)	
<input type="checkbox"/>	Social Media		bad	(empty)	
<input type="checkbox"/>	string	540	good service!	(empty)	
<input type="checkbox"/>	string		okok	70 (empty)	
<input type="checkbox"/>	string		great service!	(empty)	
<input type="checkbox"/>	string	345	good service!	30 (empty)	
<input type="checkbox"/>	Survey		brilliant	(empty)	
<input type="checkbox"/>	Survey		The product is excellent, and the suppor	(empty)	
<input type="checkbox"/>	Survey		good	1 (empty)	
<input type="checkbox"/>	Survey		gd	3 (empty)	
<input checked="" type="checkbox"/>	Survey		Excellent Service!	0 (empty)	
<input type="checkbox"/>	Survey		The service was slow, and the product qu	(empty)	
<input type="checkbox"/>	Test Channel		This is a test feedback.	(empty)	

1 to 16 of 16



## Activity 2 : Sentiment Score update

- 1) In the application navigator , open 'customer feedback table'
- 2) Click on new
- 3) Fill the details ; channel as 'message', feedback text as 'poor'
- 4) Click on submit
- 5) Now , in the record that we have created we can see the sentiment score updated to '-1'.



The screenshot shows the ServiceNow 'Customer Feedback - Create' form. The interface includes a left-hand navigation pane, a top header bar, and a main form area. Red boxes and numbers 1 through 5 highlight specific elements:

- 1**: Points to the 'customer' filter in the left-hand navigation pane.
- 2**: Points to the 'All' link under the 'Customer Feedback' section in the left-hand navigation pane.
- 3**: Points to the 'Customer Feedback New record' link in the top header bar.
- 4**: Points to the 'Submit' button in the main form area.
- 5**: Points to the 'Submit' button in the left-hand navigation pane.

The form fields visible include:

- Sentiment Score (text input)
- Feedback ID (text input)
- Channel (dropdown menu, currently set to 'message')
- Feedback Text (text input, currently containing 'poor')
- Timestamp (text input)

servicenow All

customer

FAVORITES

No Results

ALL RESULTS

Customer Sentiment Analysis PL...

Customer Feedback

Create New

All

Sentiment Reports

Create New

All

System Security

Security Center

Customer Actions

Customer Feedbacks

Channel

Search

Actions on selected rows...

New

Channel	Feedback ID	Feedback Text	Sentiment Score	Timestamp
email		bad	-1	(empty)
message	123	poor	-1	2024-10-30 22:22:01
message		bad	-1	(empty)
message		brilliant	0	(empty)
message		poor	-1	(empty)
Social Media		bad	(empty)	
string		okok	70	(empty)
string		great service!	(empty)	
string	540	good service!	(empty)	
string	345	good service!	30	(empty)
Survey		Excellent Service!	0	(empty)
Survey		brilliant	(empty)	
Survey		The product is excellent, and the suppor	(empty)	
Survey		good	1	(empty)
Survey		gd	(empty)	

1 to 17 of 17

### Activity 3 : Check Email

- 1) Go to application navigator ,open Emails under system logs
- 2) Now , can see the Email which is sent

servicenow All

system logs

FAVORITES

No Results

ALL RESULTS

System Logs

Client Interactions

Transactions

Transactions (All user)

Transactions (Background)

Client Transactions

Emails

Push Notifications

Events

Imports

Emails

Subject

Search

Actions on selected rows...

New

Created	Recipients	Subject	Type	User ID
2024-10-25 02:40:24	problem.admin@example.com,problem.coordi...	Incident INC0010188 has been assigned to group Problem Solving	send-ready	(empty)
2024-10-25 02:40:24	alejandra.prenatt@example.com,aileen.mot...	New Negative Feedback Received	send-ready	(empty)
2024-10-25 01:00:04	aileen.mottern@example.com	Restocking Request For APC 42U 3100 SP2 NetShelter Rack	send-ready	(empty)
2024-10-25 01:00:04	aileen.mottern@example.com	Restocking Request For Dell Inc. PowerEdge M710HD Blade Server	send-ready	(empty)
2024-10-25 01:00:04	aileen.mottern@example.com	Restocking Request For Fujitsu 1TB Hybrid Solid State Drive	send-ready	(empty)

1 to 5 of 5

### Conclusion



The Customer Sentiment Analysis Platform provides a comprehensive solution for collecting, analyzing, and responding to customer feedback across multiple channels. By leveraging AI for sentiment analysis and automated workflows, this platform enables real-time identification of trends in customer satisfaction, early detection of negative feedback, and prompt responses to improve the overall customer experience. The platform's structured approach to data collection and sentiment scoring facilitates actionable insights that empower customer service teams to address concerns proactively, resulting in improved customer satisfaction and loyalty.

This project demonstrates the power of ServiceNow's capabilities in integrating automation, AI, and reporting tools to create a streamlined feedback management system, ultimately enhancing business responsiveness and driving better customer outcomes.

