Laptop Request System Implementation

Team Id: NM2025TMID18159

Team Members:

Team Leader : Dhanalakshmi S

Team Member 1:Shemida paul Y

Team Member 2: Vaisali S

Team Member 3:Banu priya M

Problem Statement

Employees in the organization need a quick and efficient way to request laptops for work. The current process is manual and prone to delays, with no dynamic form behavior to guide users or ensure accurate data collection. To address this, a Service Catalog item needs to be created, allowing users to easily request a laptop, with dynamic fields, clear instructions, and additional functionality like resetting the form if needed. The solution should also ensure all changes are tracked for governance and deployment.

Objective

Design and implement a ServiceNow catalog item that enables employees to submit laptop requests dynamically with form validations, clear guidance, and

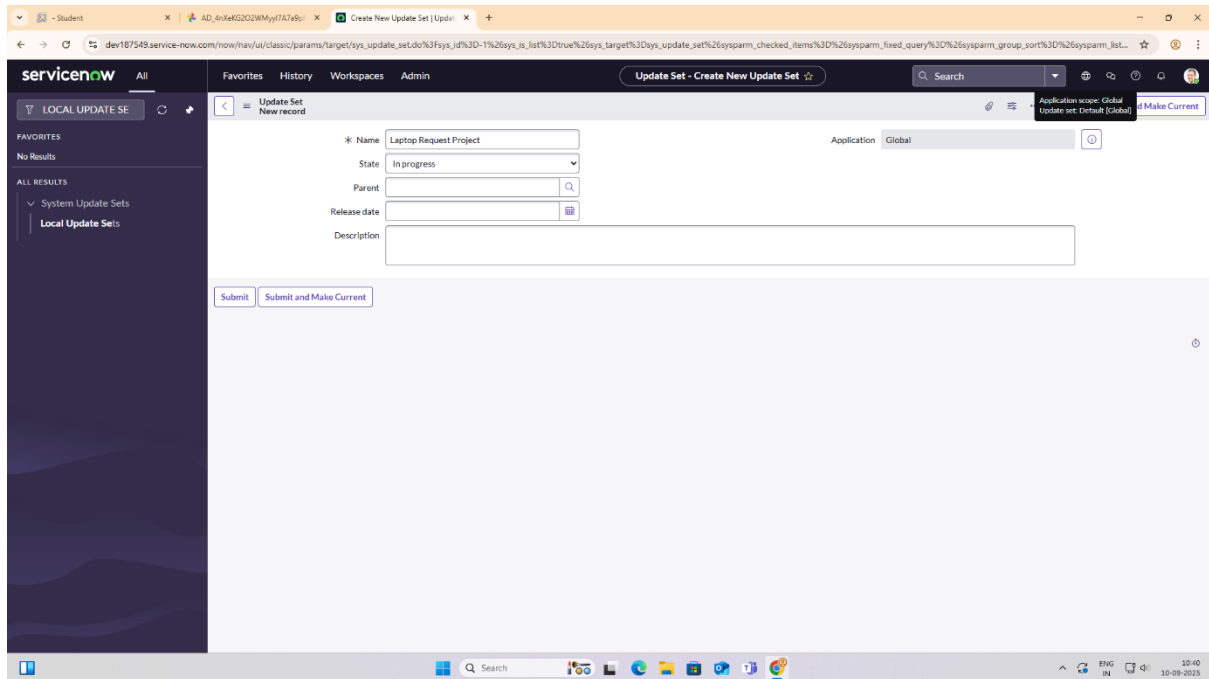governance controls, improving speed, accuracy, and accountability in the process.

Skills

UIPath RPA,Tanzu Application Service

TASK INITIATION

Milestone 1: Create Local Update Set

Activity 1: Create Update Set in ServiceNow

1. Open ServiceNow instance.

2. Click on All → search for Update Sets.

3. Select Local Update Sets under System Update Sets.

4. Click on New.

5. Fill in the following details:

Name: Laptop Request

6. Click Submit and make it Current.

7.By clicking on the button it activates the update set

Milestone 2.1: Create Service Catalog Item

Activity 1: Create New Service Catalog Item in ServiceNow

1. Open service now.

2. Click on All >> service catalog

3. Select maintain items under catalog definitions

4. Click on New.

5.Fill the following details to create a new catalog item

> Name: Laptop Request

> Catalog: service Catalog

> Category: Hardware

> Short Description: Use this item to request a new laptop

6.Click on 'SAVE'
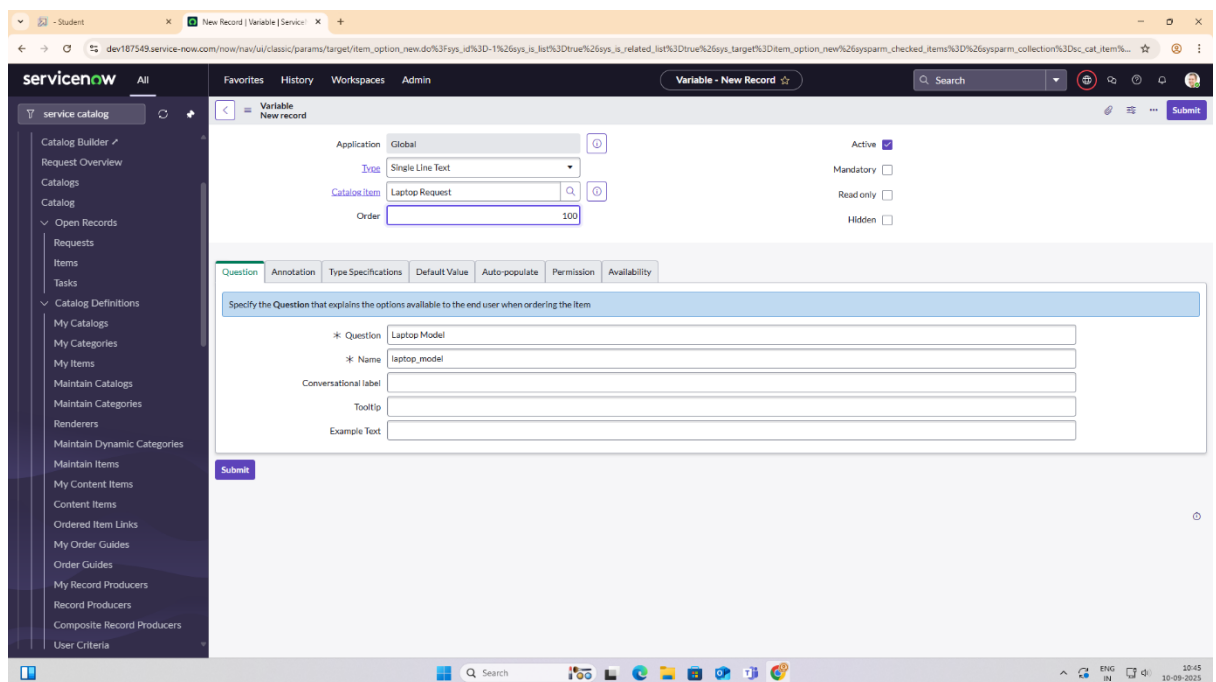
Activity 2: Add Variables to the Catalog Item

Step1:

- After saving the catalog item form scroll down and click on variable(related list)

- Click on new and enter the details as below

  1. Variable 1:Laptop Model

     Type: Single line text

     Name: laptop_model

     Order:100

- Click on submit

- Again click on new and add Remaining variables in the above process

  2. Variable 2:Justification

     Type: Multi line text

     Name: justification

     Order:200

  3. Variable 3:Additional Accessories

Type: Checkbox

Name: additional_accessories

Order:300

4. Variable 4: Accessories Details
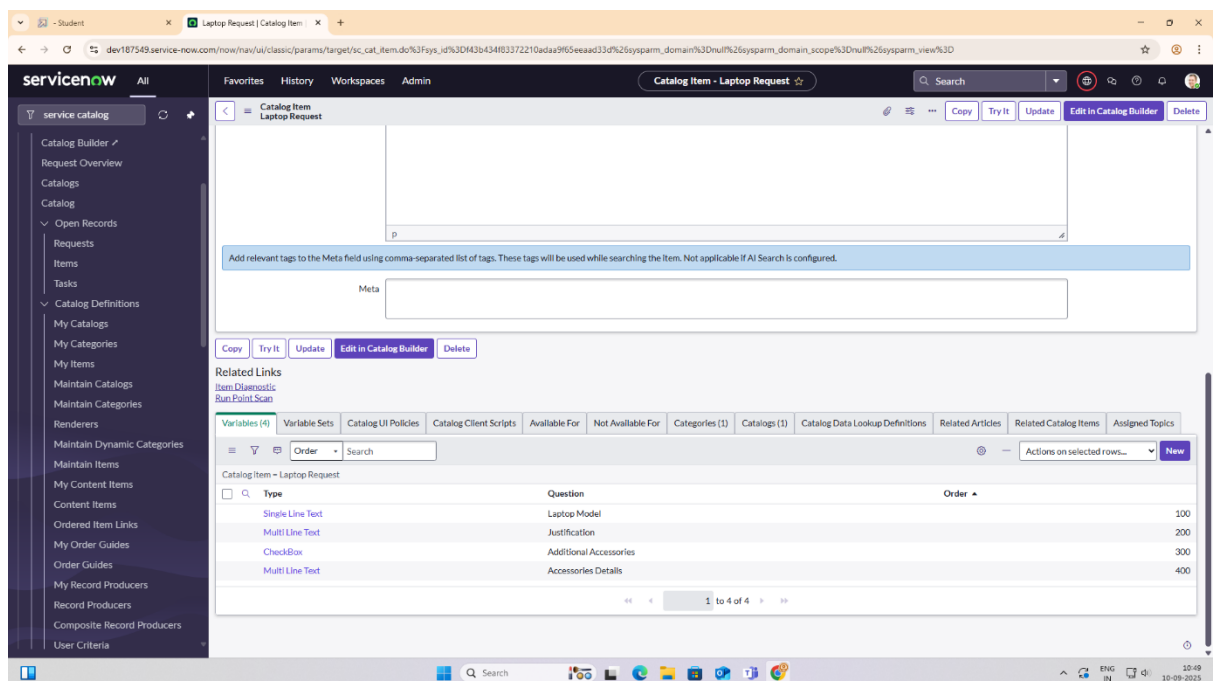
Type: Multi line text

Name:accessories_details

Order:400

Step2:

- After adding above variable which are added to newly created catalog item
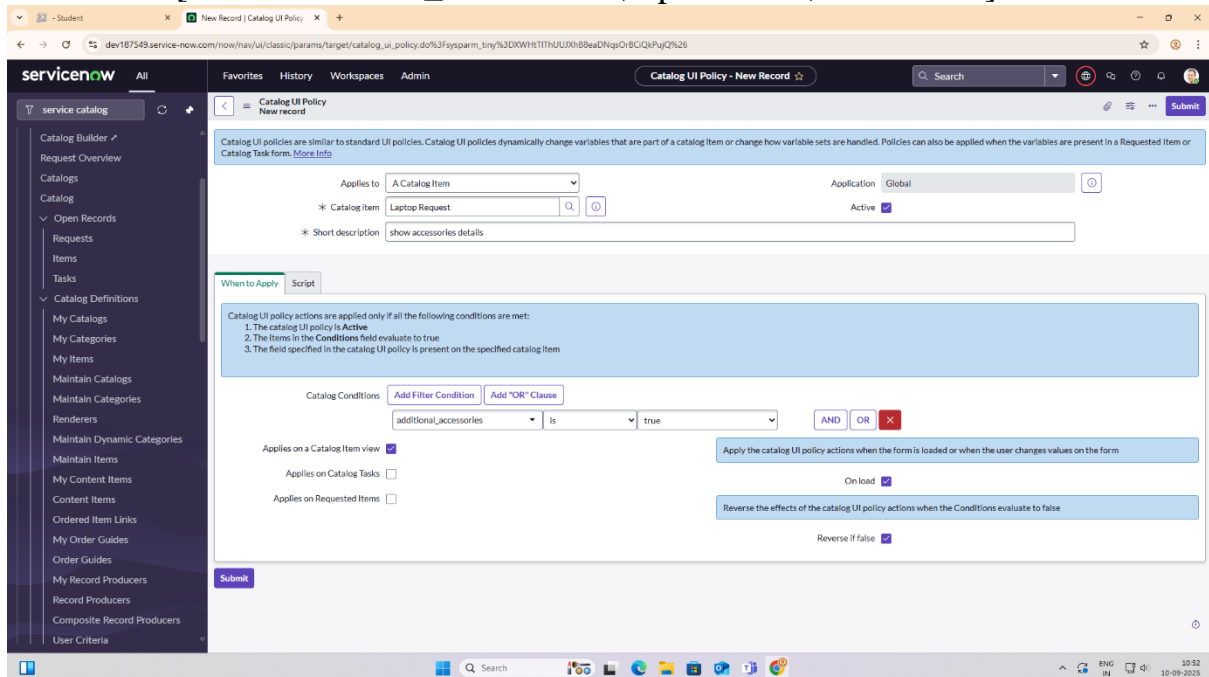
- Then save the catalog item form

Milestone 3: Create Catalog UI Policies

Activity 1: Configure Catalog UI Policy for Dynamic Behavior

1. Click on all>> search for service catalog
2. Select maintain item under catalog definition

3. Search for 'laptop request' which is created before
4. Select 'laptop request' and scroll down click on "Catalog Ui policies"
5. In the catalog ui policies related list tab click on new
6. Give short description as: show accessories details
7. Set the Catalog Condition in the related list tab 'when to apply'
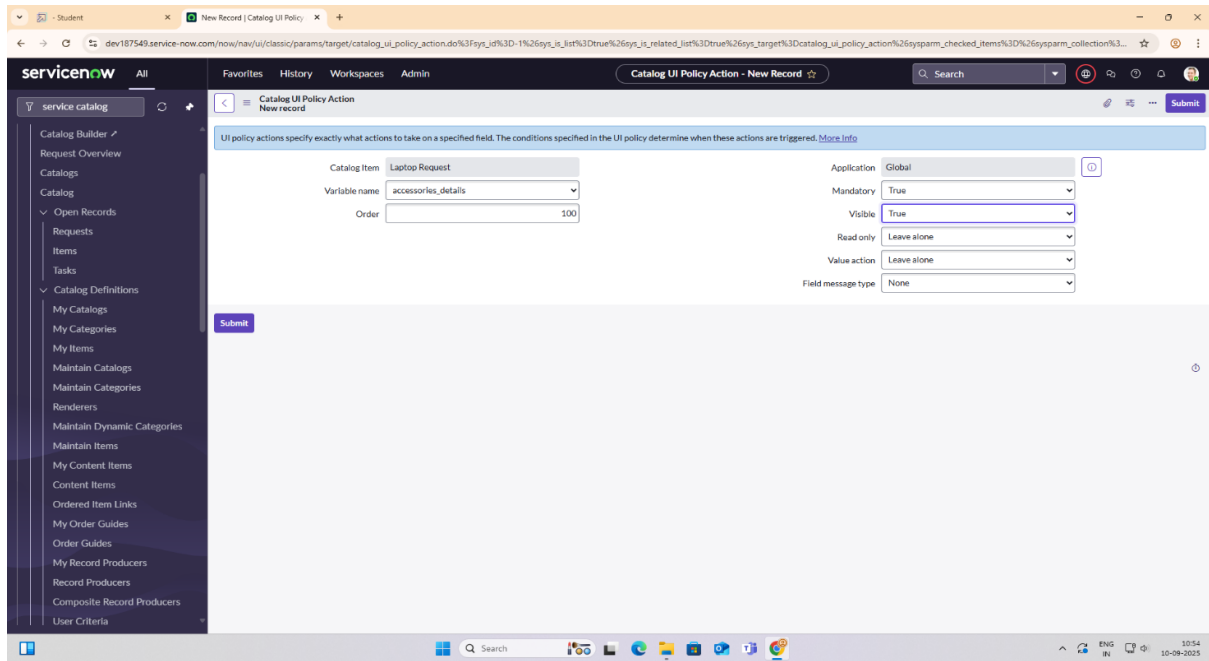   [field: additional_ accessories, operator: is, value: true]

8. Click on **save**.(do not click on submit)
9. Scroll down and select 'catalog ui action'
10. Then click on new button
11. Select variable name as: accessories_details
    Order:100
    Mandatory: True
    Visible : True
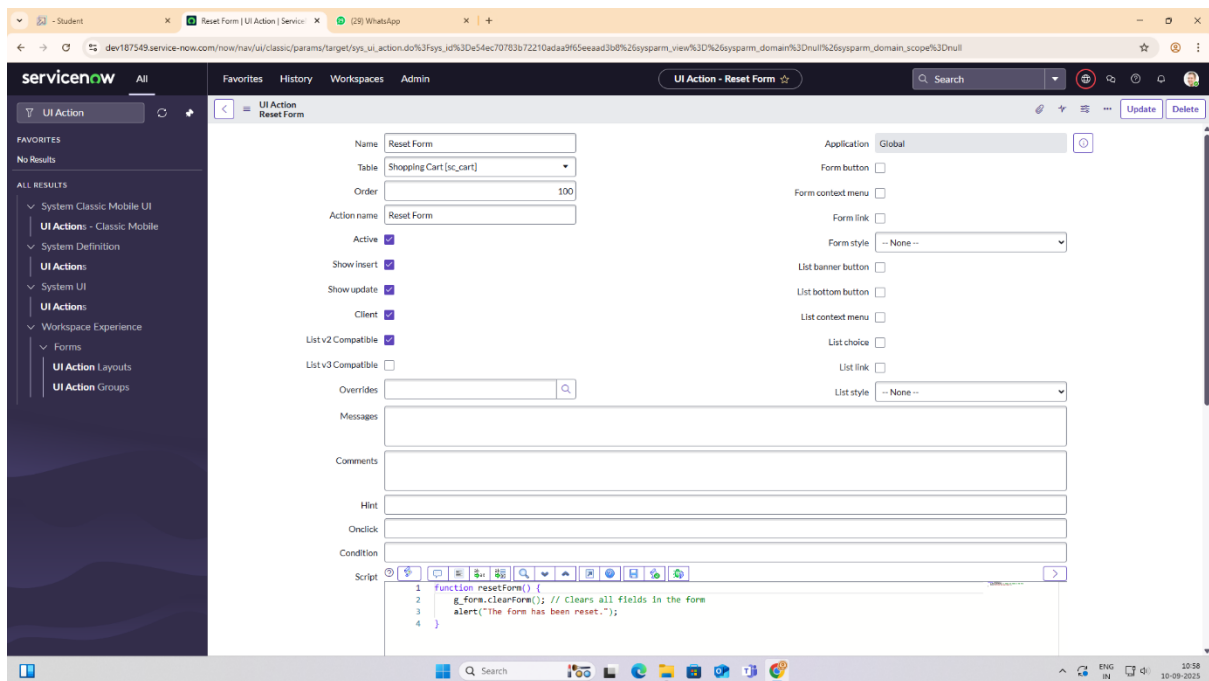
12.   Click on save and again click save button of the catalog ui policy form

Milestone 4: Create UI Action

Activity 1: Create Reset Form UI Action

1. Open service now.
2. Click on All >> search for ui action

3. Select ui actions under system definition
4. Click on new
5. Fill the following details to create ui action
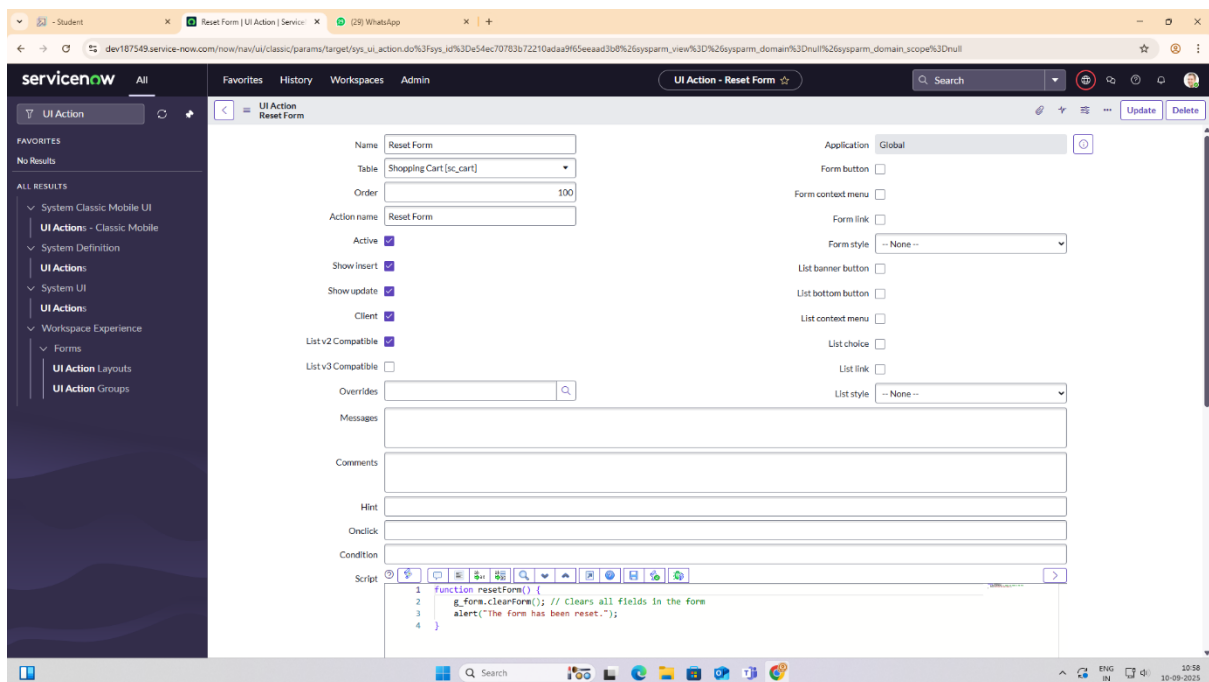
    Table: shopping cart(sc_cart)

    Order:100

    Action name: Reset form

     Client : checked

Script:

    function resetForm() {

    g_form.clearForm(); // Clears all fields in the form

    alert("The form has been reset.");

}

Click on save
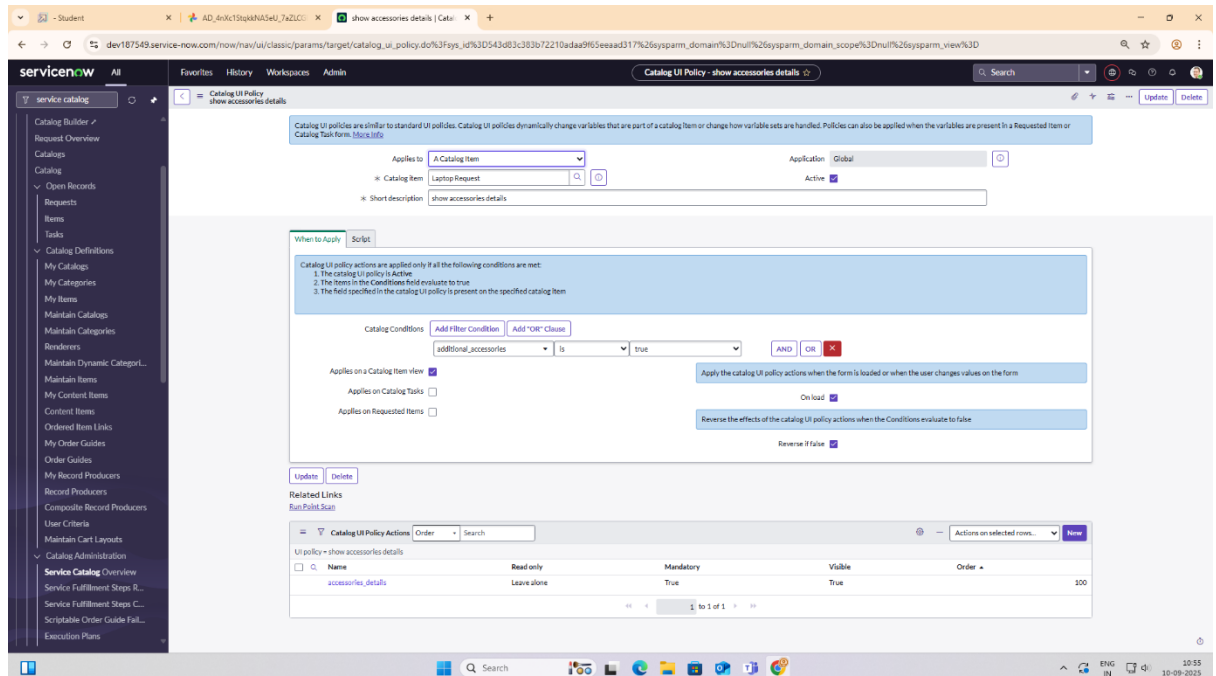
Milestone 5: Exporting Changes to Another Instance

Activity 1: Export Update Set as XML

1. Click on All >> search for update sets
2. Select local update set
3. Select created update set i.e. 'Laptop Request Project'

4. Set the state to 'Complete'

5. In the related list Update tab, updates are visible which we perform under this update set.

6. Click on export to XML ,it download one file


Milestone 6: Retrieving the Update Set

Activity 1: Import Update Set from XML into Another Instance

1. Open another instance in incognito window

2. Login with credentials

3. Click on all>> search for update sets

4. Select "Retrieved update set" under system update set

5. It open retrieved update set list and scroll down

6. Click on Import update set from XML
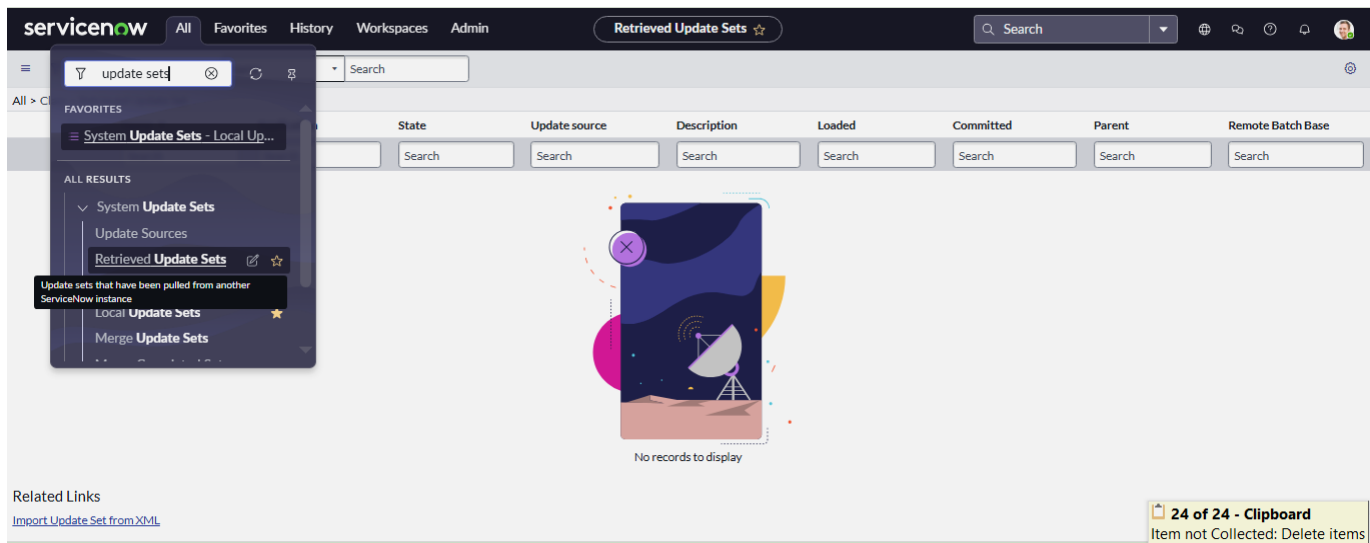
7. Upload the downloaded file in XML file

8. Click on Upload and it gets uploaded.
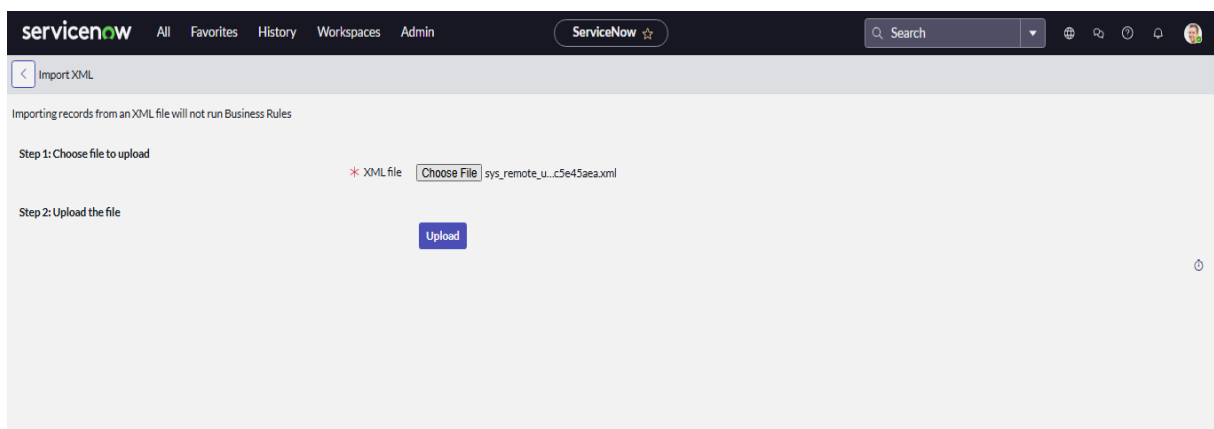
9. Open retrieved update set 'laptop request project'

10. Click on preview update set

11. And click on commit update set

12. And also see the related tab updates

13. After commiting update set in this instance we get all updates which are done in the previous instance

Milestone 7: Test Catalog Item

Activity 1: Verify Functionality of the Laptop Request Catalog Item

1. Search for service catalog in application navigator in target instance
2. Select catalog under service catalog
3. Select hardware category and search for 'laptop request' item
4. Select laptop request item and open it
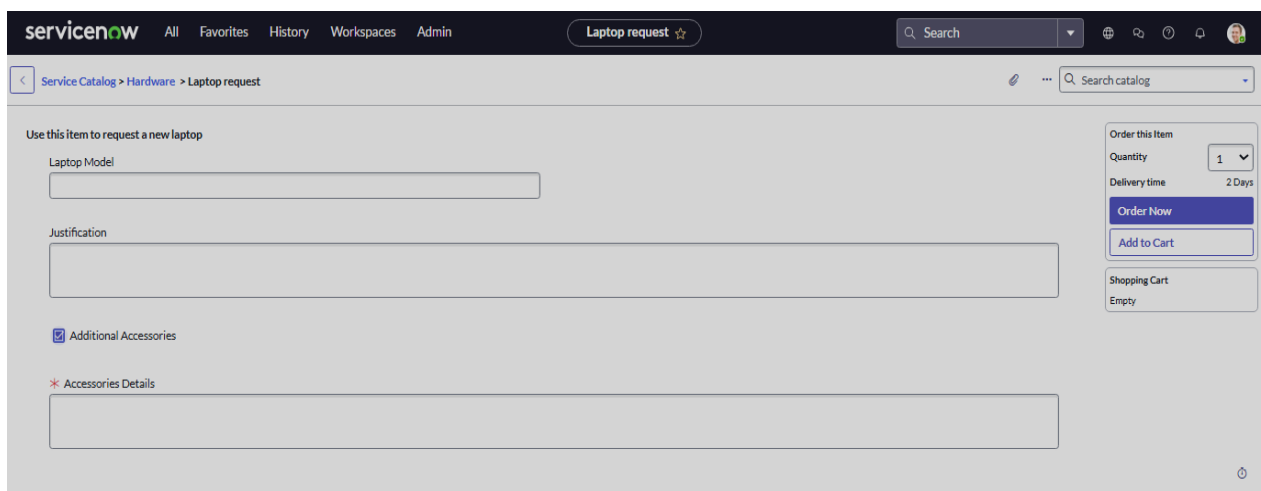5. It shows three variables only

6. As per our scenario, when we click on additional accessories checkbox then accessories details fields is visible and that should be mandatory
7. Now see the results,it fulfills our requirements

Conclusion:

 The Laptop Request Catalog Item project successfully streamlines the process of requesting laptops within the organization by leveraging ServiceNow's Service Catalog capabilities. Through the implementation of a dynamic catalog item, the project ensures that users have an intuitive and user-friendly interface, reducing errors and improving efficiency.This project demonstrates how ServiceNow can be used to replace manual, error-prone processes with automated, efficient, and user-centric solutions. It not only improves service delivery but also enhances employee satisfaction by providing a modern and streamlined request experience.