

# **Laptop Request Catalog Item**

**Team ID: LTVIP2026TMIDS37778**

**Team Size: 4**

**Team Leader: Puchakayala Madhu**

**Team member: Kosireddy Dameswara Rao**

**Team member: Kolli Mounika**

**Team member: Nadella Teja**

## **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.

## **Category**

ServiceNow System Administartor.

## **Skills**

UIPath RPA,Tanzu Application Service

## **TASK INITIATION Milestone 1: Create Local Update Set**

### **Activity 1: Create Update Set in ServiceNow**

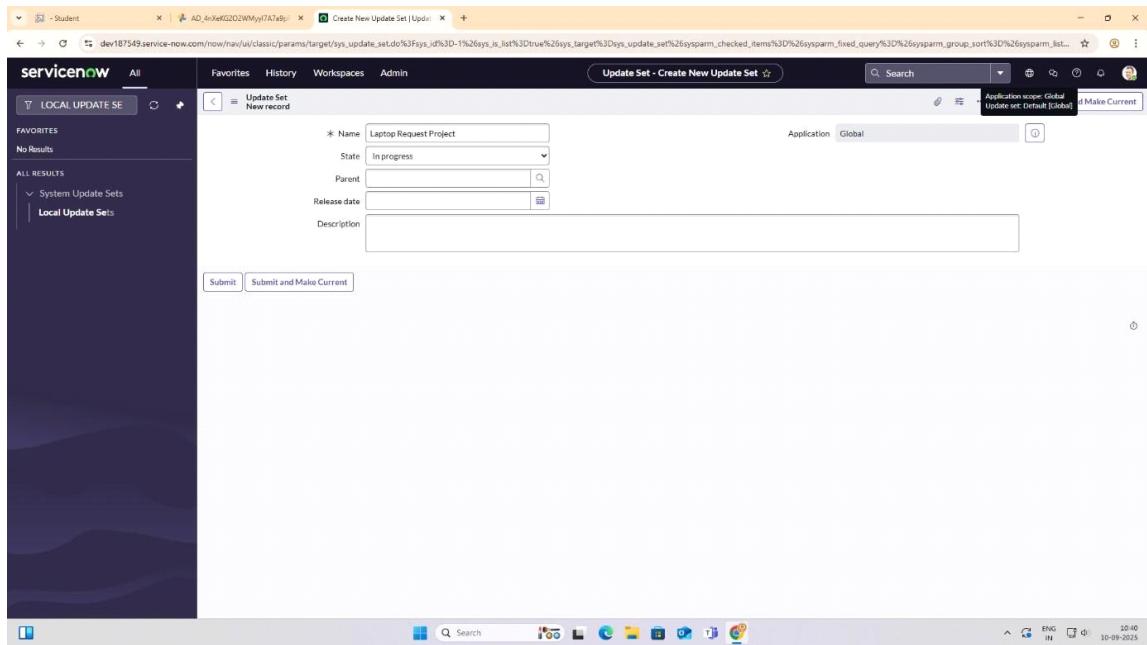
- Open ServiceNow instance.
- Click on All → search for Update Sets.
- Select Local Update Sets under System Update Sets.
- Click on New.
- Fill in the following details:

Name:

Laptop

Request

- Click Submit and make it Current.
- By clicking on the button it activates the update set



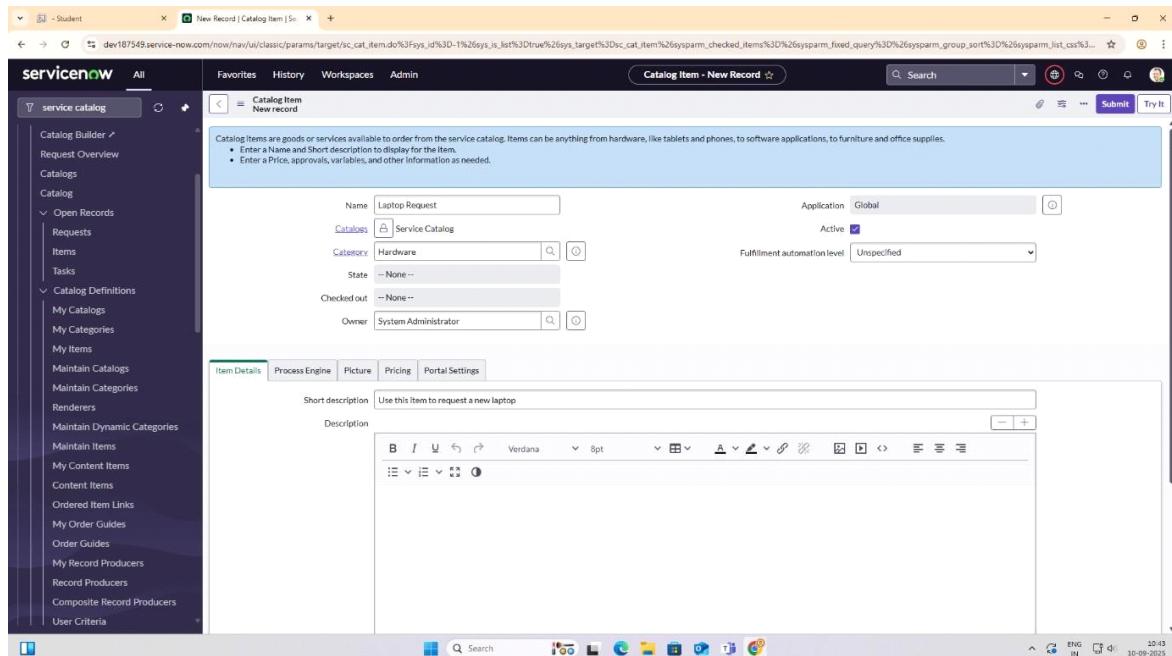
## Milestone 2: Create Service Catalog Item

### Activity 1: Create New Service Catalog Item in ServiceNow

- Open service now.
- Click on All >> service catalog
- Select maintain items under catalog definitions
- Click on New.

The screenshot shows the ServiceNow interface for managing catalog items. The left sidebar is titled 'service catalog' and contains various navigation links such as Catalog Builder, Request Overview, Catalogs, Catalog, Open Records, Requests, Items, Tasks, Catalog Definitions, My Catalogs, My Categories, My Items, Maintain Catalogs, Maintain Categories, Renderers, Maintain Dynamic Categories, Maintain Items, My Content Items, Content Items, Ordered Item Links, My Order Guides, Order Guides, My Record Producers, Record Producers, Composite Record Producers, and User Criteria. The main content area is titled 'Catalog Items' and displays a table of items. The table columns include Name, Short description, Active, Roles, Catalogs, Category, Price, Type, and Updated. The table lists numerous items like '3M Privacy Filter - Lenovo X1 Carbon', '3M Privacy Filter - MacBook Pro', '3M Privacy Filter - Macbook Pro Retina', 'Access', 'Acrobat', 'Add network switch to datacenter cabinet', 'Add/Remove users from group', 'Adobe Acrobat Pro', 'Adobe Creative Cloud', 'Apple iPad 3', 'Apple iPhone 13', 'Apple iPhone 13 pro', 'Apple iPhone 4 Cable', 'Apple iPhone 5', 'Apple iPhone 5 Cable', 'Apple iPhone 6s', 'Apple iPhone 6s Plus', 'Apple MacBook Pro 15''', 'Apple Thunderbolt to Ethernet Adapter', and 'Apple USB Ethernet Adapter'. The 'Category' column for the last two items is highlighted in blue, indicating they belong to the 'Service Catalog' category.

- 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
- 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

• Variable

1:Laptop

Model

Type:

Single line

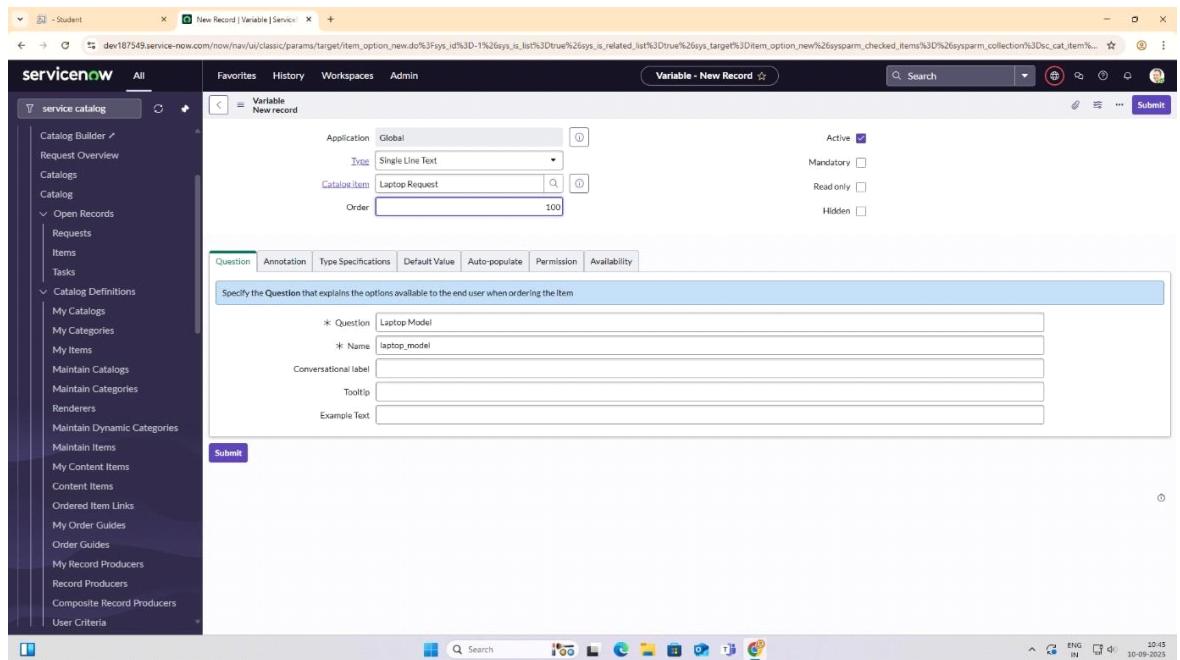
text Name:

laptop\_mod

el Order:100

- Click on submit

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



- Variable 2: Justification

Type: Multi

line text

Name:

justification

Order: 200

- Variable 3: Additional Accessories

Type

e: Checkbox Name:

additional\_accessories

Order: 300

- Variable 4:

Accessories

Details Type:

Multi line

text

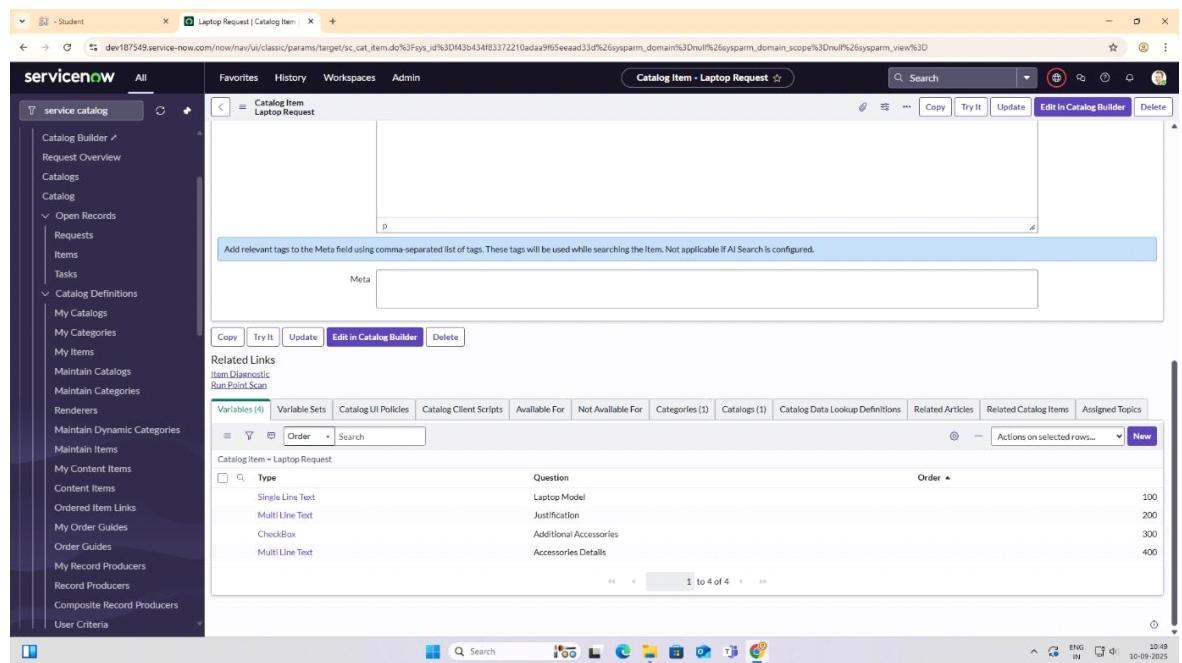
Name:accesso

ries\_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

- Click on all>> search for service catalog
- Select maintain item under catalog definition
- Search for 'laptop request' which is created before
- Select 'laptop request' and scroll down click on "Catalog Ui policies"
- In the catalog ui policies related list tab click on new
- Give short description as: show accessories details

- Set the Catalog Condition in the related list tab  
 ‘when to apply’ [field: additional\_  
 accessories, operator: is, value: true]

The screenshot shows the ServiceNow interface for creating a new Catalog UI Policy. The left sidebar is the 'service catalog' menu. The main area is titled 'Catalog UI Policy - New Record'. The 'Applies to' dropdown is set to 'A Catalog Item' and has 'Laptop Request' selected. The 'Short description' field contains 'show accessories details'. The 'When to Apply' tab is active, showing a condition: 'additional\_accessories is true'. There are options to apply actions on load or reverse them if false.

- Click on **save**.(do not click on submit)
- Scroll down and select ‘catalog ui action’
- Then click on new button
- Select variable name as: accessories\_details

O

rder:

100

Man

dato

ry:

True

Visi

ble :

True

- Click on save and again click save button of the catalog ui policy form

Catalog Item: Laptop Request  
Variable name: accessories\_details  
Order: 100

Application: Global  
Mandatory: True  
Visible: True  
Read only: Leave alone  
Value action: Leave alone  
Field message type: None

Name: Reset Form  
Table: Shopping Cart [sc\_cart]  
Order: 100  
Action name: Reset Form

Client:  
Show insert:   
Show update:

Overrides:

Comments:

Hint:

Onclick:

Condition:

```
1 function resetForm() {  
2     g_form.clearForm(); // Clears all fields in the Form  
3     alert("The form has been reset.");  
4 }
```

## Milestone 4: Create UI Action

### Activity 1: Create Reset Form UI Action

- Open service now.
- Click on All >> search for ui action
- Select ui actions under system definition
- Click on new
- Fill the following details to

create ui action Table:

shopping cart(sc\_cart)

Order:100

Action name:

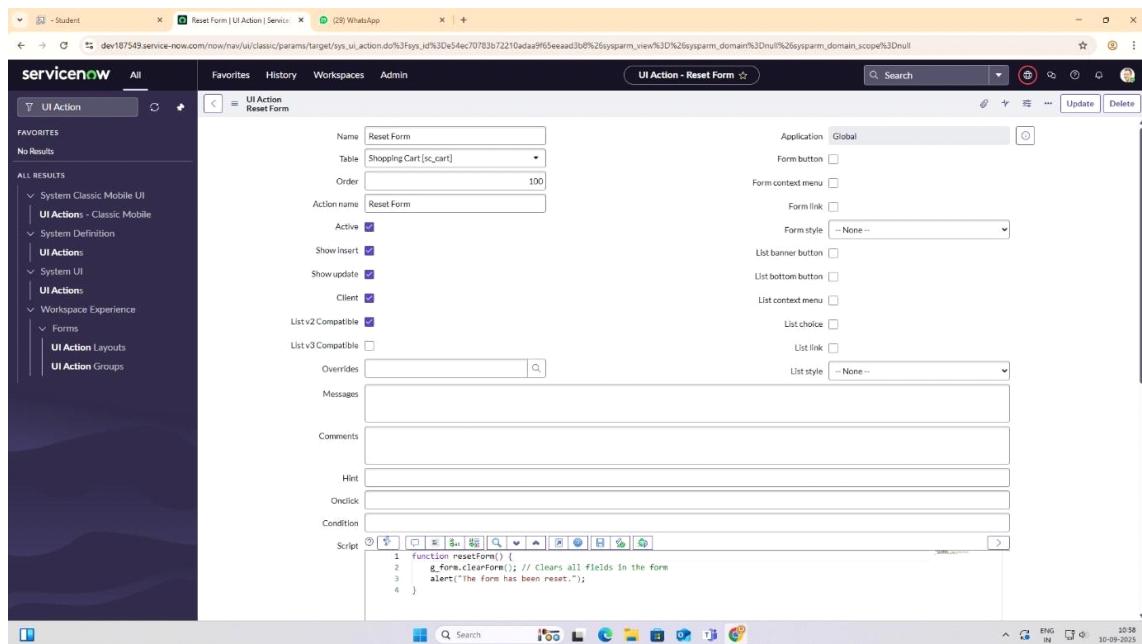
Reset form

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

- Click on All >> search for update sets
- Select local update set
- Select created update set i.e. ‘Laptop Request Project’
- Set the state to ‘Complete’
- In the related list Update tab, updates are visible which we perform under this update set.
- Click on export to XML ,it download one file

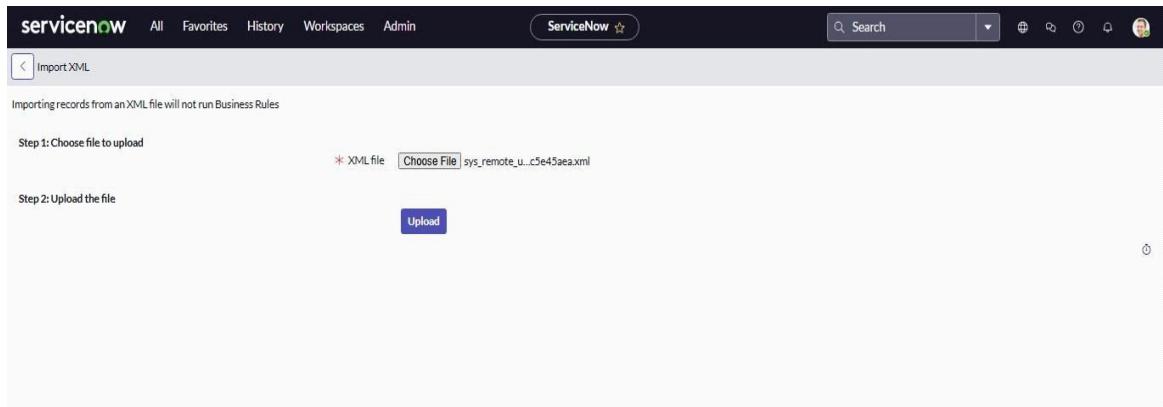
## Milestone 6: Retrieving the Update Set

### Activity 1: Import Update Set from XML into Another Instance

- Open another instance in incognito window
- Login with credentials
- Click on all>> search for update sets
- Select “Retrieved update set” under system update set
- It open retrieved update set list and scroll down
- Click on Import update set from XML

24 of 24 - Clipboard  
Item not Collected: Delete items

- Upload the downloaded file in XML file
- Click on Upload and it gets uploaded.



- Open retrieved update set ‘laptop request project’
- Click on preview update set
- And click on commit update set
- And also see the related tab updates
- After committing update set in this instance we get all updates which are done in the previous instance

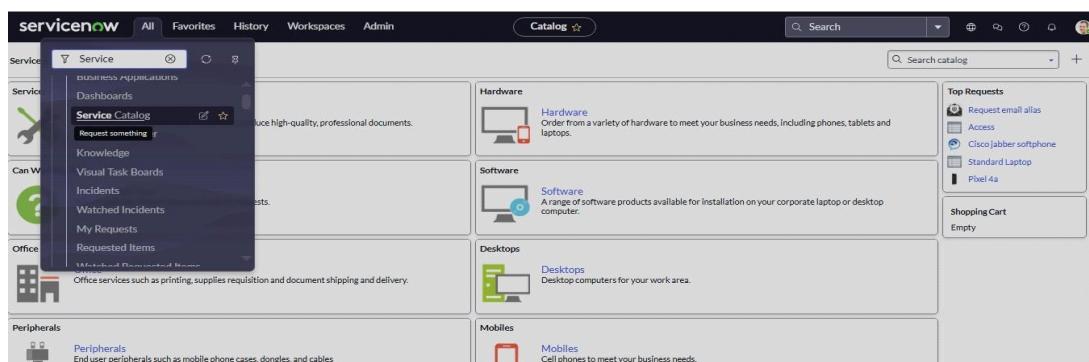
Name	Laptop Request	Committed	(empty)
Application	Global	Inserted	(empty)
Update source	(empty)	Deleted	(empty)
Parent	(empty)		
State	Loaded		
Loaded	2025-09-07 08:22:24		
Description			
Application name	Global		

Buttons: Update, Delete, Preview Update Set, Related Links.

## Milestone 7: Test Catalog Item

### Activity 1: Verify Functionality of the Laptop Request Catalog Item

- Search for service catalog in application navigator in target instance
- Select catalog under service catalog
- Select hardware category and search for ‘laptop request’ item
- Select laptop request item and open it
- It shows three variables only

A screenshot of the ServiceNow catalog page for the 'Hardware' category. The top navigation bar shows 'servicenow', 'All', 'Favorites', 'History', 'Workspaces', 'Admin', and a 'Hardware' button. The main content area displays a list of items:

- iPad pro: Request for iPad pro, Preview
- Laptop Model: Use this item to request a new laptop
- Laptop request: Use this item to request a new laptop
- Sales Laptop: Acer Aspire NX, Preview
- Standard Laptop: Lenovo - Carbon x1, Preview
- Apple Watch: Apple Watch - Their most personal device ever, Preview

Each item has a small icon to its left and price information to its right.

- 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

The screenshot shows the ServiceNow Service Catalog interface for a 'Laptop request'. At the top, there's a navigation bar with links for 'All', 'Favorites', 'History', 'Workspaces', and 'Admin'. The title 'Laptop request' is displayed above a search bar. Below the search bar, a breadcrumb trail shows 'Service Catalog > Hardware > Laptop request'. The main content area contains fields for 'Laptop Model' and 'Justification', both represented by large text input boxes. To the right of these fields is a sidebar titled 'Order this Item' which includes dropdowns for 'Quantity' (set to 1) and 'Delivery time' (set to 2 Days), and a blue 'Order Now' button. Below this is another sidebar for 'Shopping Cart' which is currently empty. There are also checkboxes for 'Additional Accessories' and 'Accessories Details'.

## 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.