



## Laptop Request Catalog Item

**Team ID : NM2025TMID07340**

**Team Size : 4**

**Team Leader : SHALINI M**

**Team member : SANTHOSH R**

**Team member : SABHAREESAN G**

**Team member : SARAN R**

**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.

### TASK INITIATION

#### Milestone 1 : Update sets

Activity 1 : create Local Update Set

1. Open service now.
2. Click on All >> search for update sets
3. Select local update sets under system update sets
4. Click on new
5. Fill the following details to create a update set as: “Laptop Request”

6. Click on submit and make current
7. By clicking on the button it activates the update set .

The screenshot shows the 'Update Set - Create Laptop Request Project 2' page. The 'Name' field is populated with 'Laptop Request Project'. The 'State' dropdown is set to 'In progress'. The 'Description' field is empty. At the bottom of the form, there are three buttons: 'Submit', 'Save', and 'Submit and Make Current'. The 'Submit and Make Current' button is highlighted with a red box.

## Milestone 2: Service Catalog Item

### Activity: Create Service Catalog Item

1. Open service now.
2. Click on All >> service catalog
3. Select maintain items under catalog definitions
4. Click on New.

The screenshot shows the 'Catalog Items' list screen. In the sidebar, 'Catalog Definitions' is selected. A new item is being created, indicated by the 'New' button at the top right. The 'Name' field of the new item is currently 'Laptop Request'.

Name	Short description	Active	Roles	Catalogs	Category	Price	Type	Updated
Search	Search	Search	Search	Search	Search	\$43.19	Item	2022-11-20 20:46:33
3M Privacy Filter - Lenovo X1 Carbon	Privacy Filter - X1 Carbon	true		Service Catalog	Peripherals	\$42.23	Item	2022-11-20 20:46:33
3M Privacy Filter - MacBook Pro	Privacy Filter	true		Service Catalog	Peripherals	\$40.31	Item	2022-11-20 16:00:00
3M Privacy Filter - Macbook Pro Retina	Privacy Filter	true		Service Catalog	Peripherals	\$0.00	Item	2016-08-18 10:52:30
401(k) Plan Enrollment	Enroll in or modify your 401(k) plan	true		Human Resources Catalog	Benefits	\$0.00	Item	2016-05-20 17:59:56
401(k) Plan Enrollment Inquiry	Ask a question about your retirement plan...	true		Human Resources Catalog	Benefits	\$0.00	Item	2016-05-20 17:59:55
401(k) Retirement Plan Inquiry	Ask a question about your existing plan	true		Human Resources Catalog	Benefits	\$139.99	Item	2022-11-20 20:46:33
Access	Microsoft Access	true		Service Catalog	Software	\$139.99	Item	2022-11-20 20:46:33
Acrobat	Adobe Acrobat	true		Service Catalog	Software	\$139.99	Item	2022-12-05 20:46:33
Add Beneficiaries	Add or modify your beneficiaries	true		Human Resources Catalog	Benefits	\$0.00	Item	2016-06-13 16:48:57
Add network switch to datacenter cabinet	This standard change template describes ...	true		Service Catalog	Network Standard Changes	\$0.00	Item	2024-10-21 07:20:03
Add/Remove users from group	Add/Remove users from group	true		Service Catalog	Services	\$0.00	Item	2022-12-05 10:17:33

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 Variable

### 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

Type	Question	Order
Single Line Text	Laptop Model	100
Multi Line Text	Justification	200
CheckBox	Additional Accessories	300
Multi Line Text	Accessories Details	400

## Milestone 3: UI POLICY

### Activity: Create Catalog Ui policies

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

The screenshot shows the ServiceNow Catalog UI Policy configuration screen. The policy is named 'Catalog UI Policy - Show Accessories Details'. It applies to 'A Catalog Item' named 'Laptop Request'. The application is set to 'Global' and is active. The 'When to Apply' section is set to 'Script'. The 'Catalog Conditions' section contains a condition: 'additional.accessories' is true. The 'Catalog Policy Actions' section contains one action named 'accessories\_details' with an order of 100.

## Milestone 4: UI ACTION

### Activity 1: Create 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

UI Action - Reset Form

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

Application: Global  
Form button:   
Form context menu:   
Form link:   
Form style: -- None --  
List banner button:   
List bottom button:   
List context menu:   
List choice:   
List link:   
List style: -- None --

## Milestone 5: Export Update set

### Activity: Exporting changes to another instances

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

Update Set - Laptop Request Project

\* Name: Laptop Request Project  
State: Complete  
Parent:   
Release date:   
Install date:   
Installed from:   
Description:

Application: Global  
Created: 2025-01-26 22:20:46  
Created by: admin  
Merged to:

Related Links: Export to XML, Merge or Extract Another Update Set, Scan Update Set, ISN Utils: Versions (0)

Customer Updates (11)

Created	Type	View	Target name	Updated by	Remote update set	Action
2025-01-26 22:40:32	Catalog UI Policy	Show Accessories Details	admin	(empty)		INSERT_OR_UPDATE
2025-01-26 22:43:26	Catalog UI Policy Action	accessories_details	admin	(empty)		INSERT_OR_UPDATE

## Milestone 6: Login to another Instance

### Activity 1: Retrieving the update set

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

The screenshot shows the ServiceNow interface with the 'Retrieved Update Sets' list. The sidebar on the left has a red box around the 'Retrieved Update Sets' link under 'System Data Management'. The main area shows a table of retrieved update sets with columns: Name, Application, State, Update source, Description, Loaded, Committed, Parent, and Remote Batch Base. One row is selected, and the 'Import Update Set from XML' button in the 'Related Links' section at the bottom is highlighted with a red box.

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

The screenshot shows the 'Import XML' dialog box. It displays a message about importing records from an XML file and not running business rules. It has two steps: 'Step 1: Choose file to upload' with a 'Choose File' button (highlighted with a red box) and 'Step 2: Upload the file' with an 'Upload' button (highlighted with a red box).

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 committing **update** set in this instance we get all updates which are done in the previous instance

The screenshot shows the ServiceNow interface with the following details:

- Left Sidebar:** Shows 'FAVORITES' with 'No Results' and 'ALL RESULTS' section containing 'System Data Management', 'Update Jobs', 'System Update Sets', 'Update Sources', 'Retrieved Update Sets', 'Update log', 'Local Update Sets', 'Merge Update Sets', 'Merge Completed Sets', 'Update Sets to Commit', and 'Update Set Commit History'.
- Top Bar:** Shows 'Favorites', 'History', 'Workspaces', 'Retrieved Update Set - Laptop Request Project', 'Search', and other navigation icons.
- Main Content:**
  - Retrieved Update Set - Laptop Request Project:** Details include Name: Laptop Request Project, Application: Global, State: Previewed, Loaded: 2025-01-27 22:14:51, and various statistics like Inserted: 11, Updated: 0, Deleted: 0, Collisions: 0, and Total: 11.
  - Buttons:** 'Update', 'Delete', 'Run Preview Again', and 'Commit Update Set' (highlighted with a red box).
  - Related Links:** 'Show All Preview Records' and 'Customer Updates (11)'.

## Milestone 7: Testing

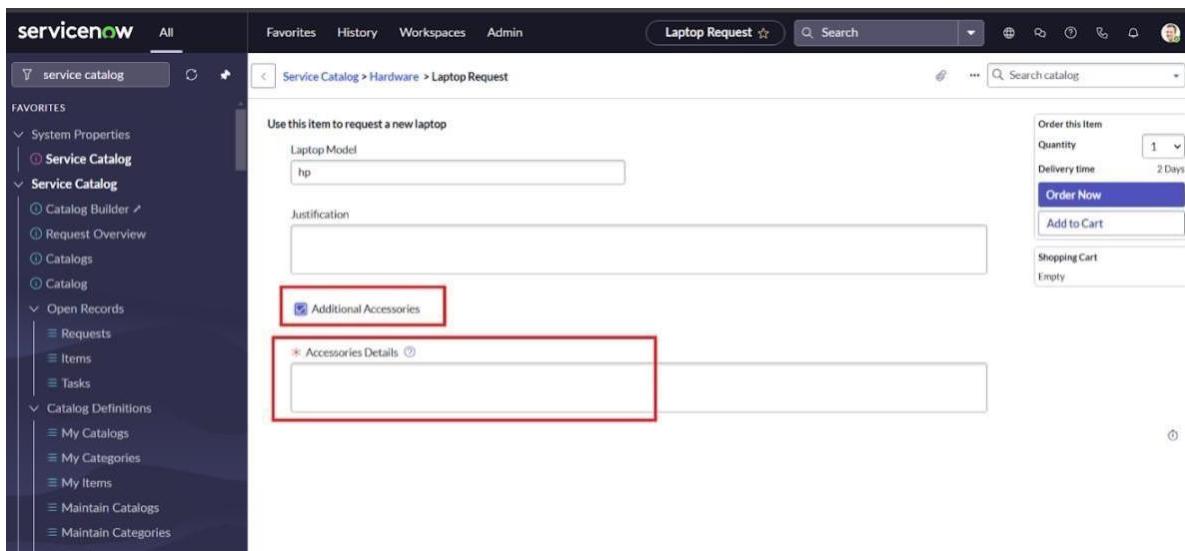
### Activity 1: Test Catalog Item

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

The screenshot shows the ServiceNow interface with the following details:

- Left Sidebar:** Shows 'FAVORITES' with 'System Properties', 'Service Catalog', 'Service Catalog', 'Catalog Builder', 'Request Overview', 'Catalogs', and 'Catalog' (highlighted with a red box). It also includes sections for 'Open Records' and 'Catalog Definitions'.
- Top Bar:** Shows 'Favorites', 'History', 'Workspaces', 'Admin', 'Laptop Request', 'Search', and other navigation icons.
- Main Content:**
  - Service Catalog > Hardware > Laptop Request:** A form with fields for 'Laptop Model' (input field), 'Justification' (text area), and 'Additional Accessories' (checkbox).
  - Order Options:** 'Order this item' section with 'Quantity' (1), 'Delivery time' (2 Days), 'Order Now' button, and 'Add to Cart' link.
  - Cart:** 'Shopping Cart' section with 'Empty' link.

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.