

Laptop Request Catalog Item

Date	31 October 2025
Team ID	NM2025TMID06949
Project Name	Laptop Request Catalog Item
Maximum Marks	4 Marks

Introduction

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.

Key Features

- **Update Set :** Create or select an update set to track changes.
- **Service Catalog Item:** Create the laptop request item in the Service Catalog.
- **UI Policy:** Define UI policies to control form behavior
- **UI Action:** Add buttons or links (UI actions) for user interactions.
- **Export Update Set:** Export the update set after completing configurations.
- **Login to Another Instance:** Access a different ServiceNow instance
- **Testing:** Test the catalog item to ensure all functionality works as expected.
- **Conclusion:** Final thoughts or wrap-up.

Pre-requisites

1. Active **ServiceNow Personal Developer Instance (PDI)** obtained from developer.servicenow.com.
2. Basic familiarity with:
 - **ServiceNow navigation and configuration**
 - **Tables, forms, and related lists**
 - **Business rules and Flow Designer**
3. GitHub or other repository (optional) for maintaining documentation and exported update sets

Requirements Functional

Requirements

- The system must allow the creation of an update set to track changes made for the catalog item.
- Users should be able to create a new catalog item specifically for laptop requests.
- The item should include relevant fields such as model, purpose, justification, and approval.
- Show/hide fields based on conditions.
- Make fields mandatory or read-only.
- Add actions like submit, reset, or cancel to the catalog item form for user interaction.
- Ability to export the completed update set for transfer to another instance.

Non-Functional Requirements

- The catalog item form should load within 2 seconds.
- Request submission should be processed without noticeable delay.
- The catalog item form should be user-friendly and intuitive.
- Field labels and help text must be clear and meaningful.
- Only authorized users (e.g., IT Admins or Catalog Admins) should be able to create or modify the catalog item.
- Data in the form should be validated to prevent injection attacks.

Phases of the Project

Ideation Phase

The **Ideation Phase** identifies the need to automate laptop requests using ServiceNow. Stakeholders discuss challenges with manual processes and propose a catalog item with dynamic forms and approvals. Key requirements are outlined, feasibility is assessed, and the idea is approved to move forward.

Requirement Analysis Phase

- Identify required fields (e.g., laptop model, justification).
- Define user roles (requester, approver, fulfiller).
- Document functional and non-functional requirements.
- Set business rules and approval conditions.
- Plan request workflow (submit → approve → fulfill).
- Use ServiceNow components like Catalog Item, UI Policies, and Update Sets.
- Validate and finalize requirements with stakeholders.

Project Planning Phase

The team executed these milestones:

1. ServiceNow Instance Setup

- Signed up at developer.servicenow.com and requested a Personal Developer Instance (PDI)
- Filled necessary details; received instance access credentials via email
- Logged in and prepared the instance for development.

2. Creation of Local Update Set

1. Access Navigation

- Open ServiceNow and go to All → Update Sets.

2. Choose Local Update Sets

- Select Local Update Sets under the *System Update Sets* module.

3. Create New Update Set

- Click New to open the update set form.
- Enter a meaningful name, e.g., "Laptop Request".

4. Save and Submit

- Click Submit to save the new update set.

5. Activate the Update Set

- After submission, click Make Current.
- This ensures all your changes are tracked under this update set.

The screenshot shows the 'Update Set - Create Laptop Request Project 2' form in ServiceNow. The 'Name' field is populated with 'Laptop Request Project 2'. The 'State' dropdown is set to 'In progress'. The 'Parent' and 'Release date' fields are empty. The 'Description' field is also empty. At the bottom, there are two buttons: 'Submit' and 'Submit and Make Current'.

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.

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'

Add variables

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

The screenshot shows the ServiceNow interface for creating a new variable. At the top, the title bar says "Variable - New Record". The main form has fields for Application (Global), Type (Single Line Text), Catalog Item (Laptop Request), Order (104), and checkboxes for Active (checked), Mandatory, Read-only, and Hidden. Below the form is a tabbed section with "Question" selected, containing fields for Question (Laptop Model), Name (laptop_model), Conversational label, Tooltip, and Example Text. A "Submit" button is at the bottom.

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

The screenshot shows the Catalog Item - Laptop Request page. It displays a table of variables with columns for Type, Question, and Order. The variables listed are: Laptop Model (Single Line Text, Order 100), Justification (Multi Line Text, Order 200), Additional Accessories (Checkbox, Order 300), and Accessories Details (Multi Line Text, Order 400). The top navigation bar includes "Edit in Catalog Builder" and "Delete" buttons.

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

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]

The screenshot shows the 'Catalog UI Policy - New Record' form in ServiceNow. The 'Applies to' dropdown is set to 'A Catalog Item' and the 'Catalog Item' dropdown is set to 'Laptop Request'. The 'Short description' field contains 'show accessories details'. The 'When to Apply' section has a catalog condition: 'additional_accessories IS true'. It also includes options for applying the policy on catalog item view, tasks, or requested items, and checkboxes for 'On load' and 'Reverse effects'.

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 'Catalog UI Policy Action - New Record' form in ServiceNow. The 'Catalog Item' dropdown is set to 'Laptop Request' and the 'Variable name' dropdown is set to 'accessories_details'. The 'Order' field is set to 100. The 'Mandatory' field is set to True, and the 'Visible' field is also set to True. Other fields like Read only, Value action, and Field message type are set to their default values.

Exporting changes to another instances

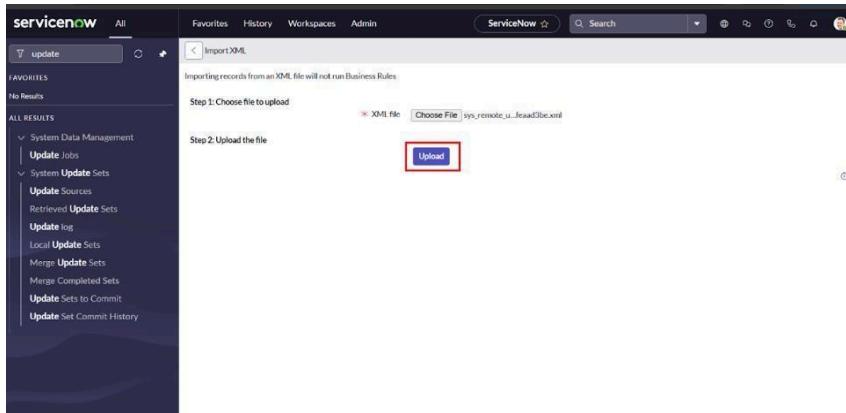
1. Click on All >> search for update sets
2. Select local update set
3. Select created update set i.e. ‘Laptop Request Project’
4. Set the state to ‘Complete’
5. In the related list Update tab, updates are visible which we perform under this update set.
6. Click on export to XML ,it download one file

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

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

This screenshot shows the 'Retrieved Update Set - Laptop Request Project' page. It displays the following details:

- Name: Laptop Request Project
- Application: Global
- Update source: Parent (with a search icon)
- State: Previewed
- Loaded: 2025-06-24 07:39:12
- Committed: 0
- Inserted: 0
- Updated: 10
- Deleted: 0
- Collisions: 0
- Total: 10

Buttons at the bottom include: Update, Delete, Run Preview Again, Commit Update Set, and a toolbar with a magnifying glass and a refresh icon.

The 'Related Links' section shows 'Customer Updates (10)' and 'Child Update Sets'. A search bar and an 'Actions on selected rows...' dropdown are also present.

This screenshot shows the 'Customer Updates (10)' table. The columns are:

	Name	Type	Target name	Table	View	Action
<input type="checkbox"/>	catalog_ui_policy_044adfb06c3522210e969b16d4013102	Catalog UI Policy	show accessories details			INSERT_OR_UPDATE
<input type="checkbox"/>	catalog_ui_policy_action_f07aff46c3522210e969b16d4013146	Catalog UI Policy Action	accessories_details			INSERT_OR_UPDATE
<input type="checkbox"/>	item_option_new_f599f82c3522210e969b16d40131fb	Variable	Accessories Details			INSERT_OR_UPDATE
<input type="checkbox"/>	item_option_new_42099bec3522210e969b16d40131ae	Variable	Additional Accessories			INSERT_OR_UPDATE
<input type="checkbox"/>	item_option_new_a8c7f742c3522210e969b16d40131cf	Variable	Laptop Model			INSERT_OR_UPDATE
<input type="checkbox"/>	item_option_new_a858582c3522210e969b16d4013123	Variable	Justification			INSERT_OR_UPDATE
<input type="checkbox"/>	sc_cat_item_11d6930ec3122210e969b16d4013199	Catalog Item	Laptop Request			INSERT_OR_UPDATE
<input type="checkbox"/>	sc_cat_item_catalog_b18ff02c3522210e969b16d40131fe	Catalog Items Catalog	Service Catalog.Laptop Request			INSERT_OR_UPDATE
<input type="checkbox"/>	sc_cat_item_category_75871342c3522210e969b16d4013103	Catalog Item Category	Hardware.Laptop Request			INSERT_OR_UPDATE
<input checked="" type="checkbox"/>	sys_ui_action_a61d9fd3522210e969b16d40131f1	UI Action	Reset Form	Shopping Cart[sc_cart]		INSERT_OR_UPDATE

At the bottom, there are navigation buttons for page 1 to 10 of 10.

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
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.