LAPTOP REQUEST CATALOG ITEM

Project Description

The Laptop Request Catalog Item project was developed to streamline and automate the process of requesting laptops within the organization. Previously, employees used manual forms or emails to request laptops, resulting in delays, lack of tracking, and inefficiencies.

By leveraging ServiceNow's Service Catalog capabilities, this project implements a user-friendly, dynamic catalog item allowing employees to request laptops efficiently with structured data capture, dynamic field visibility, reset functionalities, and deployment-ready update sets.

This implementation enhances user experience, reduces errors, and improves overall service delivery within the organization.

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.

Current Challenges:

Manual forms leading to processing delays

Lack of field validation, causing data errors

No dynamic behavior (e.g., conditional fields)

No form reset option for users to modify inputs

Objective:

to address these challenges by creating a Service Catalog Item in ServiceNow that:

Allows users to easily request laptops

Uses dynamic fields with UI Policies

Provides a reset functionality with UI Actions

Captures all changes in Update Sets for deployment

Enhances employee satisfaction with intuitive

Technical Architecture:

Pre-requisites

ServiceNow Admin access

Understanding of Update Sets, Catalog Items, UI Policies, UI Actions

Knowledge of basic deployment processes in ServiceNow

GitHub for version control documentation

Microsoft Word or Canva for final report preparation

Overview Activities:

Update Set Creation - to capture all configurations

- 1. Catalog Item Development building the Laptop Request form
- 2. UI Policy Implementation to create dynamic field behaviors

Activities

- 3. UI Action Creation to add reset functionality
- 4. Export and Import Update Sets for deployment Testing to validate fuction Overview
- 5. Update Set Creation to capture all configurations
- 6. Catalog Item Development building the Laptop Request form
- 7. UI Policy Implementation to create dynamic field behaviors Activities
- 8. Update Set Creation to capture all configurations
- 9. UI Action Creation to add reset functionality
- 10. Export and Import Update Sets for deployment
- 11. Testing to validate functionality end-to-end

Milestone 1: Update Set

Activity 1.1: Create Update Set

Navigate to System Update Sets > Local Update Sets.

Click New.

Enter Name: Laptop Request Catalog Item.

Make it Active to capture all configurations performed under this project

Update Sets in ServiceNow allow tracking of all configurations made during development.

This is crucial for deployment to other instances like testing or production environments.

Always ensure your update set is active before beginning configurations to avoid missing

changes.

Milestone 2: Service Catalog Item

Item Activity 2.1: Create Laptop Request Catalog

navigate to Service Catalog > Maintain Items.

Click New.

Enter the following details:

Name: Laptop Request

Catalog: Service Catalog

Category: Hardware

The catalog item acts as the main request form where users can initiate their laptop

requests within the ServiceNow Service Catalog portal.

Activity 2.2: Add Variables to the Catalog Item

Add the following variables to capture user inputs:

1. Laptop Type (Choice)

Options: Dell, HP, MacBook, Lenovo

2. Purpose (Single Line Text)

Example: For project development, testing, research, etc.

3. Justification (Multi-line Text)

Users can explain the business justification for their laptop request.

4. Required By Date (Date Picker)

5. Select the expected delivery date for planning and fulfillment.

Arrange variables logically to ensure user-friendly data entry. Configure variable layout using the Variable Editor for neat display within the catalog item form.

Add the following variables to capture user inputs:

6. Laptop Type (Choice)

Options: Dell, HP, MacBook, Lenovo

7. Purpose (Single Line Text)

Example: For project development, testing, research, etc.

8. Justification (Multi-line Text)

Users can explain the business justification for their laptop request.

9. Required By Date (Date Picker)

Select the expected delivery date for planning and fulfillment.

Arrange variables logically to ensure user-friendly data entry. Configure variable layout using the Variable Editor for neat display within the catalog item form.

Add the following variables to capture user inputs:

10. Purpose (Single Line Text)

Example: For project development, testing, research, etc.

10. Justification (Multi-line Text)

Users can explain the business justification for their laptop request.

11. Required By Date (Date Picker)

Select the expected delivery date for planning and fulfillment.

re user inputs:

12.Laptop Type (Choice)

Options: Dell, HP, MacBook, Lenovo

Arrange variables logically to ensure user-friendly data entry. Configure variable layout using the Variable Editor for neat display within the catalog item form.

Milestone 3: UI Policy

Activity 3.1: Create UI Policy for Dynamic Behavior

Navigate to Service Catalog > Catalog Policy > UI Policies.

Click New.

Configure: Description: Show Justification field when Laptop Type is MacBook

Activity 3.2: Add UI Policy Actions

Under UI Policy Actions:

Select Field Name: Justification

Set Visible: True

Set Mandatory: True

Condition: Laptop Type is MacBook

UI Policies make forms interactive and dynamic. In this example, the Justification field is displayed and mandatory only if the Laptop Type selected is MacBook. Similar policies can be created for other laptop types based on organizational requirement

Milestone 4: UI Action

Activity 4.1: Create UI Action for Reset Functionality

Navigate to System UI > UI Actions.

Click New.

Configure:

Name: Reset Form

Table: Catalog Item or specific Laptop Request table

Action name: reset_form

Form button: True

```
Activity 4.2: Add Client Script
```

Include a script in the UI Action to clear fields when clicked. Example:

```
function resetForm() {
  g_form.clearValue('laptop_type');
  g_form.clearValue('purpose');
  g_form.clearValue('justification');
  g_form.clearValue('required_by_date');
}
```

The reset functionality improves usability by allowing users to clear their inputs and reenter details without reloading the entire form.

Milestone 5: Export Update Set

Activity 5.1: Export Update Set for Deployment Navigate to System Update Sets > Local Update Sets. Open the active update set created earlier.

Click Export to XML.

Save the exported XML file locally with proper naming (e.g.,

LaptopRequestCatalogItem.xml).

Exporting Update Sets ensures all configurations are packaged for import into other instances like testing or production environments, maintaining deployment consistency.

Milestone 6: Import Update Set

Activity 6.1: Import Update Set in Target Instance

Log in to the target instance (e.g., UAT or Production).

Navigate to System Update Sets > Retrieved Update Sets.

Click Import Update Set from XML.

Choose the exported XML file and upload.

After import, Preview the update set to check for any issues.

Click Commit to apply the changes to the instance.

Importing and committing Update Sets ensures that all your configurations are transferred securely and systematically between instance

Milestone 7: Testing

Activity 7.1: Validate the Catalog Item Functionality

After deployment, test the Laptop Request Catalog Item end-to-end:

Navigate to Service Catalog > Can Request Items.

Open the Laptop Request item.

Test:

Visibility of fields based on UI Policy conditions

Reset button functionality to clear inputs

Form submission flow generates a proper request record

Testing ensures all configured functionalities work as expected, providing a seamless user experience.

Exploring Application Features

Features Implemented

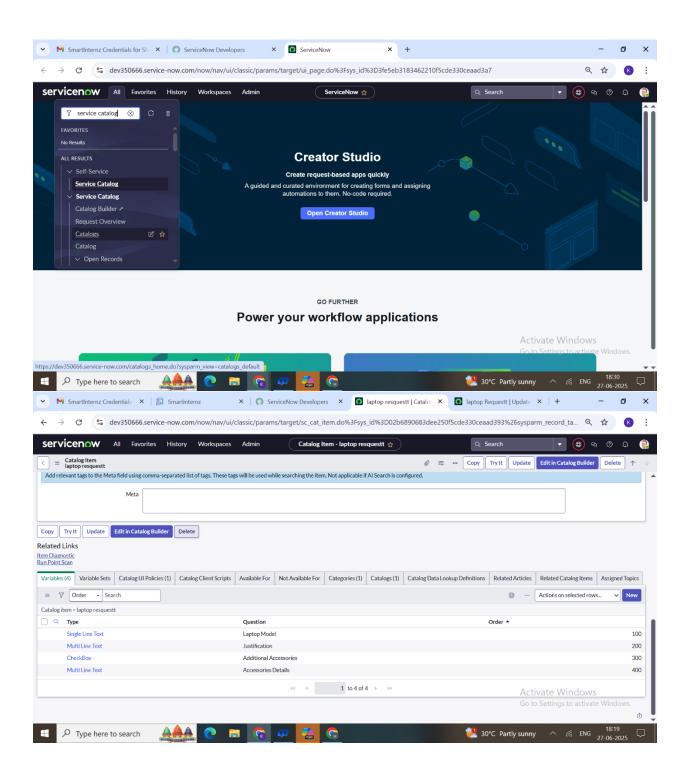
- 1. Dynamic Form Behavior UI Policies to show/hide fields
- 2. Reset Functionality UI Action with clear scripts
- 3. Structured Deployment Update Set export/import process
- 4. User-Centric Design Logical variable arrangement
- 5. Governance Ready All changes captured systematically for deployment

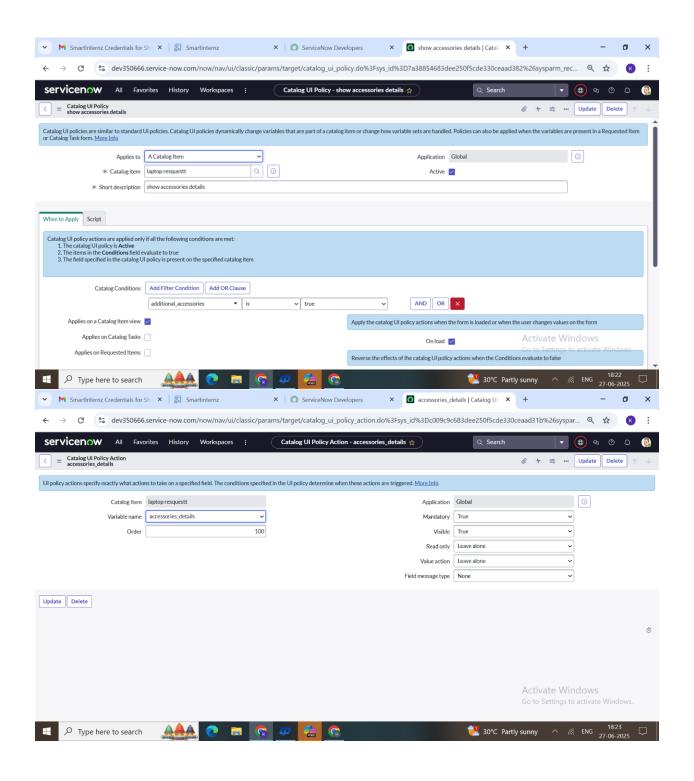
These features combine to automate the manual laptop request process, ensuring accuracy and efficiency within IT service delivery.

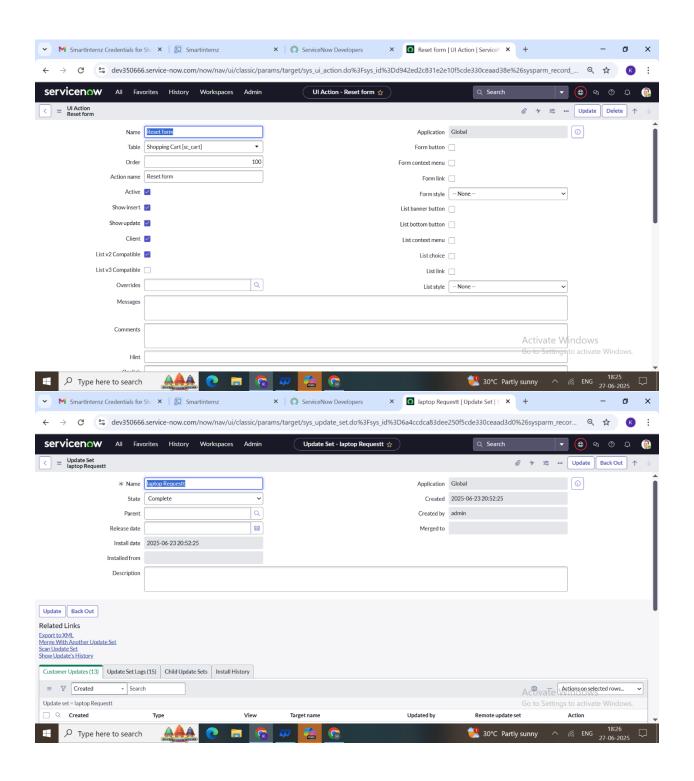
Screenshots:

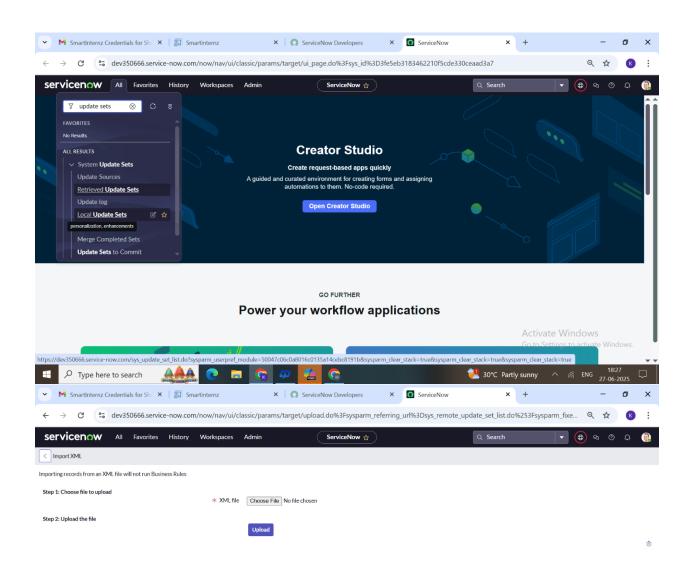
Screenshots to Include in Report

- 1. Update Set creation screen
- 2. Catalog Item creation page with all variables added
- 3. UI Policy configuration for dynamic fields
- 4. UI Action creation for Reset functionality
- 5. Export Update Set confirmation
- 6. Import Update Set preview and commit
- 7. Testing results submitted request form with field output



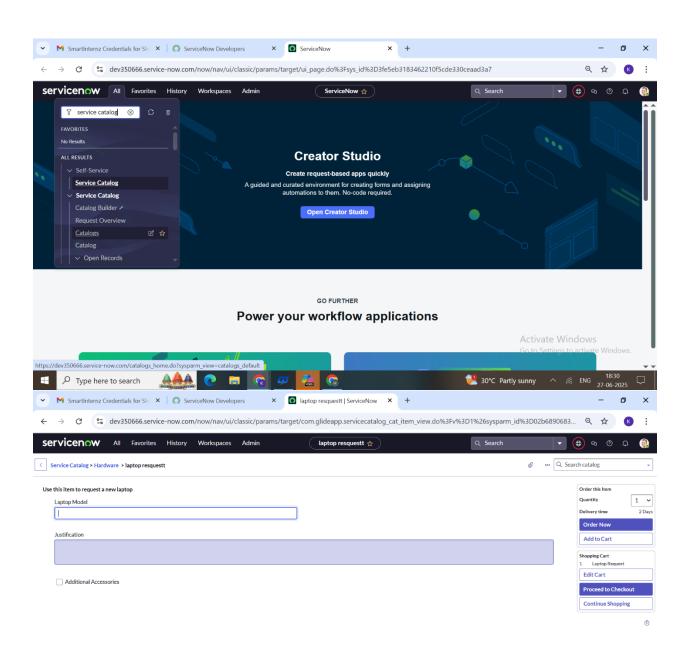






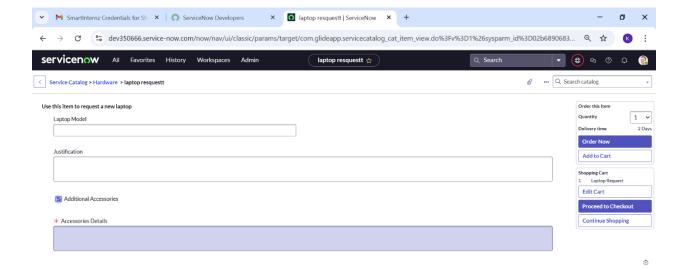
Activate Windows Go to Settings to activate Windows



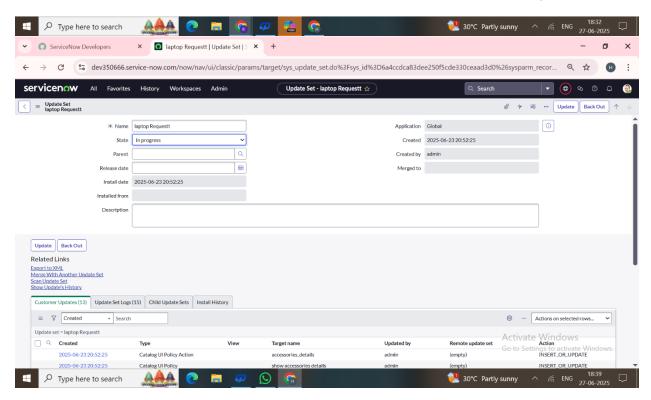


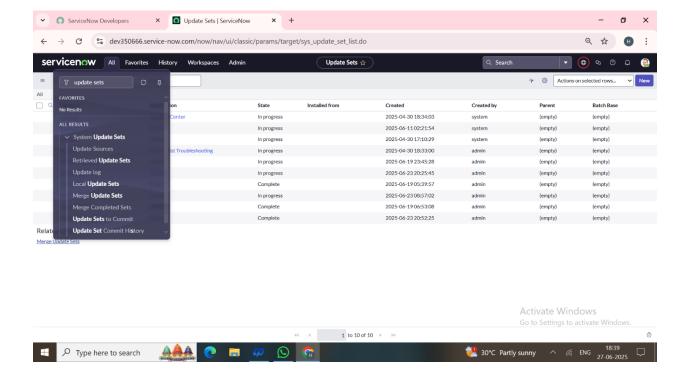
Activate Windows
Go to Settings to activate Windows.





Activate Windows
Go to Settings to activate Windows.





Video demo link:

https://drive.google.com/file/d/1x46l8UFFaloEhgsyosaTp29b1hBBDG0y/view?usp=drivesdk

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 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.