

Project Documentation

Laptop Request Catalog Item (ServiceNow Administration)

Team Details

- **Team ID:** LTVIP2025TMID30162
- **Team Size:** 3
- **Team Leader:** Sunkara Bhavana Sri Pallavi
- **Team Member:** Turumella Yogitha
- **Team Member:** Vangala Srinidhi

1. INTRODUCTION

Project Overview

Organizations often face challenges in managing IT hardware requests, especially laptops, due to outdated manual processes. These processes can lead to inefficiencies, delays, miscommunication, and a lack of transparency. The primary objective of this project is to develop a structured and automated method using ServiceNow's Service Catalog to simplify laptop request submissions. The implementation involves dynamic fields, reset functionalities, visibility conditions, and update set management to allow easy migration between instances.

Purpose

The purpose of this project is to replace the current manual workflow with a modern digital solution. Leveraging ServiceNow's capabilities, the project enables faster processing, standardized data input, enhanced user interaction, and maintainability across environments through update sets. It addresses the common issues faced by IT and employees while ensuring scalability and flexibility for future enhancements.

2. IDEATION PHASE

Problem Statement

Employees within the organization request laptops manually, either through email or verbal communication, leading to process bottlenecks. These methods often result in incomplete information, tracking difficulties, and errors. The absence of a guided form increases the burden on the IT department and reduces user satisfaction. This project aims to resolve these issues by creating a centralized, intelligent request mechanism within ServiceNow.

Empathy Map Canvas

- **Who?** Regular employees who need laptops to perform their job functions
- **What do they think and feel?** Frustrated by delays and unclear procedures
- **What do they see?** Inconsistent communication and lack of visibility into request status
- **What do they say and do?** Often ask IT teams for updates; rely on emails or follow-ups
- **What do they hear?** Complaints from peers facing similar issues
- **Pain?** Inefficiency, miscommunication, missed hardware delivery deadlines
- **Gain?** A structured, transparent, and smooth request experience

Brainstorming

During brainstorming, the team evaluated the pros and cons of multiple approaches:

- Continuing with manual emails
- Using spreadsheets or third-party tools
- Implementing a Service Catalog item in ServiceNow The ServiceNow approach was selected for its scalability, user interface capabilities, and seamless integration with the existing ITSM environment.

3. REQUIREMENT ANALYSIS

Customer Journey Map

The journey begins with a user logging into ServiceNow and accessing the service catalog. From there, the user selects the 'Laptop Request' option and completes the dynamic form. The form guides the user by showing or hiding relevant fields based on choices. Once submitted, the

request is automatically routed for processing. The entire process is designed to be intuitive, transparent, and efficient.

Solution Requirement

- A single catalog item for all laptop requests
- Fields to capture laptop model, justification, accessory needs
- Conditional display of accessory details
- A button to reset the form at any time
- Ability to capture and migrate changes via update sets
- Compatibility with any ServiceNow instance for import/export

Data Flow Diagram

User → Service Catalog → Laptop Request Form → UI Policies → Form Submission



Technology Stack

- **Platform:** ServiceNow (Orlando or higher)
- **Languages/Scripting:** JavaScript (GlideForm APIs), XML (Update Sets)
- **Modules Used:** Service Catalog, UI Policy, UI Action, Update Sets, Catalog Items
- **Testing:** Manual validation in target instance

4. PROJECT DESIGN

Problem Solution Fit

The solution design directly addresses the inefficiencies of the current system by introducing a digital form that adapts to user input. By incorporating visibility logic and mandatory fields, it ensures that users submit complete and relevant information, thereby reducing IT department backlogs.

Proposed Solution

A Laptop Request Catalog Item was developed in ServiceNow under the Hardware category. Key components include:

- A dynamic form with fields for model, justification, and accessories
- UI policies that conditionally display and validate the 'Accessories Details' field
- A reset button that clears the form for user convenience
- Export/import functionality using XML update sets for portability across instances

Solution Architecture

The architecture consists of the following layers:

- **Frontend:** Service Catalog form with dynamic UI behavior
- **Logic Layer:** UI Policies and UI Actions written in JavaScript
- **Persistence Layer:** Form data stored in ServiceNow tables and tracked via Update Sets
- **Deployment Layer:** XML-based update set export/import mechanism

5. PROJECT PLANNING & SCHEDULING

Project Planning

The project was executed in a sequential and iterative manner:

1. **Create Update Set** – to capture all development activities
2. **Catalog Item Creation** – forming the base item in the service catalog
3. **Variable Addition** – capturing user-specific request inputs
4. **UI Policies** – implementing conditional behavior for fields
5. **Reset Button** – enhancing usability via a custom UI action
6. **Testing** – validating behavior in the development instance
7. **Export and Import** – simulating deployment on a new instance

6. IMPLEMENTATION WORKFLOW

Create Local Update Set

- **Purpose:** To track all changes made during the development of the Laptop Request Catalog Item.

Steps:

1. Log in to your ServiceNow instance.
2. Navigate to:
All → Update Sets → Local Update Sets
3. Click **New** to create a new update set.
4. Fill in the following details:
 - **Name:** Laptop Request

Click **Submit** and then click **Make Current** to activate the update set.

The screenshot shows the ServiceNow Update Set creation interface. The main form has the following fields filled in:

- Name:** Laptop Request Project
- State:** Complete
- Parent:** (empty)
- Release date:** (empty)
- Install date:** (empty)
- Installed from:** (empty)
- Description:** (empty)
- Application:** Global
- Created:** 2025-06-16 02:16:13
- Created by:** admin
- Merged to:** (empty)

At the bottom of the form, there are two buttons: **Update** and **Back Out**. Below the form, under **Related Links**, are options: Export to XML, Merge With Another Update Set, and Scan Update Set. At the very bottom, there are tabs for Customer Updates (10), Update Set Logs, Child Update Sets, and Install History. A search bar and a dropdown menu are also visible at the bottom.

Create Service Catalog Item

- **Purpose:** To allow users to submit a request for a new laptop through the service catalog.

Steps:

1. Navigate to:
All → Service Catalog → Maintain Items
2. Click **New** to create a new catalog item.

3. Fill in the following details:

- **Name:** Laptop Request
- **Catalog:** Service Catalog
- **Category:** Hardware
- **Short Description:** Use this item to request a new laptop

4. Click **Save**

Name	Short description	Active	Roles	Catalogs	Category	Price	Type	Updated
Laptop Request	Use this item to request a new laptop	true		Service Catalog	Hardware	£0.00	Item	2025-06-16 02:39:56
Lenovo ThinkPad Power Adapter - 135W	For Lenovo Thinkpad, AC Adapter, 135W	true		Service Catalog	Peripherals	\$99.99	Item	2022-11-20 20:46:33
Lenovo Thinkpad Power Adapter - 90W	For Lenovo Thinkpad, T-Series, 90W	true		Service Catalog	Peripherals	\$60.00	Item	2022-11-20 20:46:33
Lenovo USB Ethernet Adapter	For Lenovo X1 Carbon	false		Service Catalog	Peripherals	\$28.79	Item	2022-11-20 20:46:33
Lenovo X1 Carbon Power Adapter	For Lenovo X1 Carbon	true		Service Catalog	Peripherals	\$60.00	Item	2022-11-20 20:46:33
Loamer Laptop	Short term, while computer is repaired/... ...	true		Service Catalog	Hardware	\$0.00	Item	2022-11-20 20:46:33
Logitech USB Headset for PC & Mac	PC/Mac Compatible Headset	true		Service Catalog	Peripherals	\$29.99	Item	2022-11-20 20:46:33
Logitech Wireless Mouse	PC / Mac Compatible	true		Service Catalog	Peripherals	\$35.88	Item	2022-11-20 20:46:33
Lotus Notes	IBM Lotus Notes	false		Service Catalog	Software	\$50.00	Item	2022-12-05 20:46:33
MacBook Air Power Adapter	Power Adapter	true		Service Catalog	Peripherals	\$75.84	Item	2022-11-20 20:46:33

Catalog items are goods or services available to order from the service catalog. Items can be anything from hardware, like tablets and phones, to software applications, to furniture and office supplies.

- Enter a Name and Short description to display for the item.
- Enter a Price, approvals, variables, and other information as needed.

Name	Laptop Request	Application	Global
Catalogs	Service Catalog	Active	<input checked="" type="checkbox"/>
Category	Hardware	Fulfillment automation level	Unspecified
State	--None--		
Checked out	--None--		
Owner	System Administrator		

Item Details

Short description: Use this item to request a new laptop

Description:

B I U (formatting toolbar)

Add Variables to the Catalog Item

- **Purpose:** To collect specific details from the requester regarding the laptop request.

Steps:

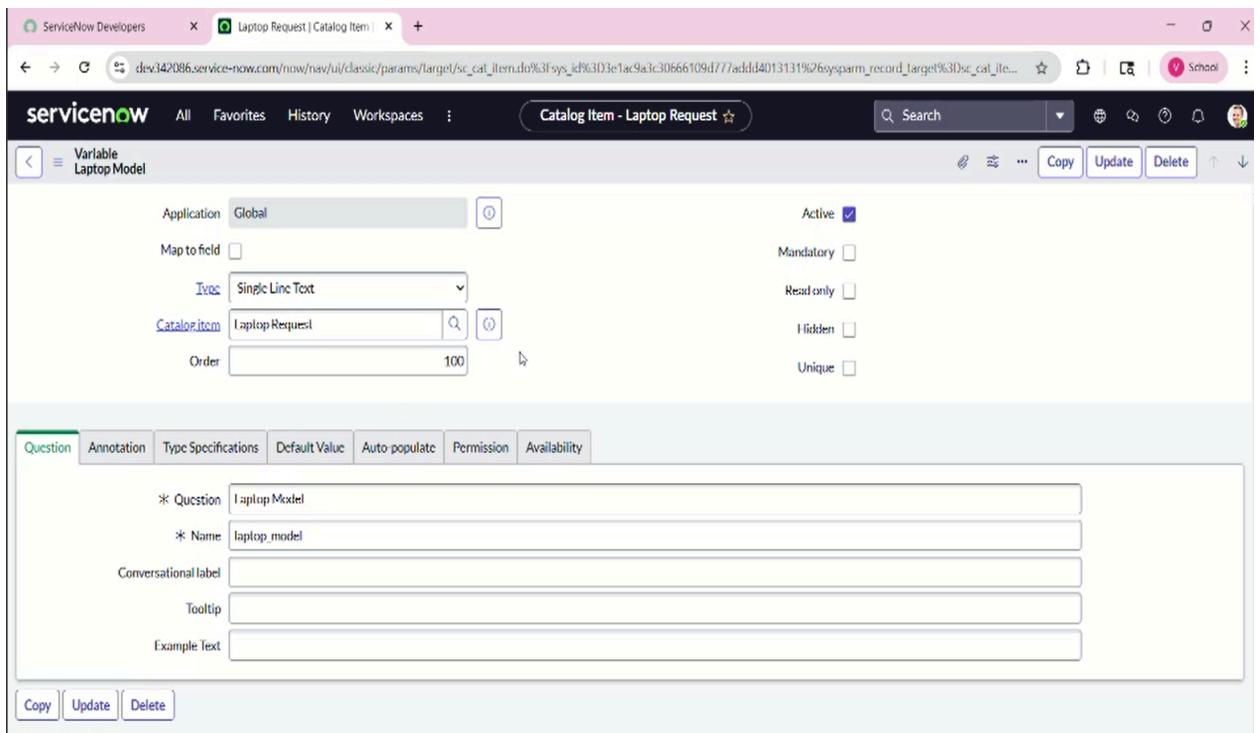
1. Scroll down to the **Variables** related list in the Catalog Item form.
2. Click **New** to add each variable:

Variable List:

Laptop Model	Type	Name	Order
Laptop Model	Single Line Text	laptop_model	100
Justification	Multi Line Text	justification	200
Additional Accessories	Checkbox	additional_accessories	300
Accessories Details	Multi Line Text	accessories_details	400

3. After adding all variables, click **Save** on the catalog item form.

The screenshot shows the ServiceNow Catalog Item - Laptop Request form. At the top, there are standard navigation buttons like Copy, Try It, Update, and Delete. Below that is a section for Related Links with links to Item Diagnostic, Show VA render type, and Run Point Scan. A tab bar at the bottom includes Variables (4), Variable Sets, Catalog UI Policies (1), Catalog Client Scripts, Available For, Not Available For, Categories (1), Catalogs (1), Catalog Data Lookup Definitions, Related Articles, and Related Catalog Items. The main area displays a table titled 'Catalog Item - Laptop Request' with a 'Variables' header. The table lists four variables: 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 table has columns for Type, Question, and Order. At the bottom of the table, there are navigation arrows and a page number indicator showing 1 to 4 of 4.



Create Catalog UI Policy

- **Purpose:** To show or hide fields based on user input, and dynamically set their properties.

Steps:

1. Go to:
All → Service Catalog → Maintain Items
2. Search and open the **Laptop Request** catalog item.
3. Scroll down to **Catalog UI Policies** related list and click **New**.
4. Fill in:
 - **Short Description:** Show Accessories Details
5. Under '**When to Apply**' tab:
 - **Field:** additional_accessories
 - **Operator:** is
 - **Value:** true
6. Click **Save** (not Submit).

Catalog UI policies are similar to standard UI policies. Catalog UI policies dynamically change variables that are part of a catalog item or change how variable sets are handled. Policies can also be applied when the variables are present in a Requested Item or Catalog Task form. [More Info](#)

Applies to: A Catalog Item
Catalog item: Laptop Request
Short description: show accessories details

When to Apply: Script

Catalog UI policy actions are applied only if all the following conditions are met:

1. The catalog UI policy is Active
2. The items in the Conditions field evaluate to true
3. The field specified in the catalog UI policy is present on the specified catalog item

Catalog Conditions: Add Filter Condition | Add OR Clause

additional accessories is true AND

Applies on a Catalog Item view: Applies on Catalog tasks: Applies on Requested Items:

Apply the catalog UI policy actions when the form is loaded or when the user changes values on the form
On load:

Reverse the effects of the catalog UI policy actions when the Conditions evaluate to false

Create Catalog UI Policy Action

- After saving the UI Policy:
1. Scroll down to **Catalog UI Policy Actions** related list and click **New**.
 2. Fill in:
 - **Variable Name:** accessories_details
 - **Order:** 100
 - **Mandatory:** True
 - **Visible:** True
 3. Click **Save**, then **Save** the UI Policy form again.

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

Update | Delete

Create UI Action

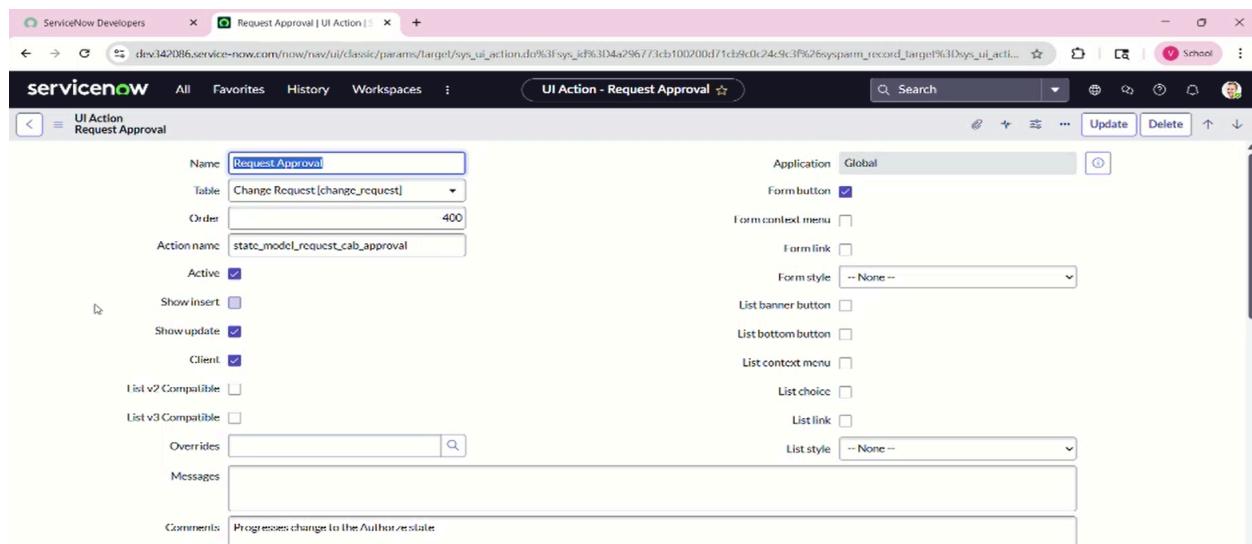
- **Purpose:** To provide users with a button that clears the form (Reset functionality).

Steps:

1. Navigate to:
All → System Definition → UI Actions
2. Click **New**.
3. Fill in the following details:
 - **Table:** sc_cart (Shopping Cart)
 - **Order:** 100
 - **Action Name:** Reset form
 - **Client:** Checked
4. Add the following script:

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

5. Click **Save**.



Export Update Set

- **Purpose:** To transfer the changes to another ServiceNow instance.

Steps:

1. Navigate to:
All → Local Update Sets
2. Open your update set: Laptop Request Project
3. Set the **State** to Complete.
4. Scroll down to see related **Update** list (all changes recorded).
5. Click **Export to XML** – this downloads an XML file.

Retrieve Update Set on Another Instance

- **Purpose:** To apply the update set on a different ServiceNow instance.

Steps:

1. **Open** the target ServiceNow instance in **Incognito Mode**
– This prevents session conflicts and ensures a clean login.
2. **Login** using the appropriate administrator credentials.
3. **Navigate** to:
All → **System Update Sets** → **Retrieved Update Sets**
– This section lists all update sets imported from external sources.
4. Click on **Import Update Set from XML** at the top.
5. **Upload** the .xml file that was exported from the source instance.
– This file contains all customizations from the original environment.
6. Once the file is uploaded:
 - Open the update set named **Laptop Request Project** from the list.
 - Click **Preview Update Set**
→ This checks for any issues, such as missing dependencies or collisions.
 - Click **Commit Update Set**
→ This will apply all the configurations and updates to the current instance.
 - **Verify** the related updates tab to confirm that all components (like forms, scripts, catalog items) are successfully added.

Testing the Catalog Item

- **Purpose:** To validate that the catalog item behaves as expected.

Steps:

1. Navigate to:
All → Service Catalog → Catalog
2. In the catalog, select the **Hardware** category.
3. Locate and open the **Laptop Request** catalog item.
4. Perform the following validations:
 - **Initial State:**
Confirm that **only three variables** (fields) are visible when the form first loads.
 - **Dynamic Behavior:**
When the user ticks the "**Additional Accessories**" checkbox:
 - The "**Accessories Details**" field should appear.
 - This field should become **mandatory** (required to submit the form).
 - **Form Validation:**
Ensure that the user **cannot submit the form** without filling the Accessories Details field if the checkbox is selected.
 - **Overall Functionality:**
All conditions, visibility rules, and mandatory field settings should behave according to the configured **Catalog UI Policies**.

7. FUNCTIONAL AND PERFORMANCE TESTING

Performance Testing

Testing ensured that:

- The form only displayed three fields initially
- Ticking the 'Additional Accessories' checkbox revealed the 'Accessories Details' field
- If the checkbox is selected and the accessory details field is left empty, the form cannot be submitted
- Reset button correctly clears all fields and shows an alert message

Testing was performed in multiple user scenarios to ensure robustness and compliance with specifications.

The screenshot shows a ServiceNow web interface for a 'Laptop Request' form. The top navigation bar includes 'ServiceNow Developers', 'Laptop Request | ServiceNow', and other browser tabs. The main menu has links for 'All', 'Favorites', 'History', 'Workspaces', and 'Admin'. A search bar is at the top right. The page title is 'Laptop Request'. The main content area contains a form with the following fields:

- 'Use this item to request a new laptop'
- 'Laptop Model' input field
- 'justification' input field
- A checked checkbox labeled 'Additional Accessories' with a tooltip 'Accessories Details'.
- A sidebar on the right titled 'Order this Item' with dropdowns for 'Quantity' (set to 1) and 'Delivery time' (set to 2 Days). It features a prominent blue 'Order Now' button, an 'Add to Cart' button, and a 'Shopping Cart' section with an 'Empty' link.

The 'Accessories Details' field is highlighted with a light purple background, indicating it is the focus of the test.

8. ADVANTAGES & DISADVANTAGES

Advantages

- Faster form submission and processing
- Elimination of manual request errors

- Easy customization for future form fields
- Portable deployment via update sets
- Friendly user interface

Disadvantages

- Requires knowledge of ServiceNow to maintain or modify
- Deployment requires administrator access

9. CONCLUSION

The project successfully implements a structured laptop request system using ServiceNow, addressing common bottlenecks and inefficiencies. With dynamic UI behavior, easy deployment options, and solid validation, it improves the service delivery process for both users and IT administrators.

10. FUTURE SCOPE

- Add approval workflows to validate laptop requests by team leads or managers
- Integrate notifications and alerts for submitted or approved requests
- Include analytics and dashboard features for request volume monitoring
- Expand the form to accommodate mobile devices and other hardware categories

11. APPENDIX

Source Code (UI Action Script)

```
function resetForm() {  
    g_form.clearForm();  
    alert("The form has been reset.");  
}
```