Laptop Request Catalog Item Final Report

1. INTRODUCTION

1.1 Project Overview
The Laptop Request Catalog Item project aims to transform the manual and delay-prone laptop request process into a dynamic, efficient ServiceNow Service Catalog experience. It provides role-based recommendations, dynamic form behavior, and automated workflows to improve accuracy and reduce administrative burden.

1.2 Purpose
This project ensures faster employee onboarding, accurate device provisioning, and reduced IT workload by leveraging ServiceNow's low-code platform to automate and streamline laptop requests.

2. IDEATION PHASE

2.1 Problem Statement
Employees and IT teams struggle with unclear specifications, lack of role-based guidance, and inefficient approval processes when requesting laptops, leading to delayed onboarding and mismatched devices.

2.2 Empathy Map Canvas
• Says: "I'm lost in options, I need guidance."
• Thinks: "Will I get the right device?"
• Does: Manually submits requests, follows up via email.
• Feels: Anxious, frustrated.
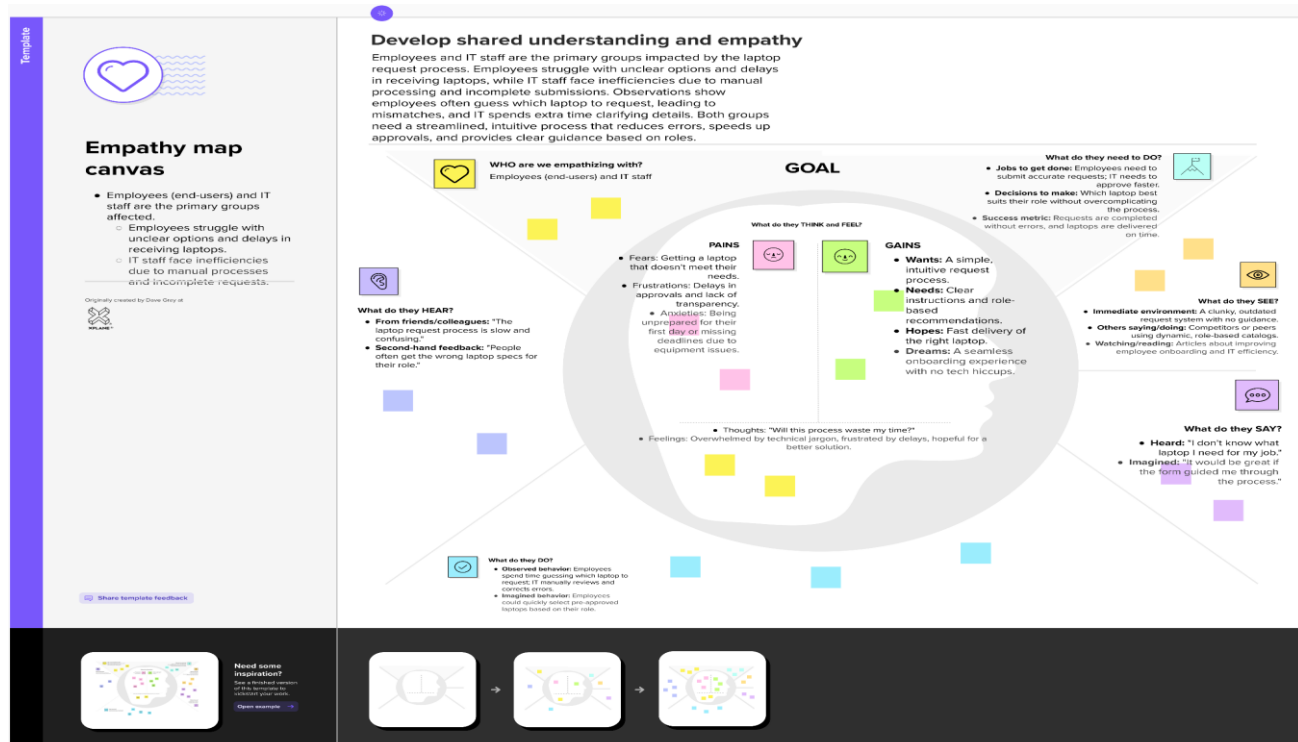
2.3 Brainstorming
The team explored dynamic UI policies, role-based suggestions, approval workflows, inventory integration, and reset functionality to capture all improvement ideas before prioritizing.
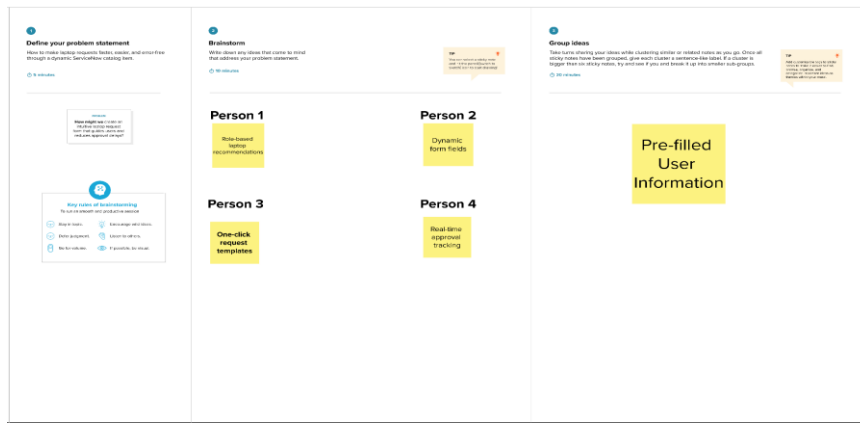
# 3. REQUIREMENT ANALYSIS

3.1 Customer Journey Map

Users access the form, select role-based options, submit the request, receive notifications, and track status in real time.

3.2 Solution Requirement

- Custom catalog item with variables
- Dynamic field visibility and validation
- Approval workflow with comments
- Email notifications
- CMDB integration for inventory checks
- Real-time status tracking

3.3 Data Flow Diagram

Data flows from the user interface through validation scripts to ServiceNow tables, triggering workflows, notifications, and status updates.
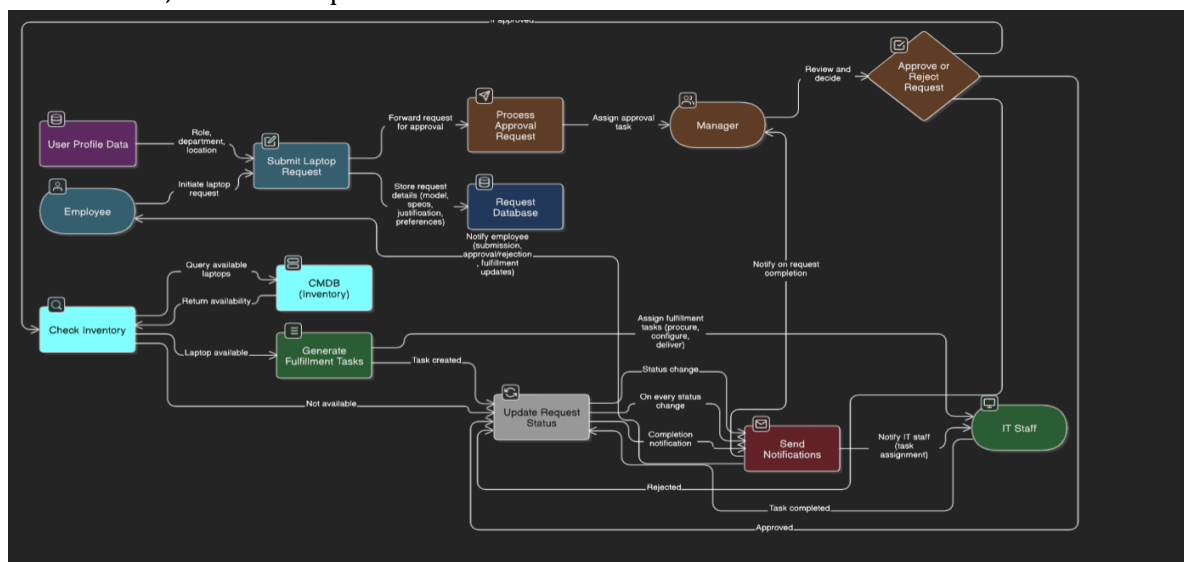
3.4 Technology Stack

- UI: ServiceNow UI Framework (Jelly, AngularJS)
- Logic: Glide Script, Flow Designer
- Database: ServiceNow CMDB
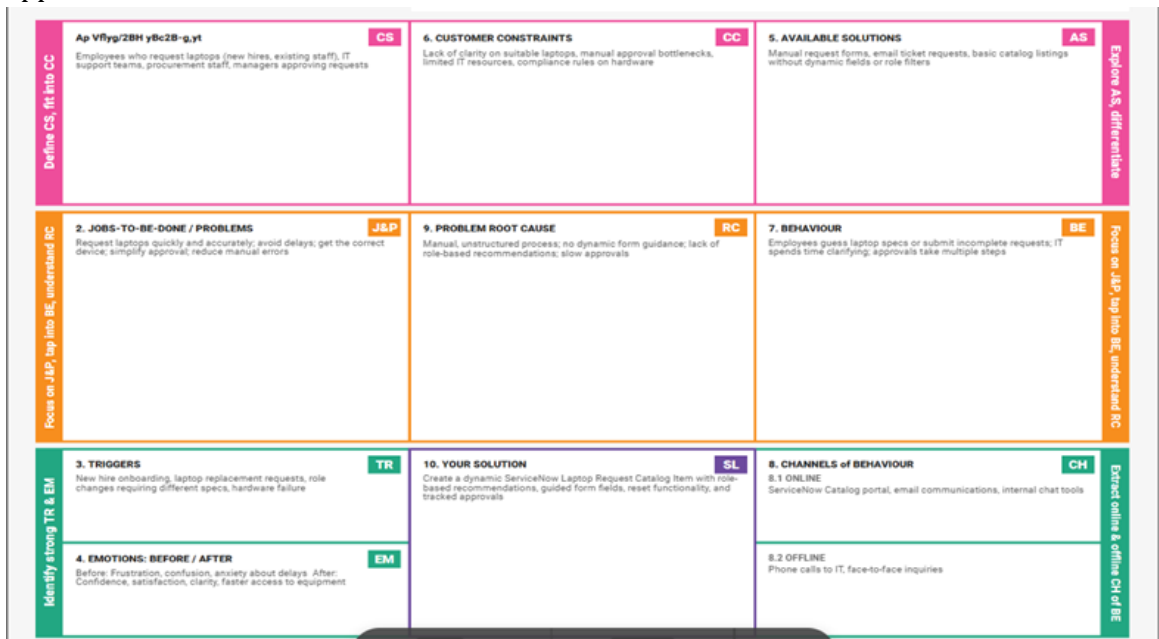- Integration: REST APIs

• Notifications: ServiceNow Notification Engine
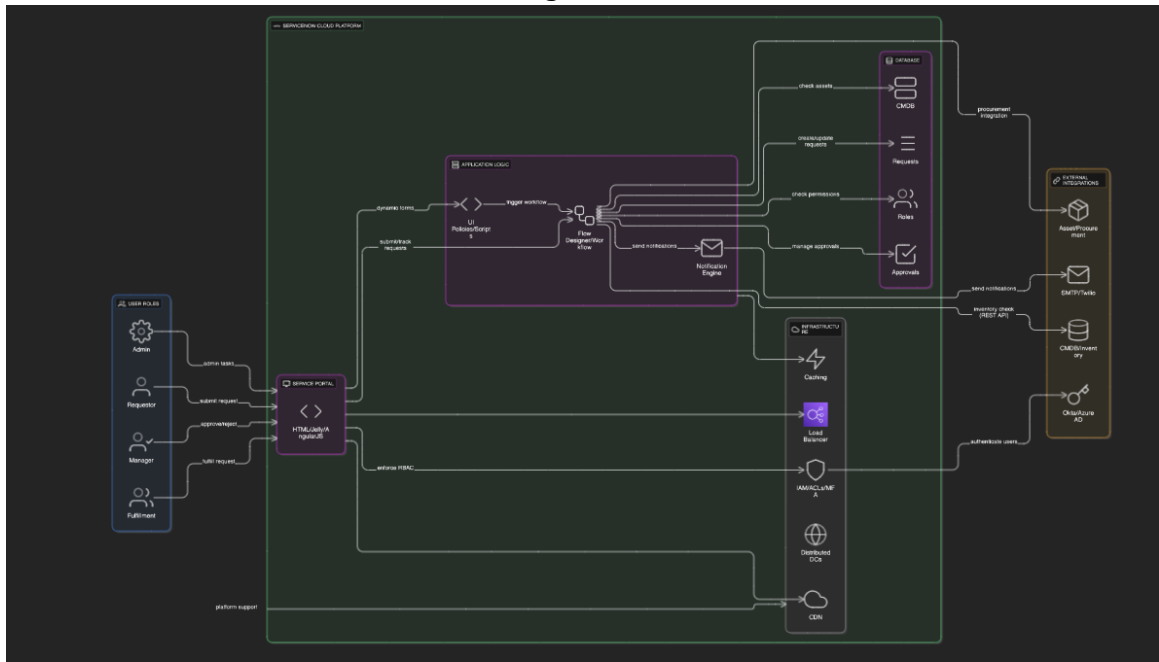
4. PROJECT DESIGN

4.1 Problem–Solution Fit

The solution directly addresses delays and errors by guiding users with dynamic fields and automating approvals.


4.2 Proposed Solution

A ServiceNow Service Catalog item that provides:
• Role-based laptop recommendations
• UI policies for guided form behavior
• Automated approval workflows
• Inventory validation via CMDB
• Reset and export functionality

4.3 Solution Architecture

The architecture comprises:
• UI Layer: Catalog forms with variables and policies
• Logic Layer: Client scripts, business rules, workflow engine
• Data Layer: Task, Request, CMDB tables
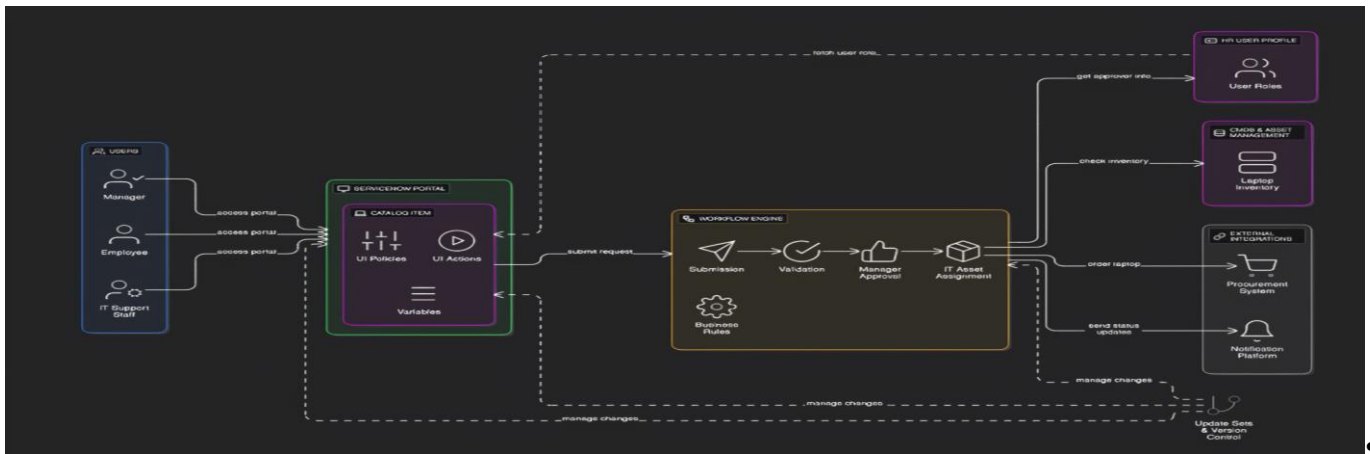
Integration Layer: REST API calls to CMDB

5. PROJECT PLANNING & SCHEDULING

5.1 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.

5.2 Creation of 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'

5.3 Adding variables:

Step1:
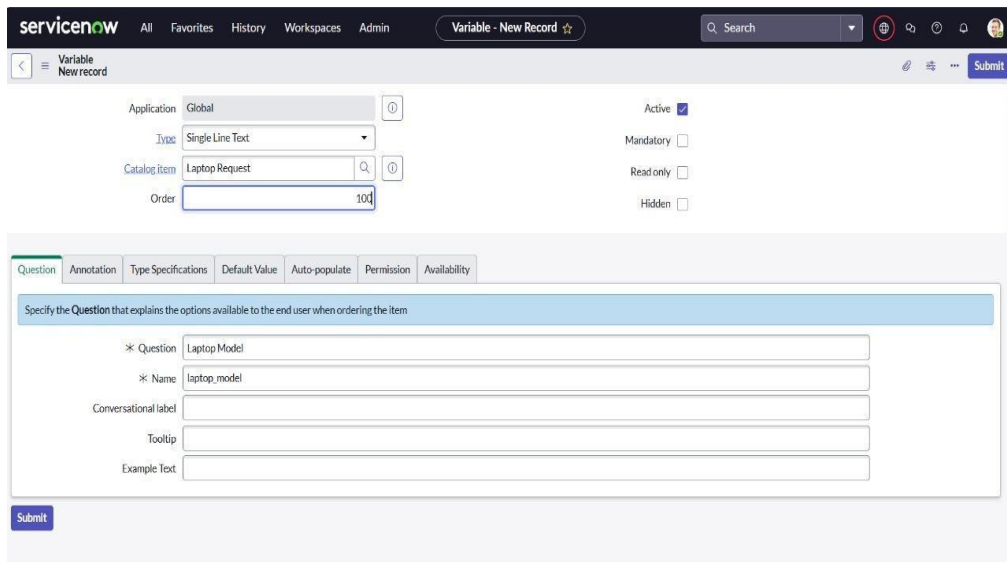
- After saving the catalog item form scroll down and click on variable(related list)

- 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
Variable 2:Justification

Type: Multi line text Name:

justification Order:200

2. Variable 3:Additional

   Accessories Type: Checkbox

   Name: additional_accessories

   Order:300

3. 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
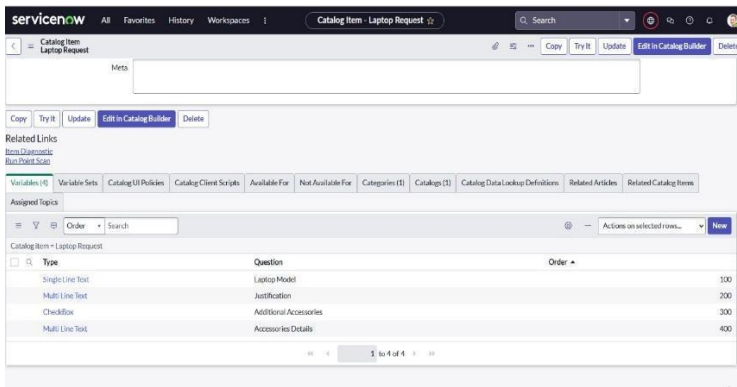
5.4 Create Catalog Ui policies:

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:10

    0 Mandatory:

True Visible :

　　　　　True

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


5.5 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

5.6 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
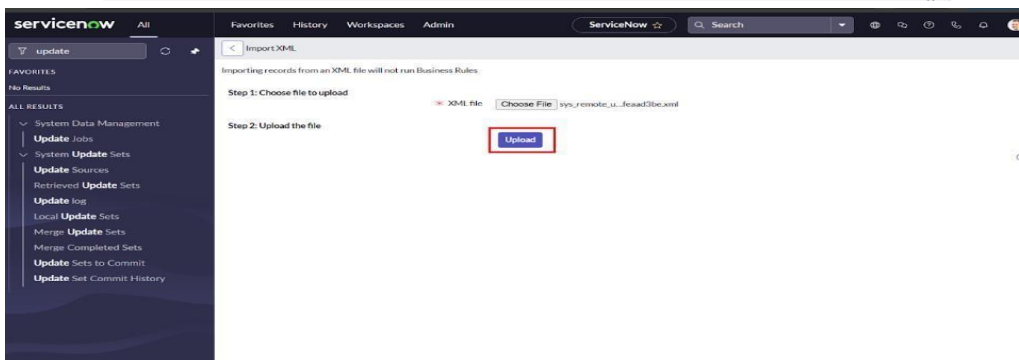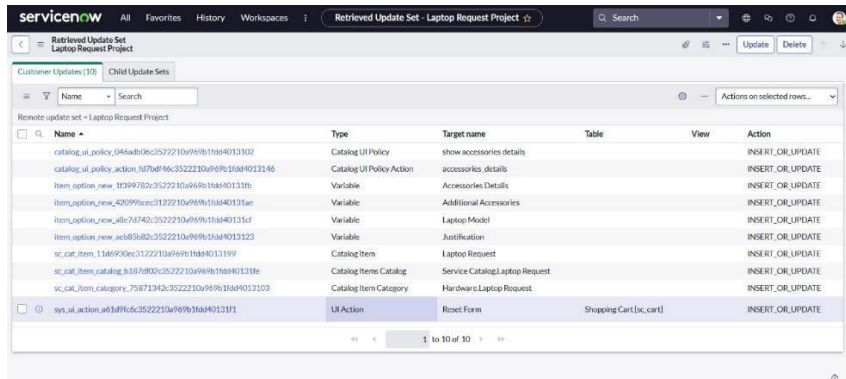
7. Upload the downloaded file in XML file
8. Open retrieved update set 'laptop request project'

9. Click on preview update set
10. And click on commit update set
11. And also see the related tab updates
12. After committing update set in this instance we get all updates which are done in the previous instance.
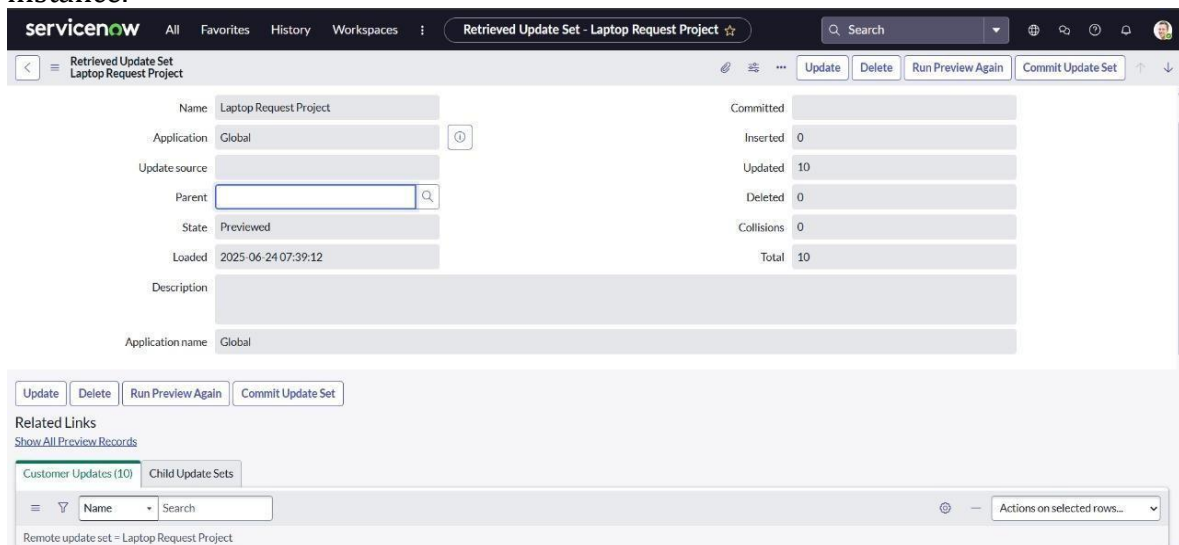
5.7 Test Catalog Item:

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.

1. As per our scenario, when we click on additional accessories checkbox then accessories details fields is visible and that should be mandatory
2. Now see the results, it fulfills our requirements.

6. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing

Validated auto-numbering, field visibility logic, workflow triggers, notification delivery, and status tracking without latency.

7. RESULTS

7.1 Output Screenshots

Screenshots of form, dynamic behavior, and CMDB integration verify successful implementation.

8. ADVANTAGES & DISADVANTAGES

Advantages:
• Streamlined requests
• Reduced errors
• Scalable framework
Disadvantages:
• Requires ServiceNow expertise
• Depends on platform licensing

9. CONCLUSION

The project delivered a robust ServiceNow solution that automates and enhances the laptop request process, improving efficiency and user satisfaction.

10. FUTURE SCOPE

• Mobile-friendly interface
• Advanced reporting dashboards
• Integration with HR and asset management systems
• Bulk request and analytics features