**Solution Architecture**

|  |  |
| --- | --- |
| Date | 28/10/2025 |
| Team id | NM2025TMID03568 |
| Project name | Laptop Request Catlog Item |
| Maximum mark | 5 Marks |

This document outlines the high-level architecture for the **Laptop Request Catalog Item**. It shows the key components within the ServiceNow platform and how they interact to automate the request-to-fulfillment process.

**1. Key Components**

* **Service Portal:** The user-facing interface where an employee (requester) finds and submits the "Laptop Request" form.
* **Catalog Item (sc\_cat\_item):** The form itself. It is configured with:
  + **Variables:** To capture user input (e.g., *Laptop Model*, *Justification*, *Software Needed*).
  + **UI Policies / Client Scripts:** To control form behavior (e.g., make *Justification* mandatory, show laptop specs based on selection).
* **Flow Designer (Workflow Engine):** This is the core automation component. It is a no-code/low-code tool that executes the entire backend process, including approvals, tasks, and notifications.
* **ServiceNow Tables (Data Model):** The database layer that stores all records generated by the process. The key tables are:
  + sc\_request (Request)
  + sc\_req\_item (Requested Item)
  + sc\_task (Catalog Task)
  + sysapproval\_approver (Approval)
* **Notification Engine:** The built-in ServiceNow module that sends automated email notifications to the user and the manager at key stages.

**2. Process Flow Diagram**

This diagram shows the end-to-end flow of the solution, from user submission to fulfillment.

[Actor] (Interface) [Action] (System Component)

---------------------------------------------------------------------------------------

[Requester] -> (Service Portal) -> [Submits Laptop Form]

|

V

(Flow Designer) ----> [1. Creates Request (REQ)]

|

V

(Flow Designer) ----> [2. Creates Requested Item (RITM)]

|

V

(Flow Designer) ----> [3. Looks up User's Manager]

|

V

(Flow Designer) ----> [4. Creates Approval Record]

|

V

[Manager] -> (Email or Portal) -> [Approves Request]

|

V

(Flow Designer) ----> [5. Checks Approval Status: 'Approved']

|

V

(FlowDesigner) ----> [6. Creates Catalog Task (SCTASK)]

|

V

(Flow Designer) ----> [7. Assigns Task to 'IT Hardware Team']

|

V

[IT Team] -> (Platform UI) -> [Completes Task]

|

V

(Flow Designer) ----> [8. Closes RITM as 'Complete']

|

V

(Notification Engine) -> [Sends 'Completed' Email to Requester]

**3. Data Model**

The solution generates three primary records for each request, linked in a parent-child relationship:

1. **Request (sc\_request):** The "wrapper" or "shopping cart" for the order. This record has a REQ number (e.g., REQ0010001).
2. **Requested Item (sc\_req\_item):** This is the *actual* laptop request. It has an RITM number and stores all the data from the form variables (e.g., "MacBook Pro," "Business Justification..."). The workflow is attached to this record.
3. **Catalog Task (sc\_task):** The "to-do" record for the IT Hardware team (e.g., "Prepare and provision MacBook Pro for User X"). It has an SCTASK number and is a child of the RITM.

**Relationship:** 1 (REQ) -> 1 (RITM) -> 1 (SCTASK)