Automated Network Request Management in ServiceNow

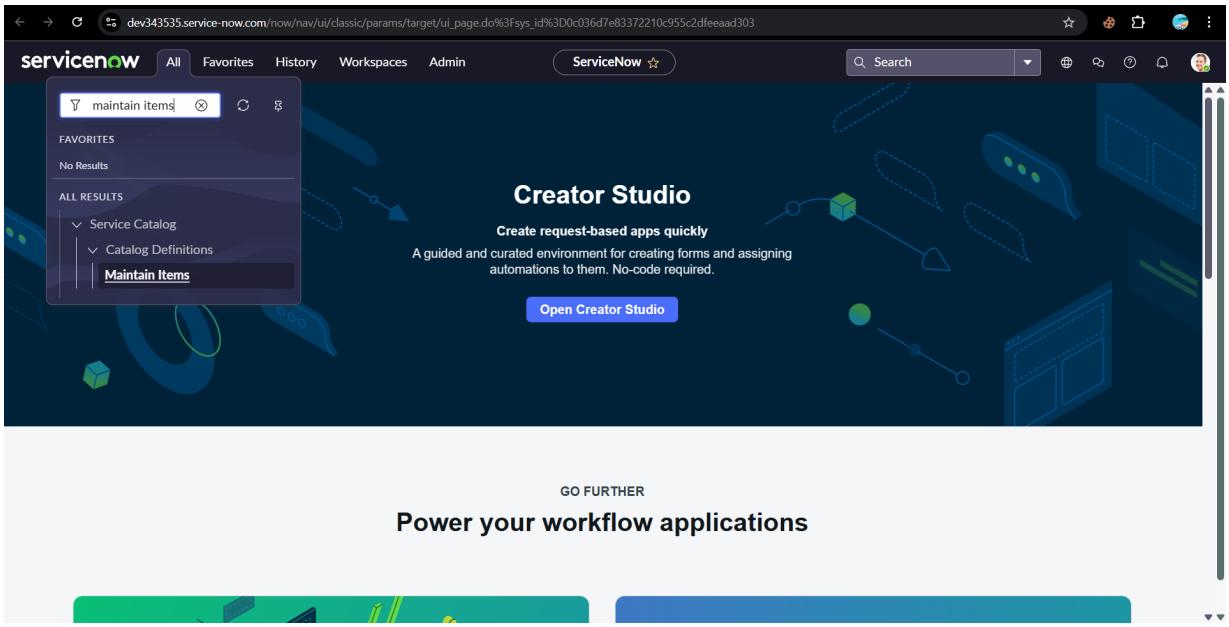
INTRODUCTION:

This project provides an automated solution in **ServiceNow** to manage network-related service requests. Through a self-service portal, users can easily submit requests, which are then validated, approved, and routed for fulfillment. Automated workflows handle approvals, notifications, and task assignments, while optional integrations with network tools reduce manual effort. The system also offers real-time updates and reporting to improve efficiency, transparency, and SLA tracking.

Process 1: Creation of Service Catalog – "Network Request"

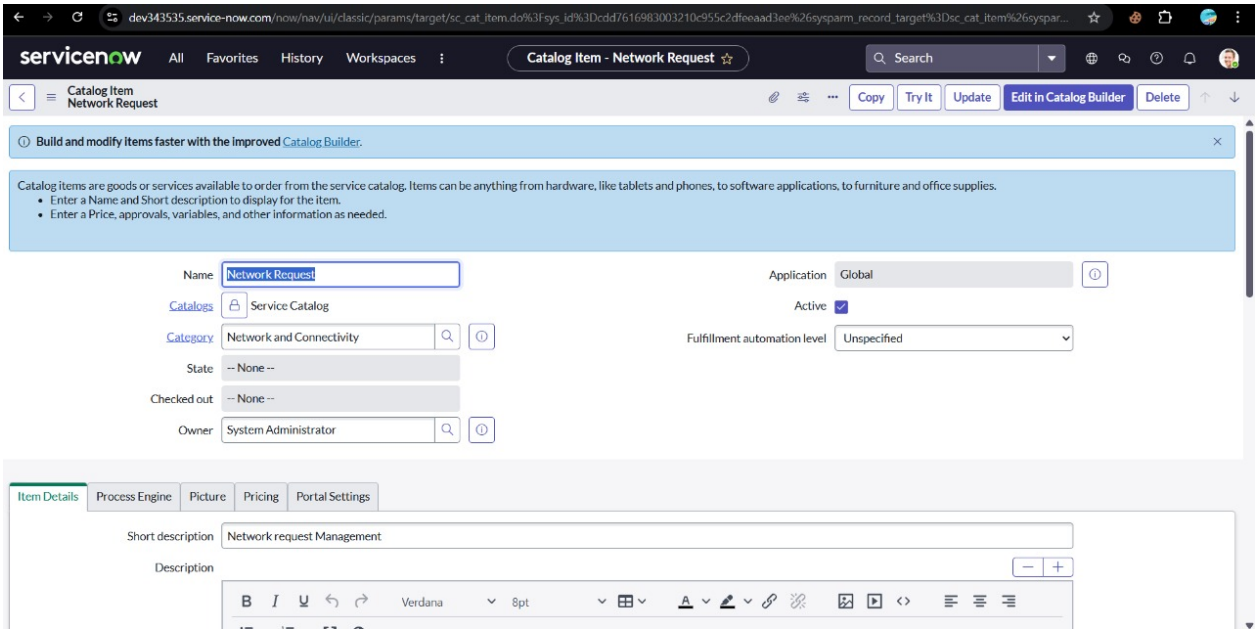
Step 1: Navigate to Service Catalog

1. Open the **Application Navigator** in ServiceNow.
2. Go to:
   All → Service Catalog → Maintain Items



Step 2: Create New Catalog Item

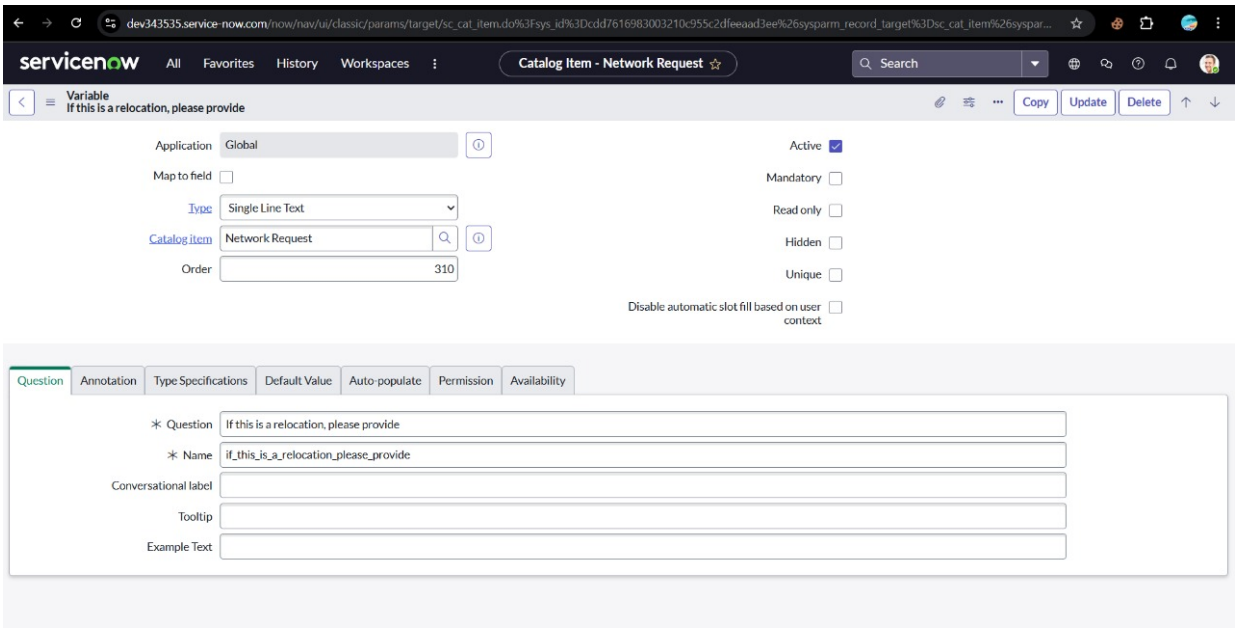
1. Click on **New**.
2. Fill the following details:
    - **Name**: Network Request
    - **Catalog**: Service Catalog
    - **Category**: Network and conncetivity
    - **Short Description**: Network Request Management
3. Click on **Save**.



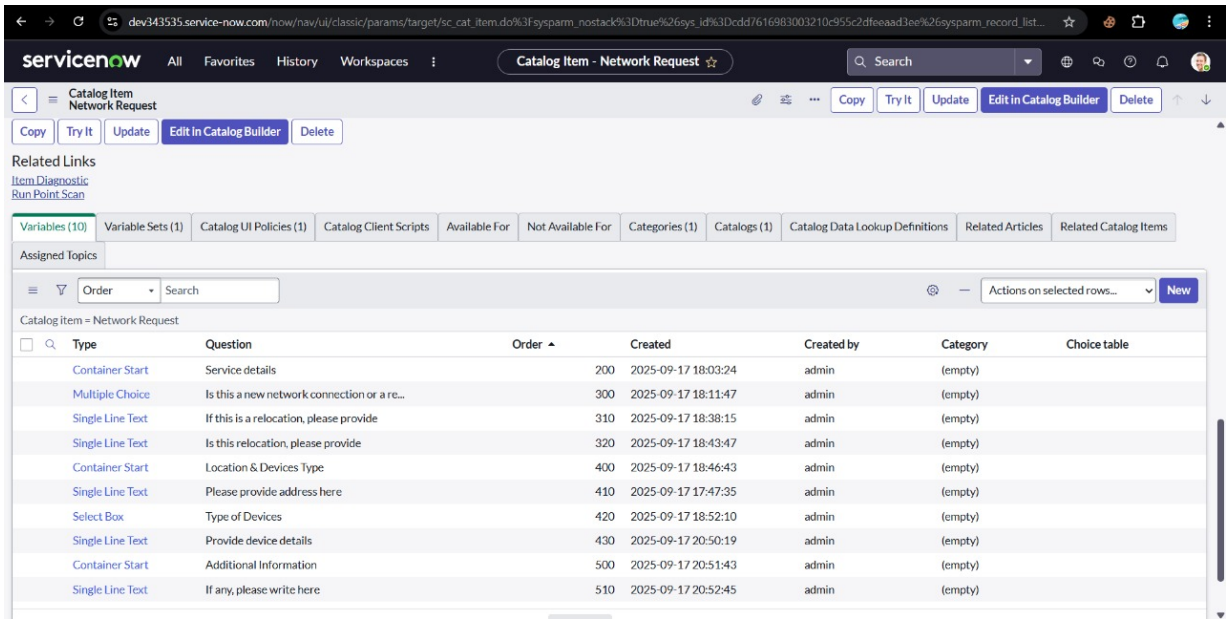
Step 3: Configure Variables

1. Open the newly created **Network Request** catalog item.
2. Scroll down to the **Variables** related list → Click **New** for each variable.
3. Fill out the following for each variable:
    - **Type**: Single line text, Choice, Reference, etc.

- **Order**: e.g., 100, 200, 300 (controls display order)
- **Question**: Label shown on the form
- **Name**: Technical name (used in scripts)
- **Tooltip**: Info shown on mouse hover
- **Example Text**: Placeholder help text
- **Mandatory / Read-Only**: As required
- **Auto-populate**: Use dot-walking for dependent values



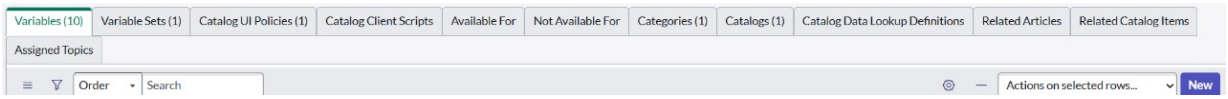
Type	Question	Order ▲
Container Start	Service details	200
Multiple Choice	Is this a new network connection or a re...	300
Single Line Text	If this is a relocation, please provide	310
Single Line Text	Is this relocation, please provide	320
Container Start	Location & Devices Type	400
Single Line Text	Please provide address here	410
Select Box	Type of Devices	420
Single Line Text	Provide device details	430
Container Start	Additional Information	500
Single Line Text	If any, please write here	510



Step 5: Configure Variable Set – Requester Information

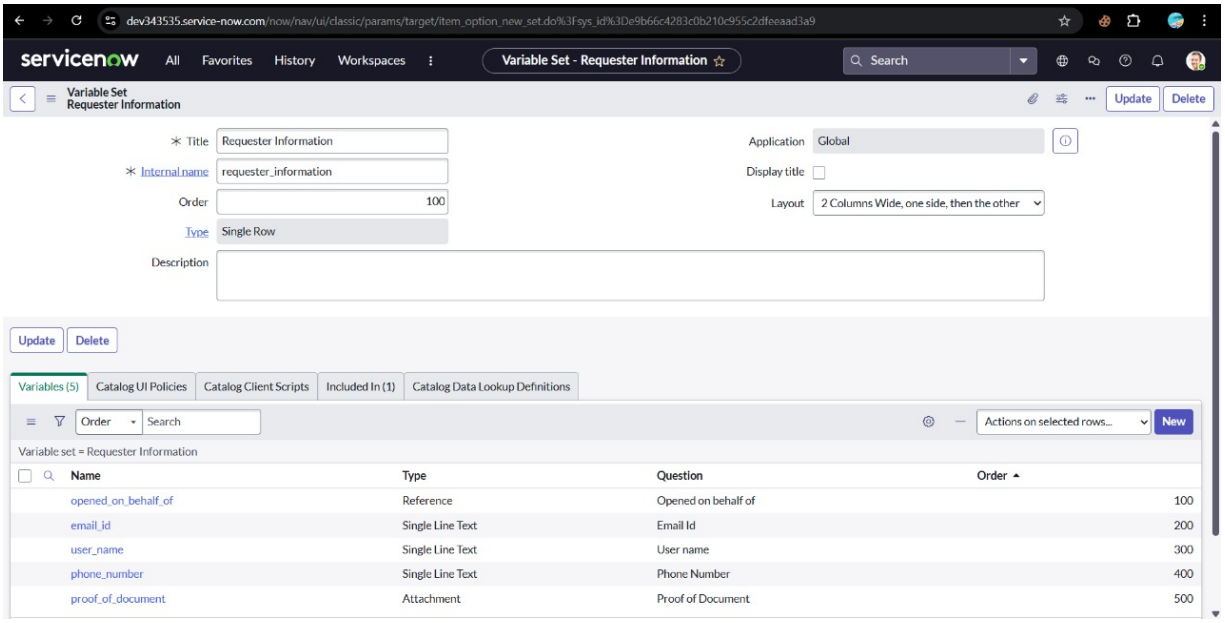
5.1 Create Variable Set

1. Navigate to **Variable Sets** under Service Catalog.



2. Click on **New.**
3. Fill the following details:

- **Title**: Requester information
- **Internal Name**: requester_information (auto-filled)
- **Order**: 100
- **Type**: Single Row
- **Layout**: 2 Columns Wide, one side, then the other
- Check the box: **Display title**



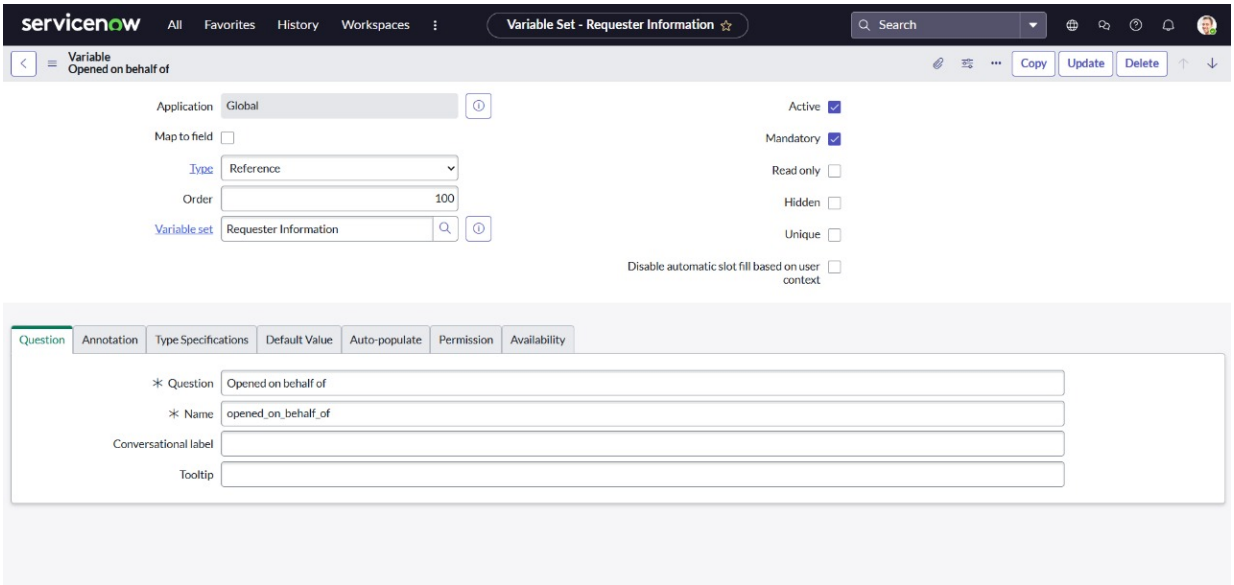
Step 5.2: Add Variables to the Variable Set "Requester Information"

After creating the variable set, now it's time to add the variables one by one.

1.Opened on behalf of

- Type: **Reference**
- Reference to: **User [sys_user]**

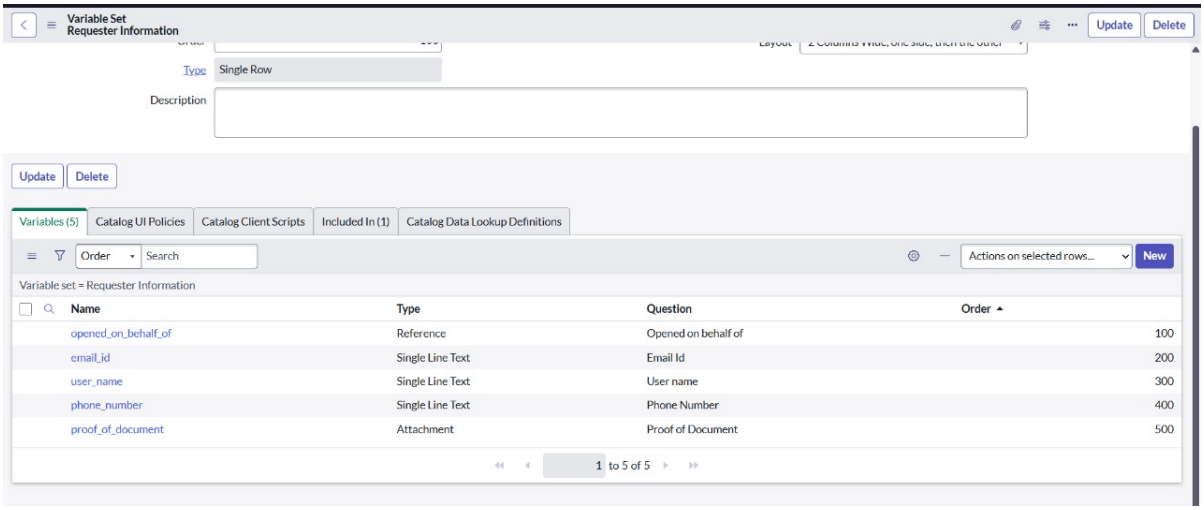
- Name: opened_on_behalf_of
- Order: 100
- This allows the requester to select a user they are raising the request for.



2. **Email ID**
    - Type: **Single Line Text**
    - Name: email_id
    - Order: 200
    - This will be auto-filled based on the user selected in "Opened on behalf of".
    - You can use a script or dot-walking to populate the email field.
3. **User Name**
    - Type: **Single Line Text**
    - Name: user_name
    - Order: 300
    - This will also be auto-populated based on the user selected.
    - Fetch the full name from the User table.
4. **Phone Number**
    - Type: **Single Line Text**
    - Name: phone_number
    - Order: 400

- Same as above, it can be fetched using dot-walking or client script.

5. **Proof of Document**
   - Type: **Attachment**
   - Name: proof_of_document
   - Order: 500
   - This allows users to upload a file (such as proof or ID documents).



When a user is selected in the **Opened on behalf of** field, we want to automatically populate:

- Email ID
- User Name
- Phone Number

Steps to Auto-populate Fields

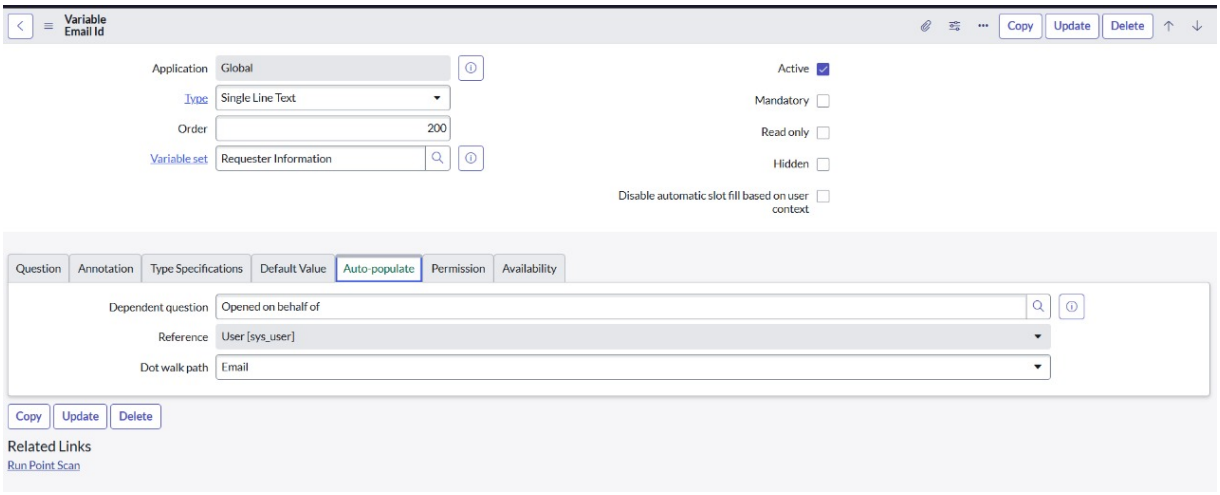
1.Open the Variable Set

- Navigate to: **Service Catalog > Catalog Variable Sets**
- Open your variable set: **Requester Information**

2.Create a Catalog Client Script

- Navigate to: **Service Catalog > Catalog Client Scripts**
- Click **New**
- Fill in details:

- **Name**: Auto Populate User Info
- **Applies to**: Catalog Item
- **Variable Set**: Select *Requester Information*
- **UI Type**: All
- **Type**: onChange



Step 6: Catalog UI Policy Configuration

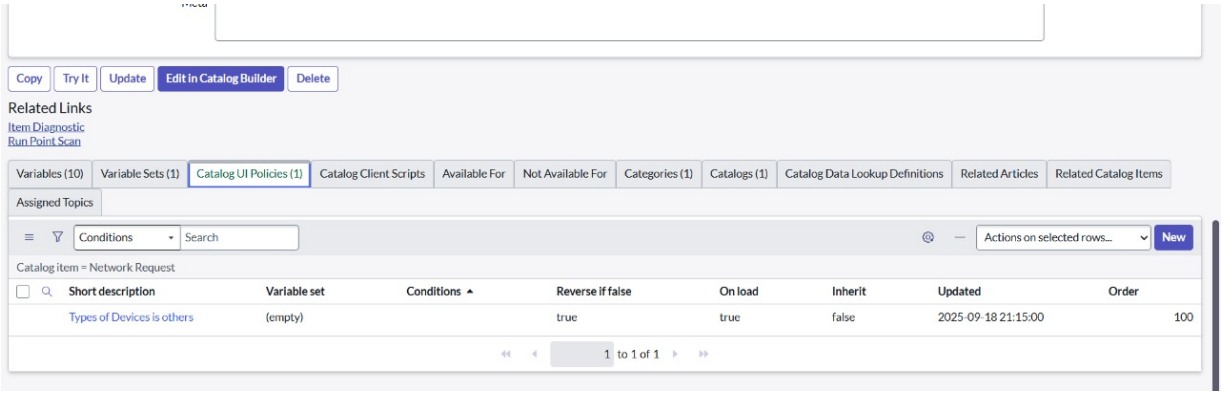
Goal: Show " Provide device details here " field when **Types of Devices = Others**.

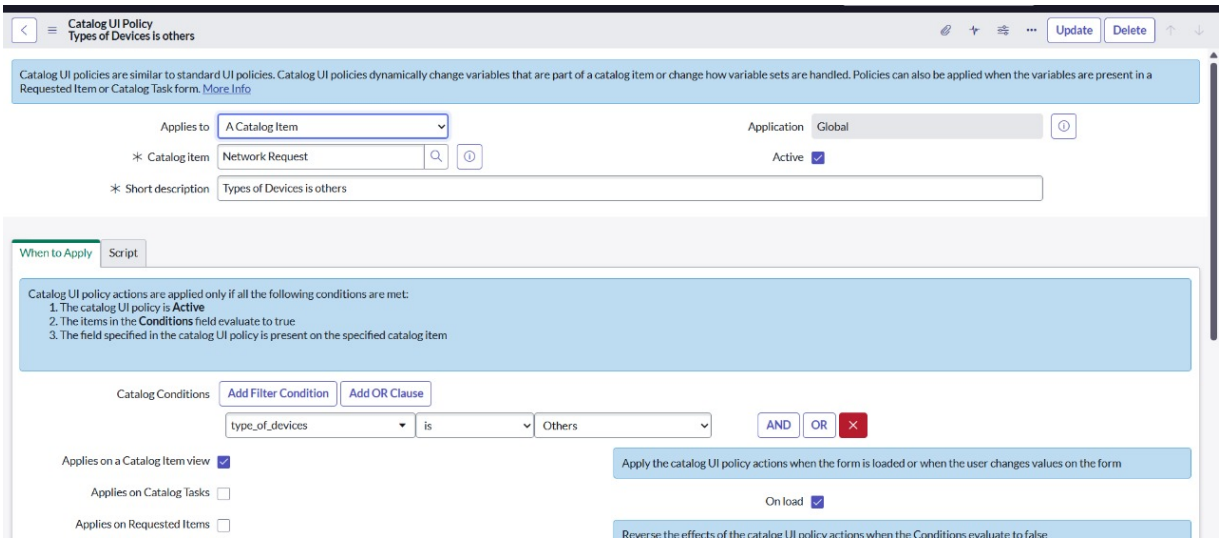
1. Navigate to the **Network Request** catalog item.
2. In the related list, go to **Catalog UI Policies** → Click **New**.
3. Fill in:
   - **Applies to**: Catalog Item
   - **Catalog Item**: Network Request
   - **Condition**: Types of devices is Others
4. Click **Save**.
5. In the related list, click **New** under **UI Policy Actions**.
6. Set:

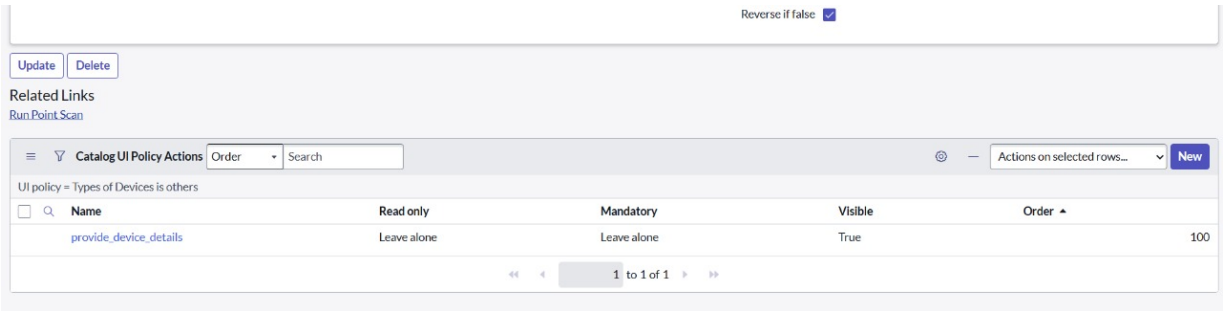
- **Catalog Item**: Network Request
- **Variable name**: Provide device details here
- **Visible**: True

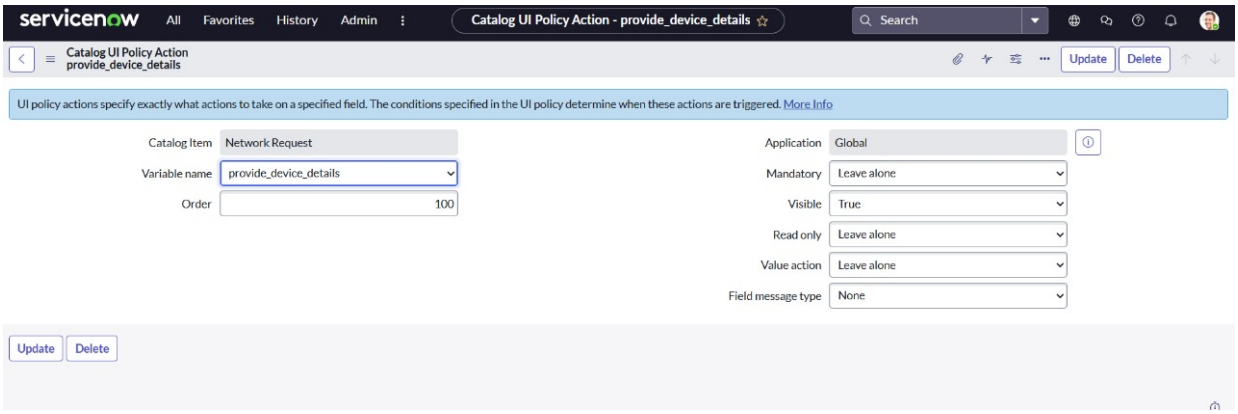
7. Click **Update** to save policy.

8. **Test the form** to ensure the field appears based on selection.







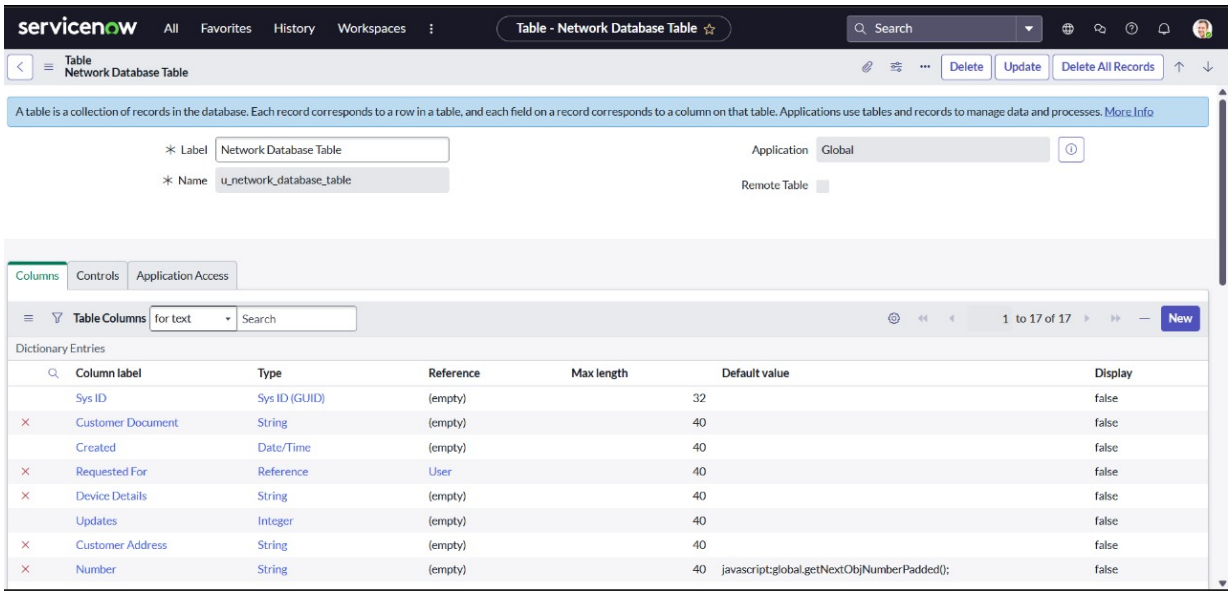


Process 2: Creation of Table and Fields in ServiceNow

Network Database Table

Step 1: Create a New Table

1. Navigate to the Application Navigator.
2. Type: Tables under the **System Definition** module.
3. Click on **Tables**.
4. On the top-right corner, click on **New** to create a new table.
5. Fill in the table details:
    - **Label**: *Network Database Table*
    - **Name**: Automatically generated (or customize if needed).
    - Keep **Auto-generate schema** checked.
6. Click **Submit** to create the table.



Step 2: Add custom fields

These fields are **custom fields** that you will manually add in the Table Columns section of your custom table.

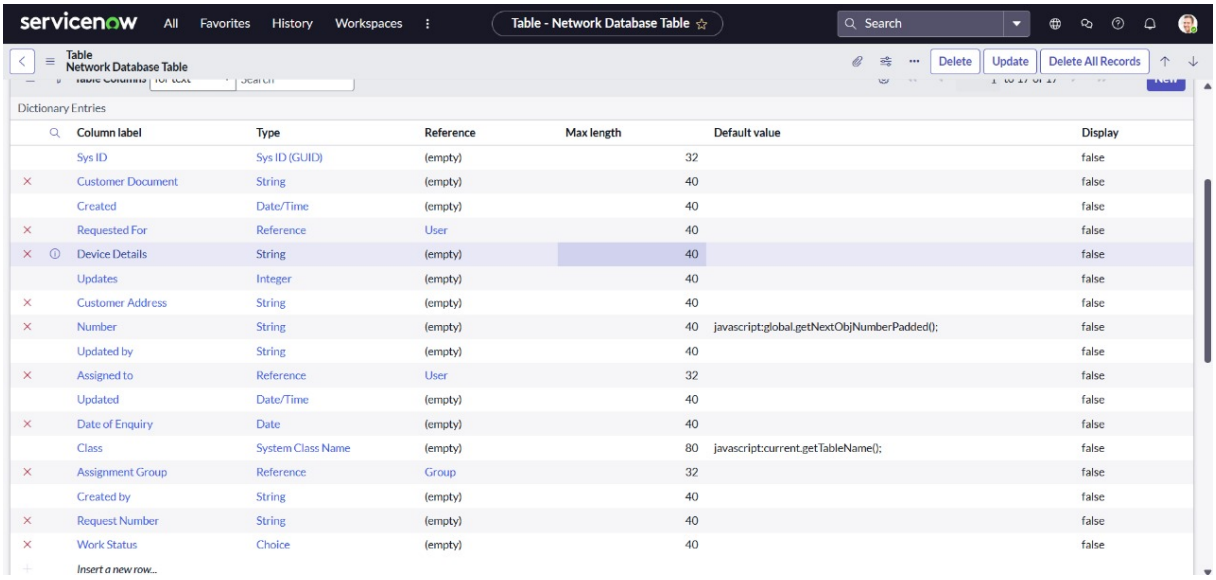
1. **Name: u_request_number**
    - **Label**: Request Number
    - **Type**: String
    - **Reference**: —
    - **Explanation**: A unique identifier for the request. Can be filled manually or auto-generated using a Business Rule.
2. **Name: u_assignment_group**
    - **Label**: Assignment Group
    - **Type**: Reference
    - **Reference**: Group (Group table)

- **Explanation**: Defines the team or group responsible for fulfilling the request.

3. **Name: u_customer_document**
    - **Label**: Customer Document
    - **Type**: String
    - **Reference**: —
    - **Explanation**: Stores a document reference or identifier related to the customer, such as an ID proof or contract reference
4. **Name: u_assigned_to**
    - **Label**: Assigned To
    - **Type**: Reference
    - **Reference**: User(User table)
    - **Explanation**: The specific user assigned to handle the request.
5. **Name: u_device_details** **Label**: Device Details
    - **Type**: String
    - **Reference**: —
    - **Explanation**: Captures technical details or specifications of the device involved in the request.
6. **Name: u_date_of_enquiry**
    - **Label**: Date of Enquiry
    - **Type**: Date
    - **Reference**: —
    - **Explanation**: The date when the enquiry was received from the customer.
7. **Name: u_customer_address**

- **Label**: Customer Address
- **Type**: String
- **Reference**: —
- **Explanation**: The physical or mailing address of the customer.

8. **Name: u_approval_state**
    - **Label**: Work Status
    - **Type**: String
    - **Reference**: —
    - **Explanation**: Indicates the current approval or work status of the request.
9. **Name: u_requested_for**
    - **Label**: Requested For
    - **Type**: String *(Normally this should be a Reference to sys_user, but in your screenshot it's String)*
    - **Reference**: — *(unless you change it to a Reference type)*
    - **Explanation:** Specifies the end-user for whom the request is being made.



Column label	Type	Reference	Max length	Default value	Display
Sys ID	Sys ID (GUID)	(empty)	32		false
Customer Document	String	(empty)	40		false
Created	Date/Time	(empty)	40		false
Requested For	Reference	User	40		false
Device Details	String	(empty)	40		false
Updates	Integer	(empty)	40		false
Customer Address	String	(empty)	40		false
Number	String	(empty)	40	javascript:global.getNextObjNumberPadded();	false
Updated by	String	(empty)	40		false
Assigned to	Reference	User	32		false
Updated	Date/Time	(empty)	40		false
Date of Enquiry	Date	(empty)	40		false
Class	System Class Name	(empty)	80	javascript:current.getTableName();	false
Assignment Group	Reference	Group	32		false
Created by	String	(empty)	40		false
Request Number	String	(empty)	40		false
Work Status	Choice	(empty)	40		false
Insert a new row...					

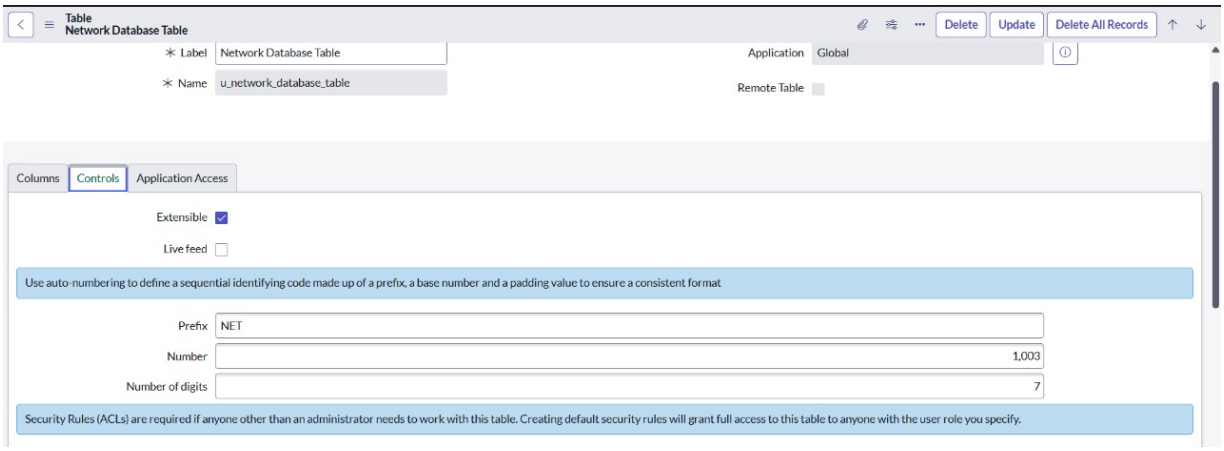
To Autopopulate Database Number

Using Number Maintenance

ServiceNow has a built-in feature called **Number Maintenance** to manage auto-number sequences for any table.

1. Navigate to:
System Definition > Number Maintenance.
2. Click **New**.
3. Fill in details:
    - **Table** → select your Network Database Table.
    - **Prefix** → NET.
    - **Current Value** → 1003 (or any starting number you want).
    - **Number of Digits** → 7.

4. Save.



Network Task Table

Step 1: Create the Child Table (Network Task Table)

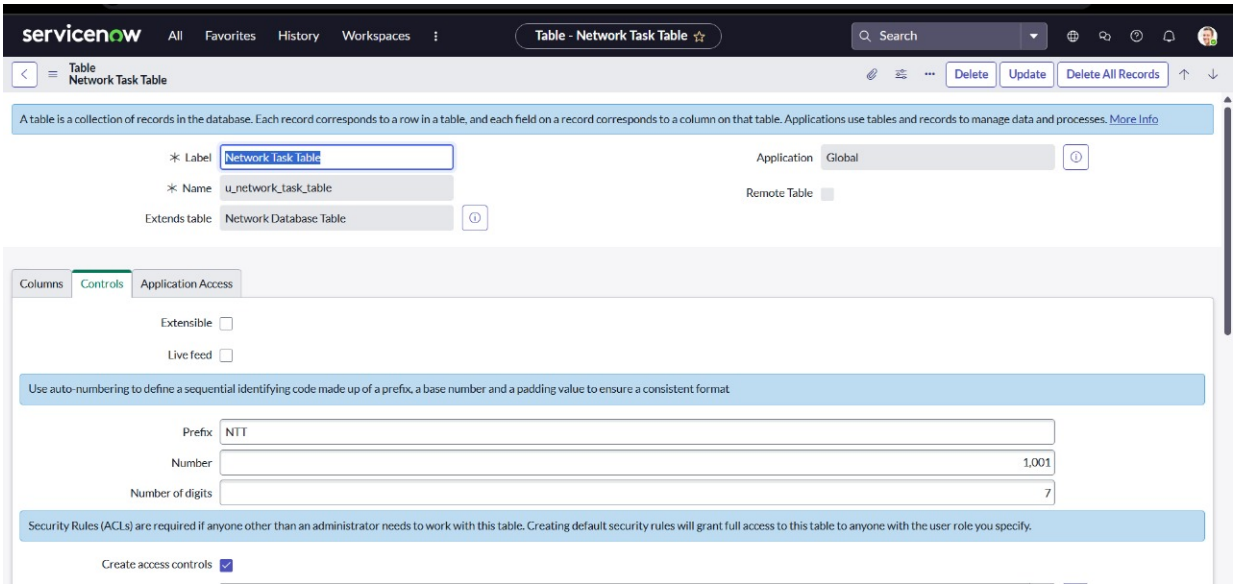
1. Navigate to:

System Definition > Tables

2. Click **New**.
3. Fill in details:
    - **Label** → Network Task Table
    - **Name** → auto-generated (u_network_task_table)
    - **Extends Table** → select **Network Database Table**
    - (u_network_database_table)

This is the important part → by choosing **Extends Table**, your Network Task Table will automatically inherit all fields from the parent.

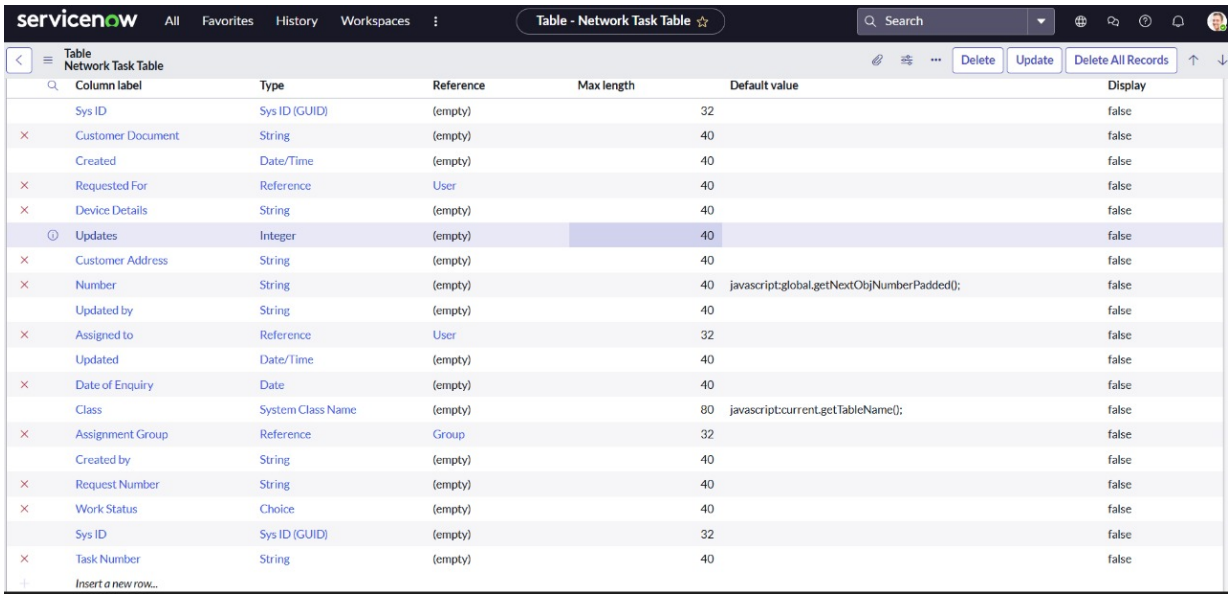
4. Save the record.



Step 2: Verify Inherited Fields

1. Open the new table (Network Task Table).
2. Go to **Columns** tab.
3. You'll see:

    - Fields from parent (Database Number, Request Number, Request For, etc.)
    - Plus any new fields you add specifically for tasks (Task Number, Work Status, Assigned to, etc.).



Column label	Type	Reference	Max length	Default value	Display
Sys ID	Sys ID (GUID)	(empty)	32		false
Customer Document	String	(empty)	40		false
Created	Date/Time	(empty)	40		false
Requested For	Reference	User	40		false
Device Details	String	(empty)	40		false
Updates	Integer	(empty)	40		false
Customer Address	String	(empty)	40		false
Number	String	(empty)	40	javascript:global.getNextObjNumberPadded();	false
Updated by	String	(empty)	40		false
Assigned to	Reference	User	32		false
Updated	Date/Time	(empty)	40		false
Date of Enquiry	Date	(empty)	40		false
Class	System Class Name	(empty)	80	javascript:current.getTableName();	false
Assignment Group	Reference	Group	32		false
Created by	String	(empty)	40		false
Request Number	String	(empty)	40		false
Work Status	Choice	(empty)	40		false
Sys ID	Sys ID (GUID)	(empty)	32		false
Task Number	String	(empty)	40		false
Insert a new row...					

Step 3: Configure Auto Numbering for Task Table

If you want separate auto numbering for **Network Tasks** (like NTT0001001):

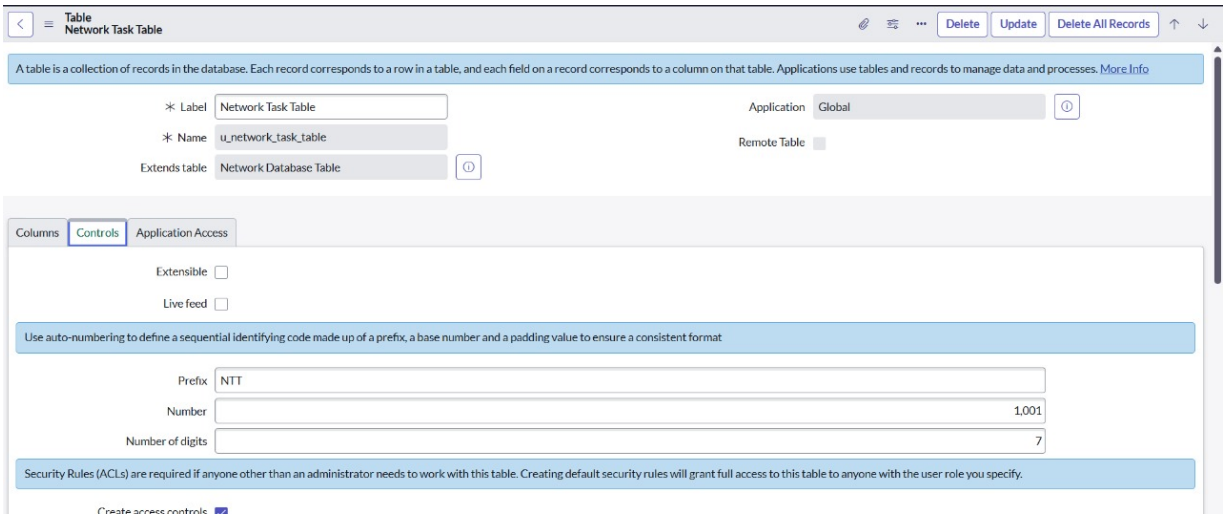
1. Navigate to **System Definition > Number Maintenance**.
2. Click **New**.
3. Fill details:
    - **Table** → Network Task Table
    - **Prefix** → NTT
    - **Current Value** → 1001
    - **Number of Digits** → 7
4. Save.

Now each task will have a unique Task Number (NTT0001001, NTT0001002 …).

Step 4: Adjust the Form Layout

1. Open a record in **Network Task Table**.
2. Right-click the header → **Configure > Form Layout**.
3. Add inherited fields (Database Number, Request Number, etc.) and new fields (Task Number, Work Notes, etc.).

Arrange as you like.



Process 3: Request Approvals Creation

The goal is to display **approval records** directly on the **Network Database table** form. By creating a relationship between **Network Database Table** and **Approval (sysapproval_approver)**:

- We can see which approvals are associated with each record.
- We avoid searching in a separate table.
- The refineQuery ensures only relevant approvals (based on source table and document ID) are shown.

Steps to Create the Related List with Script

1. **Navigate to Relationships**
   - Go to **System Definition → Relationships**.
   - Click **New**.
2. **Fill in the Relationship Details**
   - **Name** → Request Approvals
   - **Applies to table** → Network Database Table [u_user_network_database]
   - **Queries from table** → Approval [sysapproval_approver]

- **Active** → Checked.

3. Add the refineQuery Script

The script filters the approvals to only show records related to the current Network Database record. (function refineQuery(current, parent) ,

current.addQuery('source_table', parent.getTableName());

current.addQuery('document_id', parent.sys_id);

-)(current, parent);

Script Explanation:

- source_table → Ensures only approvals linked to this specific table are fetched.
- document_id → Matches the approval record to the exact parent record.
- state filter (commented out) → Can exclude approvals not required.

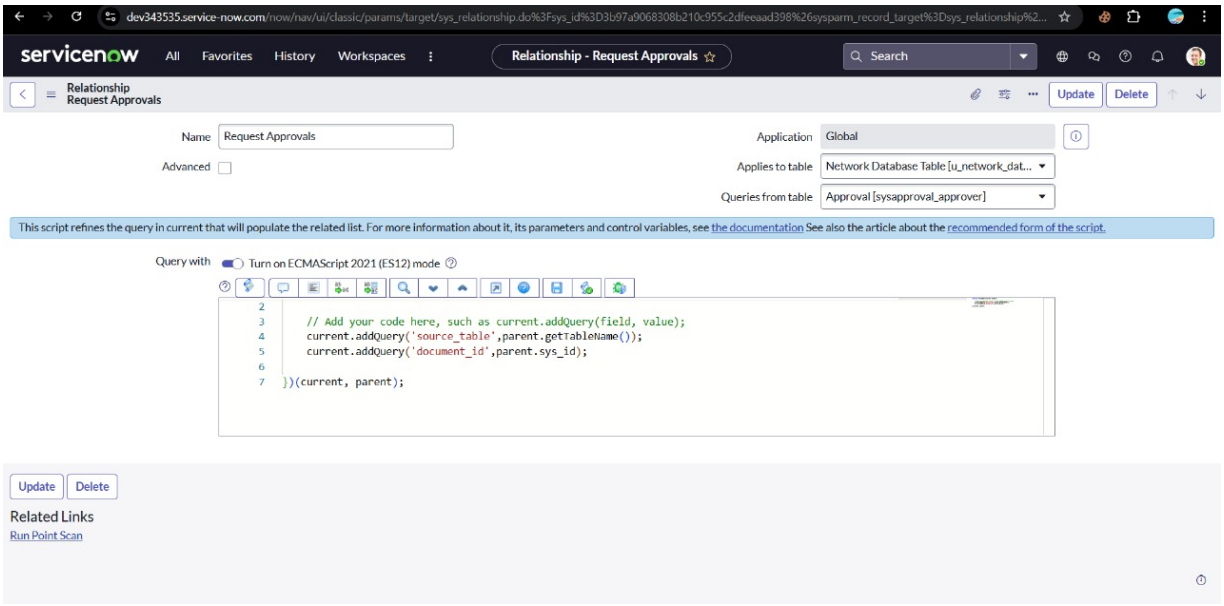
4. Save and Verify

- Click **Update**.
- Open a **Network Database Table** record.
- You should see the **Request Approvals** related list populated with the matching approval entries.

Steps to Add the Related List to the Form

1. Open any record from the **Network Database Table**.
2. Click the **context menu** (three dots in the top right of the form).
3. Navigate to **Configure > Related Lists**.
4. In the list of available related lists, select **Approval Request**.
5. Save the form configuration.
6. Refresh the record — you should now see the **Request Approvals** related list at the bottom of the form, displaying:

- State
- Approver
- Comments
- Approval for
- Created



Creation & Implementation of Flows, Actions in Flow Designer

Flow Designer in ServiceNow to automate the **Network Request** process. The flow manages the entire lifecycle of a request — from capturing catalog variables, creating a record in the Network Database, sending notifications, requesting approvals, handling logic conditions, and updating records — all without manual intervention.

This ensures:

Consistency in processing requests

Faster execution

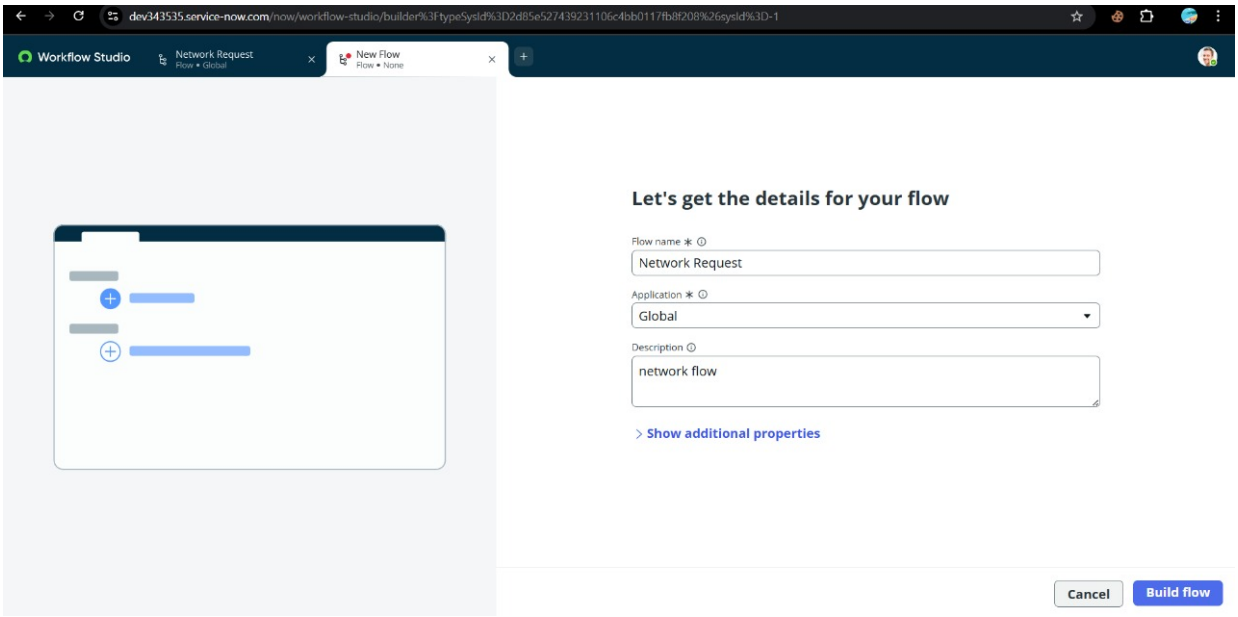
Fewer manual errors

Clear traceability of actions

Steps to Create the Flow

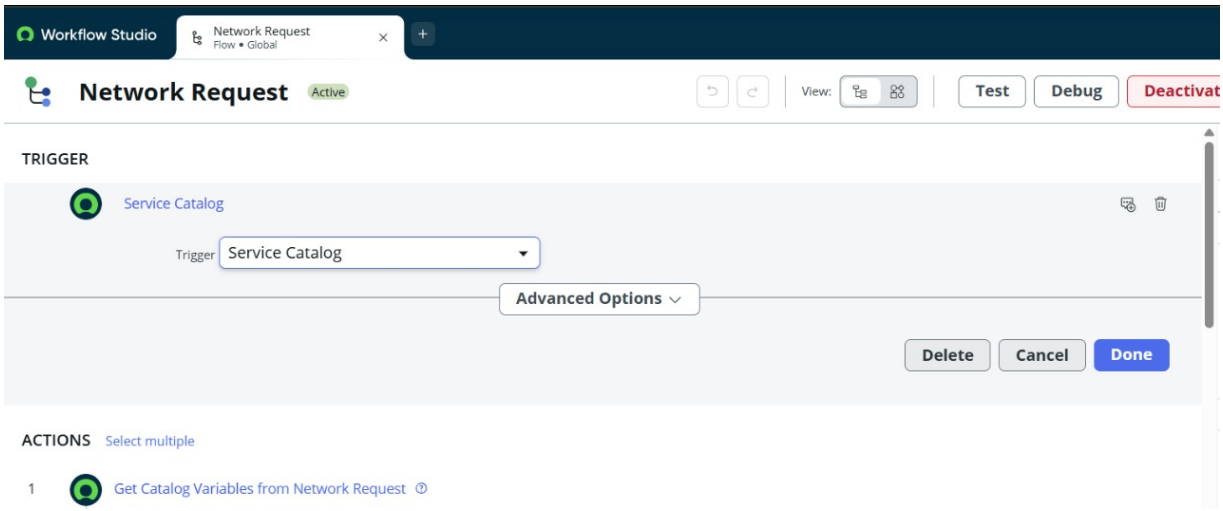
1. Creating the Flow

- Navigate to **Flow Designer** home page.
- Click **New** to create a new flow.
- Enter:
    - **Flow Name:** Network Request
    - **Description:***(e.g., Automates network request creation, approvals, and updates.)*
- 4. Click **Build Flow**.



2. Configuring the Trigger

- Click the **(+)** icon to add a trigger.
- Select:
  Trigger Type: Application → Service Catalog
- Click **Done**.

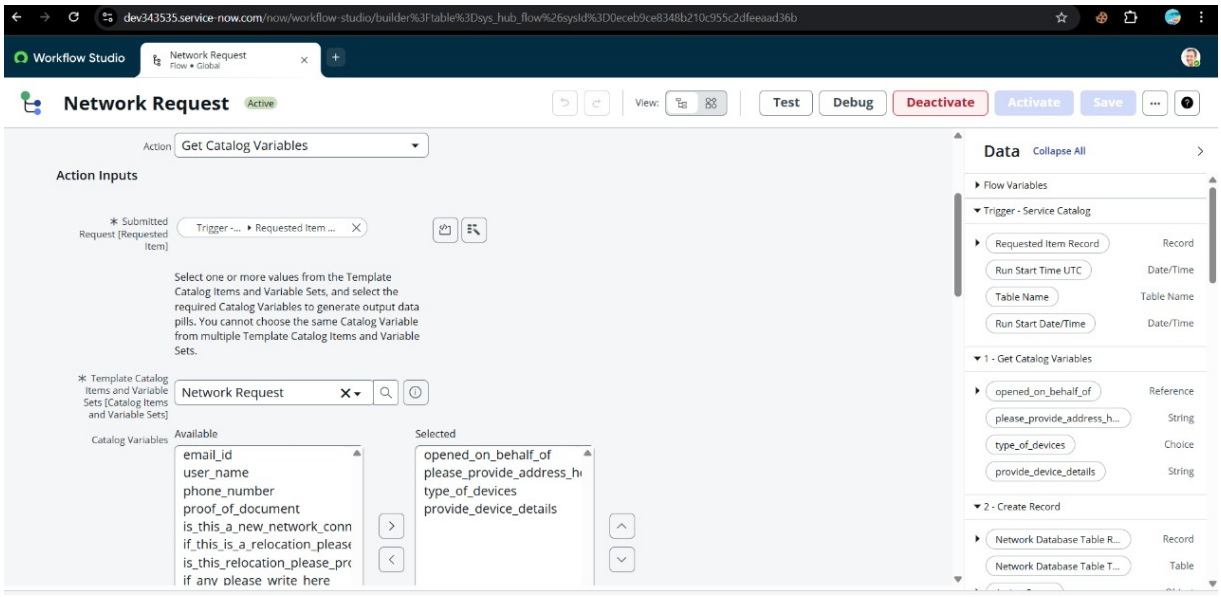


3. Adding Actions

A. Get Catalog Variables

1. Click **Actions**.
2. Search for **Get Catalog Variables**.
3. Select **Get Catalog Variables**.
4. Configure **Action Inputs**:

   Trigger → Service Catalog → Requested Item
5. In **Template catalog items**:
   - **Select Table:** Network Request
   - Move required variables to the **Selected** area.
6. Click **Done**.

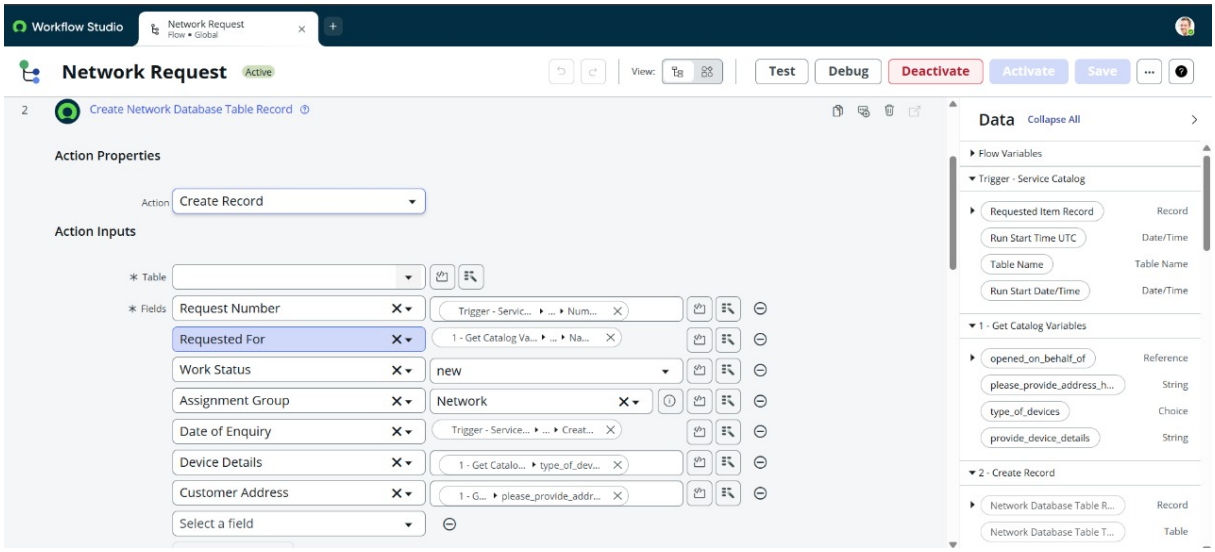


B. Create Record

1. Add a new action → **Create Record**.
2. Select **Table:** Network Database.
3. Click **Add Fields** and configure:

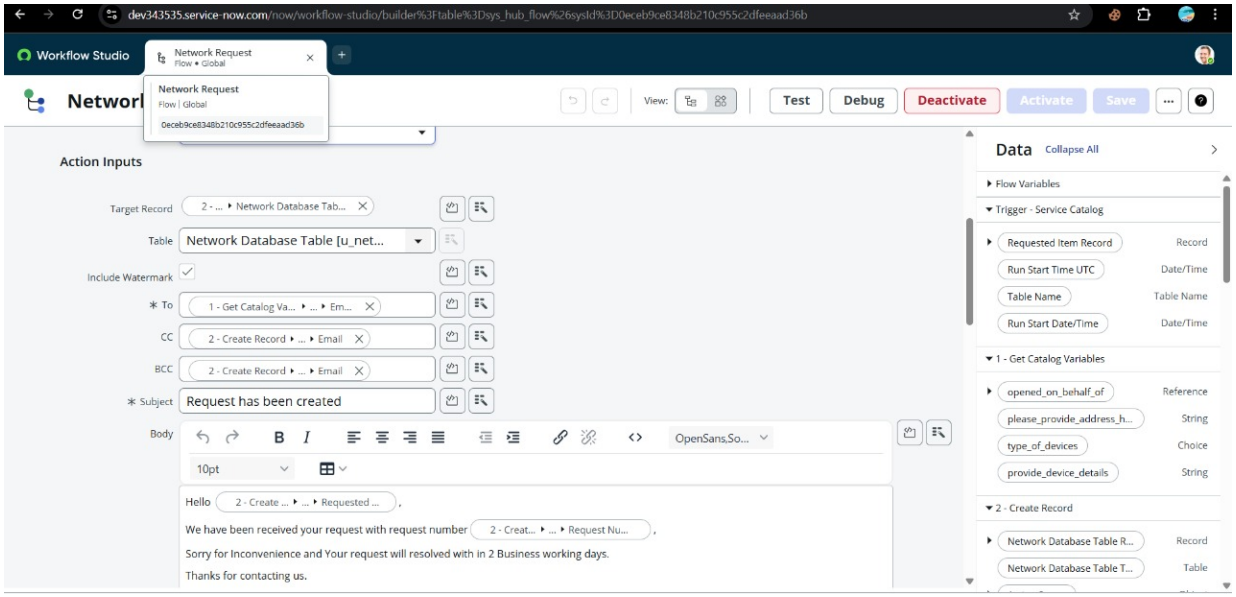
   o Map catalog variables to the respective table fields as per your requirements .

4. Click **Done**.



C. Send Email

1. Add a new action → **Send Email**.
2. **Target Record:** Select → **Create Record** → **Network Database Table** (auto-selected).
3. Configure:
    - **To / CC / BCC:** Static or dynamic recipients.
    - **Subject & Body:** Use variables and static text as shown in the design screenshot.
4. Click **Done**



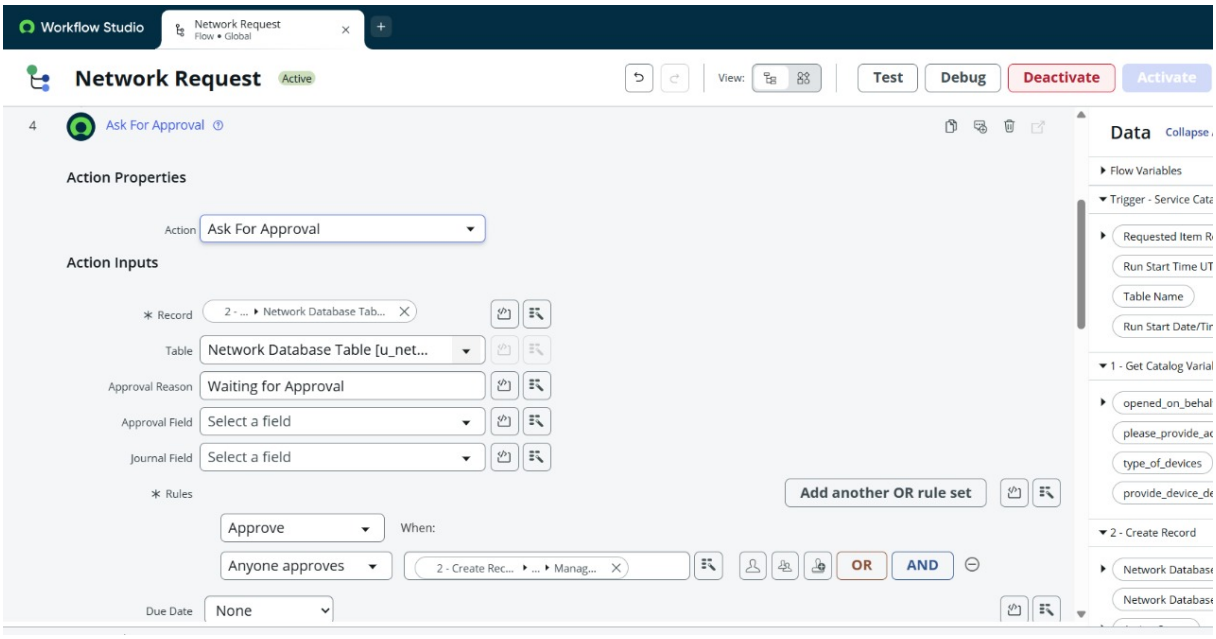
D. Ask for Approvals

1. Add a new action → **Ask for Approval**.
2. **Target Record:** Create Record → Network Database Table.
3. Configure:
    - **Approval Reason:** "Waiting for Approval".
    - **Approval Rules:** Approve, Reject, Approve/Reject.
    - **Approval Type:** Anyone approves, Everyone

approves, etc. (static/dynamic assignment).

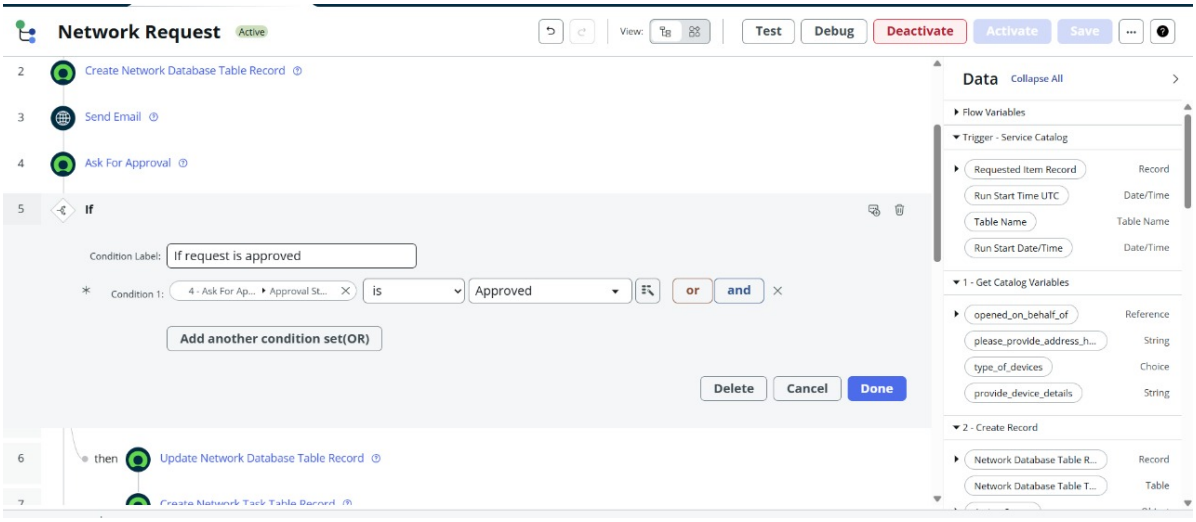
- Here we chose abel tuter

4. Click **Done**



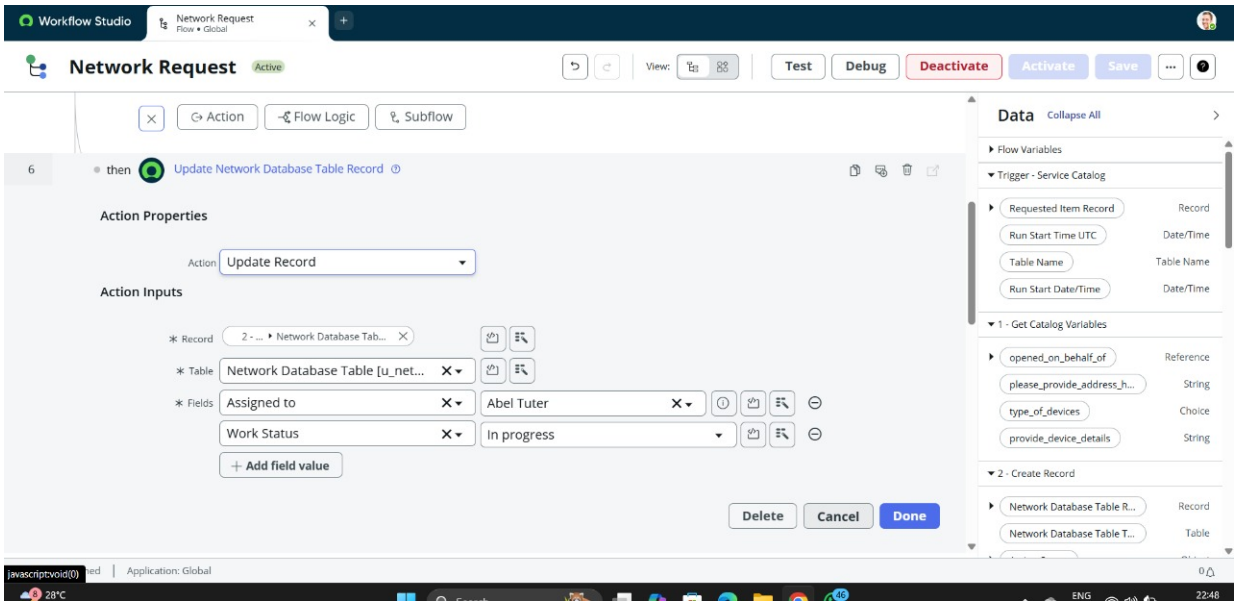
E. Flow Logic (If Condition)

1. Add a new action → **Flow Logic** → **If Condition**.
2. Configure:
   Condition: "Ask for approvals" state is Approved .
3. Click **Done**.



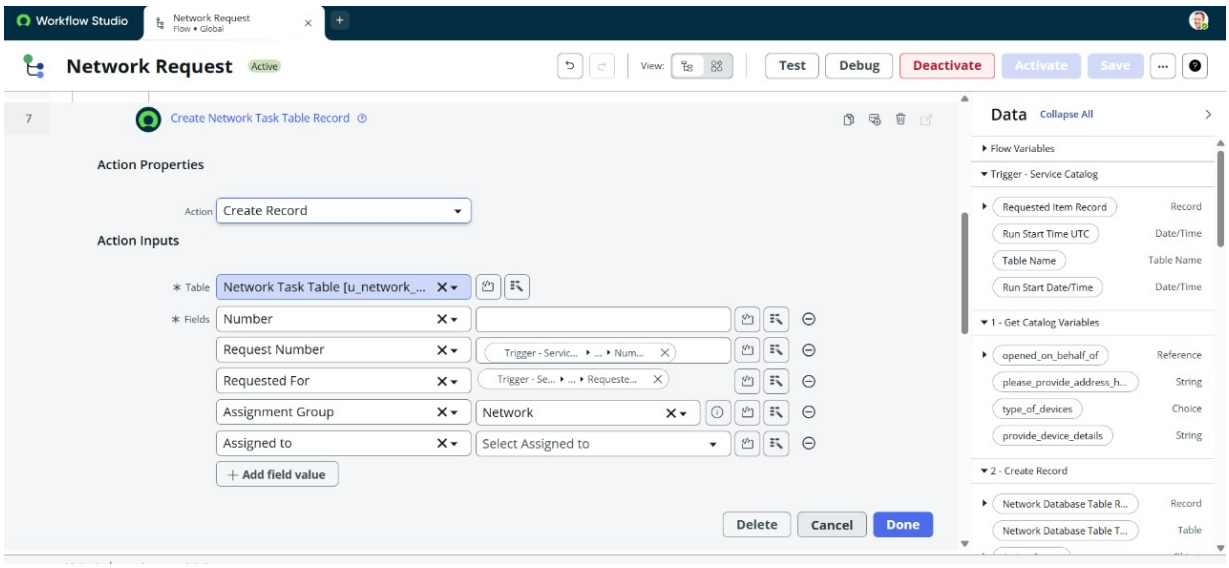
F. Update Record

1. Add a new action → **Update Record**.
2. **Target Record:** Create Record → Network Database Table (autoselected).
3. Configure required fields (like Assigned to -> Abraham Lincoln Work Status -> Work in Progress).
4. Click **Done**.



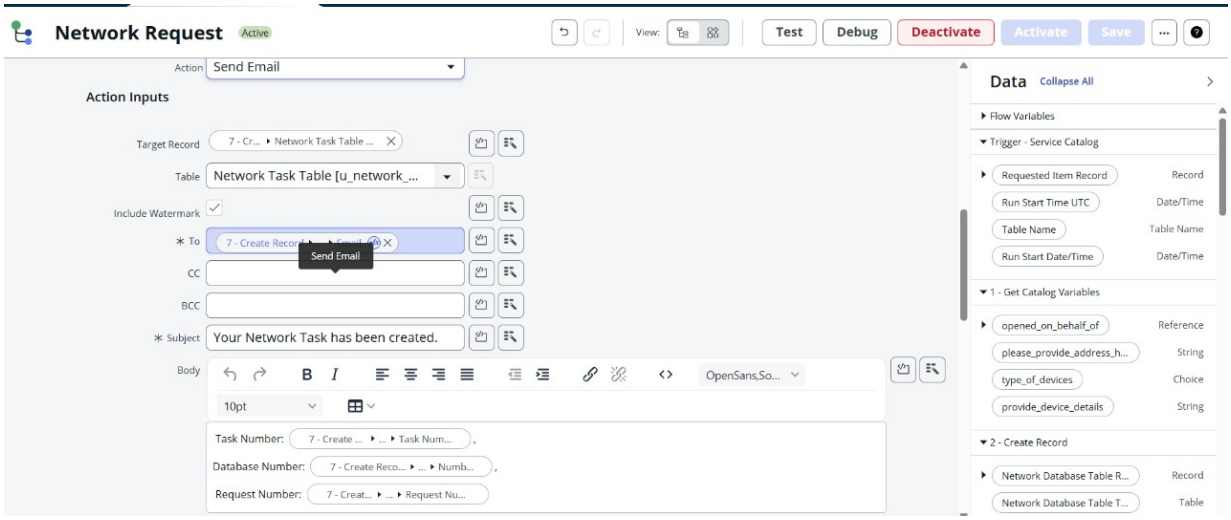
G: Create Network Task Table Record

1. Add a new action → **Create Record**.
2. Select **Table** → *Network Task Table [u_network_task]*.
3. Under **Fields**, map Service Catalog variables to the table fields:
   - **Database Number** → Auto-populated (Number Maintenance / Business Rule).
   - **Request Number** → Map from Catalog Variable (e.g., *Request Number*).
   - **Requested For** → Map from Catalog Variable (Requested For).
   - **Description** → Map from Catalog Variable (Description of request).
   - **Priority** → Map from Catalog Variable (Priority).
   - **Assignment Group** → Network Assignment Group (static or from variable).
   - **Assigned To** → Leave blank initially (will be set later after approval).
4. Click **Done**.



H. Send Email (Request Created)

a. Add a new action → **Send Email**.
b. Target Record → *Create Network Task Table Record*.
c. Configure:
   - **To:** Requestor / Requested For.
   - **Subject:**"Your Network Task has been created."
   - **Body:** Include Task Number, Database Number, Request Number.
d. Click **Done**.



I. Ask for Approval

a. Add a new action → **Ask For Approval**.

b. Target Record → *Network Task Table Record*.

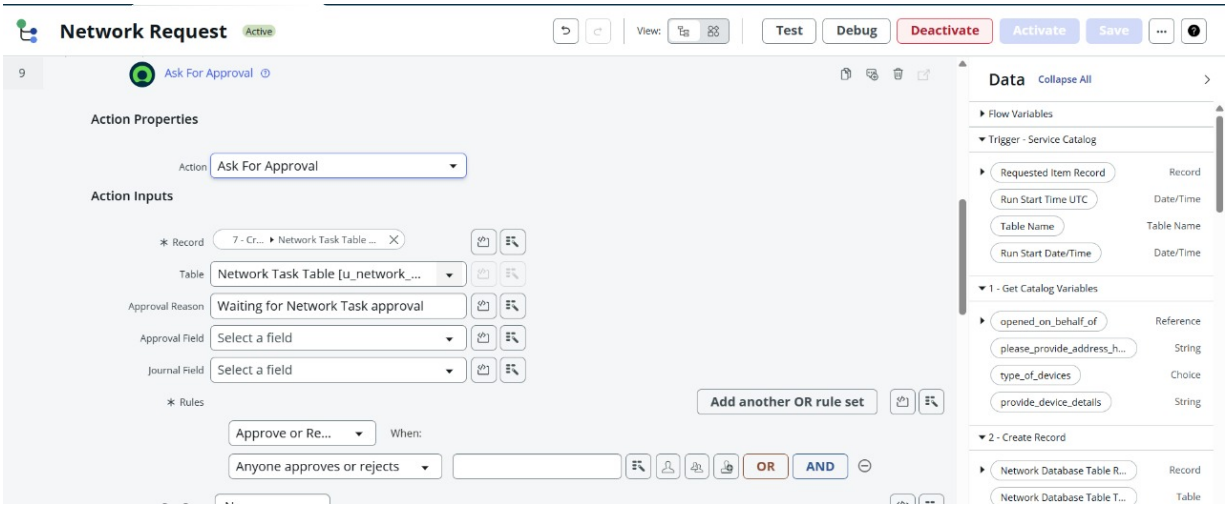
c. Configure:

   a. **Approval Reason:** "Waiting for Network Task approval".

   b. **Approval Rules:** Approve / Reject.

   c. **Approval Type:** Choose (e.g., *Anyone Approves*).

d. Click **Done**.



J. If Condition – Approval Status Changes

a. Add action → **If Condition**.

b. Condition → *Approval State is Approved*.

c. In the **Then branch**:

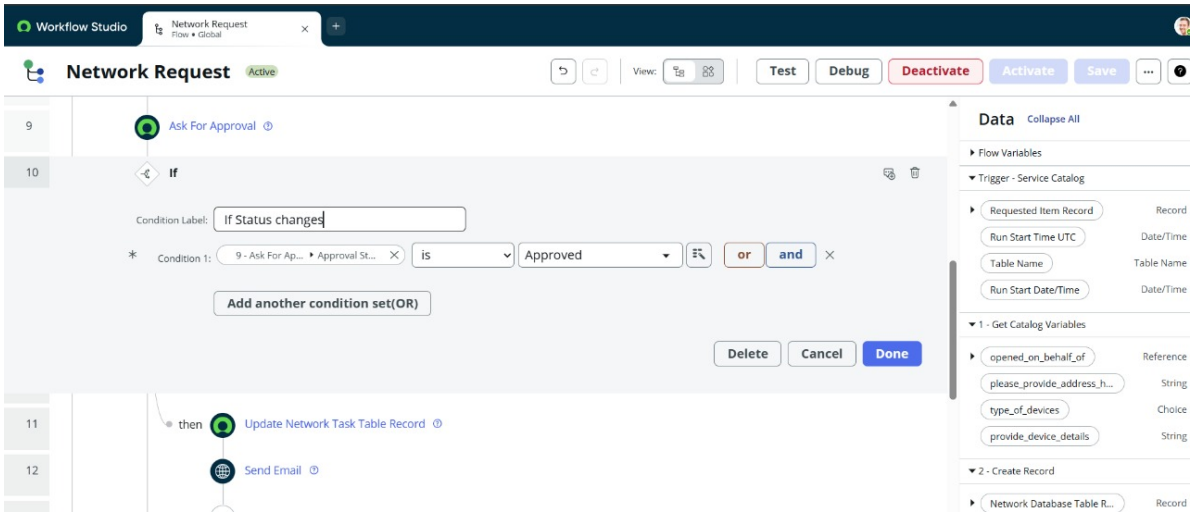
Update Record

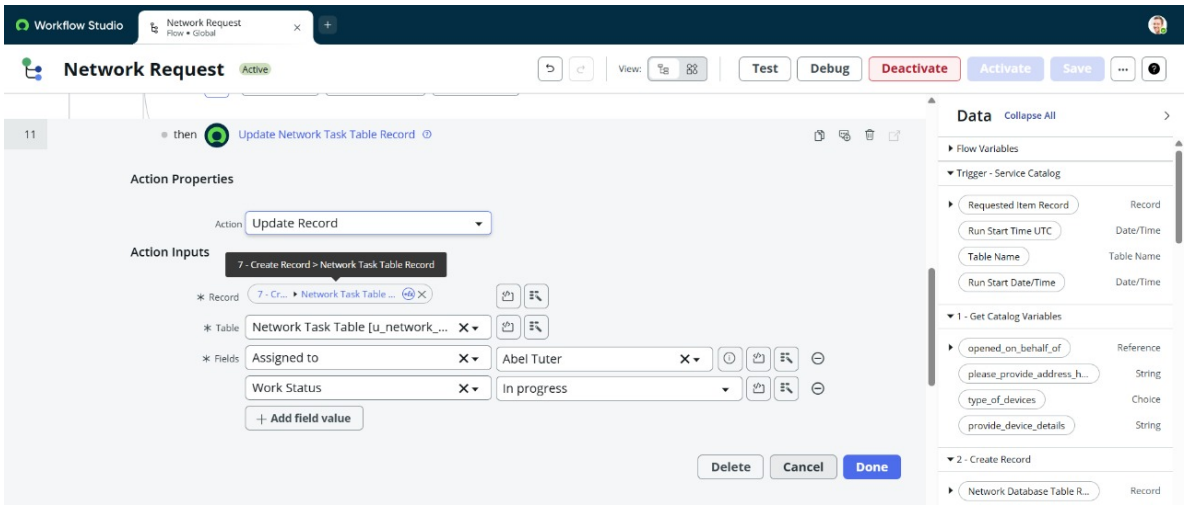
- Target Record → *Network Task Table Record*.
- Update fields:
  a. Assigned To → *Adam Ringle*.
  b. Work Status → *Work in Progress*.

d. Click **Done**.

Send Email (Approved)

Add action → Send Email.





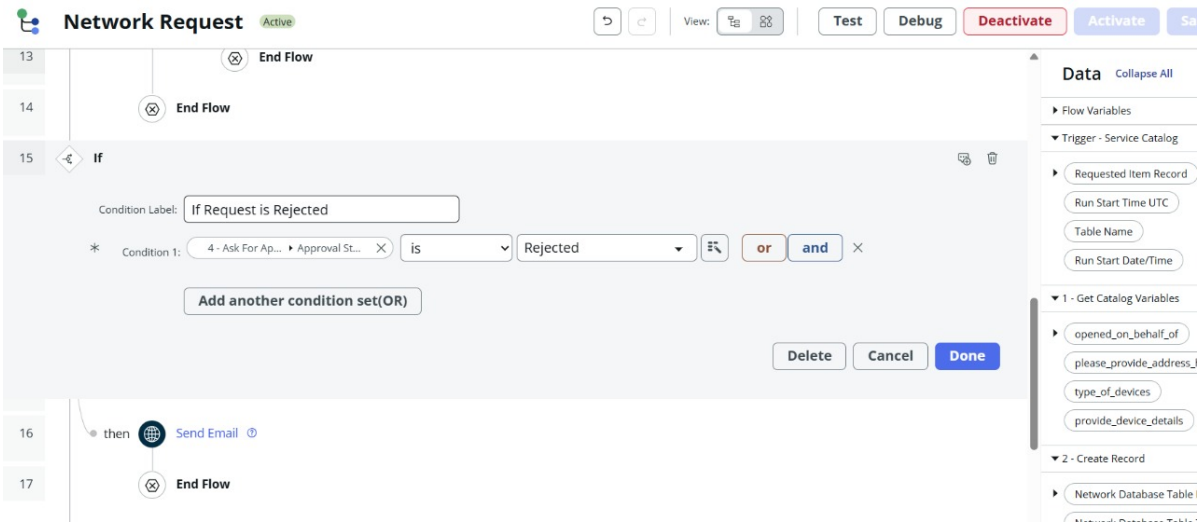
K. If Condition – Request Rejected

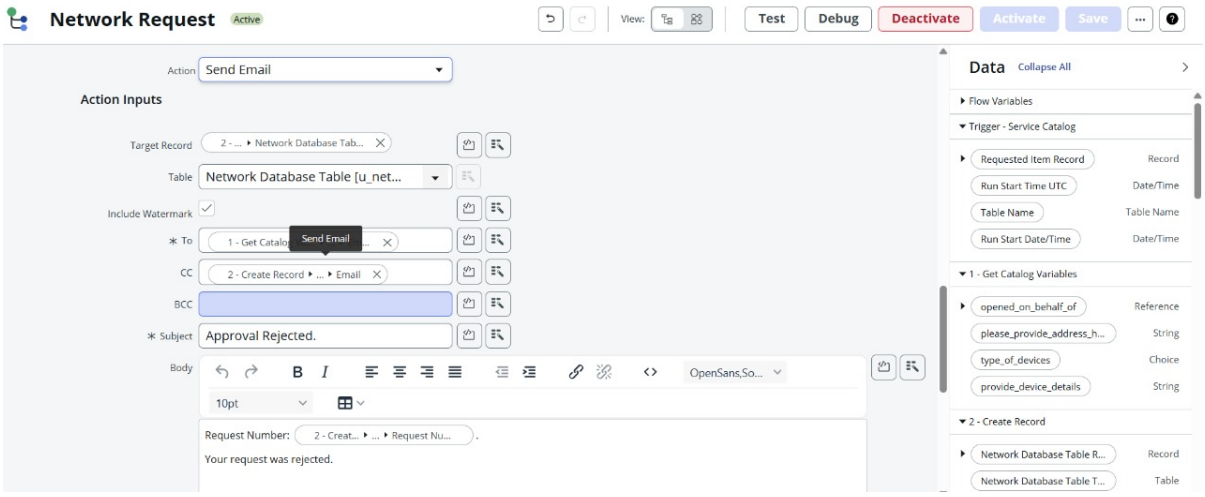
a. Add another If Condition for *Approval State is Rejected*.

b. In the Then branch:

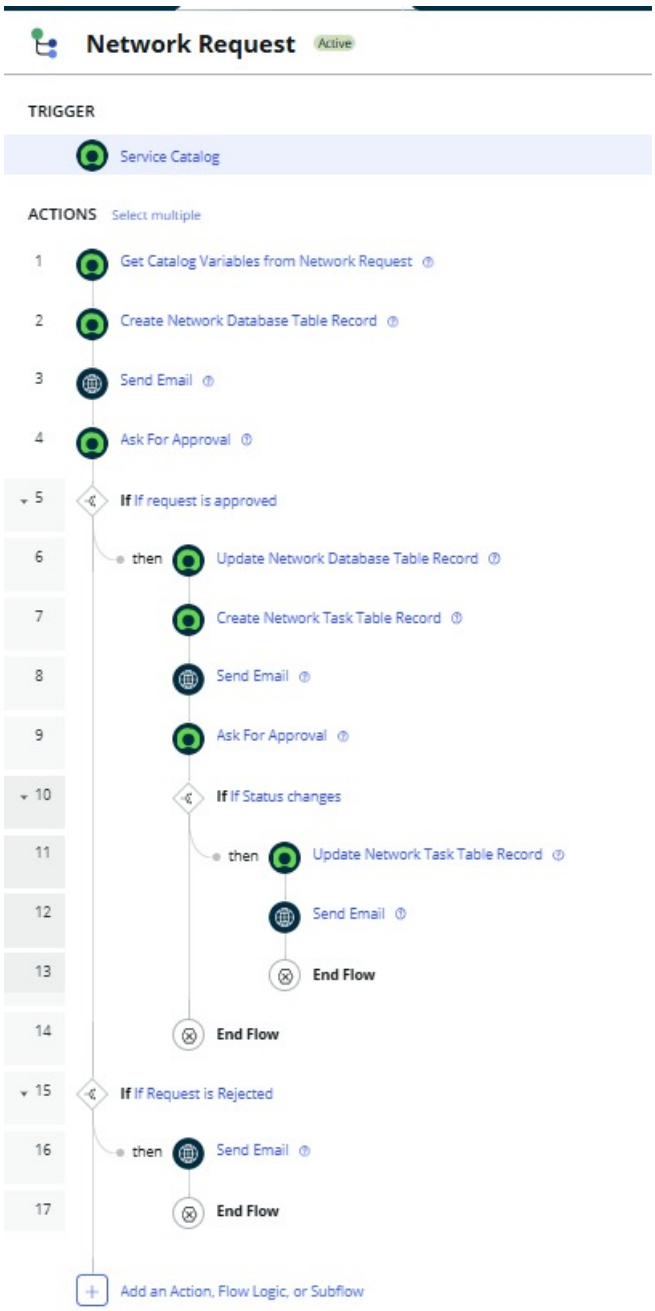
Send Email (Rejected)

- Notify requestor that their request was rejected.
- Optionally include rejection comments.


Network Request
Active
View:
Test
Debug
Deactivate
Activate
13
End Flow
14
End Flow
15
If
Condition Label:
If Request is Rejected
Condition 1:
4 - Ask For Ap... ▸ Approval St...
is
Rejected
or
and
Add another condition set(OR)
Delete
Cancel
Done
16
then
Send Email
17
End Flow
Data
Collapse All
Flow Variables
Trigger - Service Catalog
Requested Item Record
Run Start Time UTC
Table Name
Run Start Date/Time
1 - Get Catalog Variables
opened_on_behalf_of
type_of_devices
provide_device_details
2 - Create Record
Network Database Table


Network Request
Active
View:
Test
Debug
Deactivate
Activate
Save
Action
Send Email
Action Inputs
Target Record
2 - ... ▸ Network Database Tab...
Table
Network Database Table [u_net...
Include Watermark
To
1 - Get Catalog ...
Send Email
CC
2 - Create Record ▸ ... ▸ Email
BCC
Subject
Approval Rejected.
Body
OpenSans,So...
10pt
Request Number:
2 - Creat... ▸ ... ▸ Request Nu...
.
Your request was rejected.
Data
Collapse All
Flow Variables
Trigger - Service Catalog
Requested Item Record
Record
Run Start Time UTC
Date/Time
Table Name
Table Name
Run Start Date/Time
Date/Time
1 - Get Catalog Variables
opened_on_behalf_of
Reference
please_provide_address_h...
String
type_of_devices
Choice
provide_device_details
String
2 - Create Record
Network Database Table R...
Record
Network Database Table T...
Table

OVERALL FLOW:



Summary

This project delivers an efficient ServiceNow-based solution for handling network service requests. By using a dedicated service catalog, automated approval workflows, and real-time notifications, it streamlines the request process for both users and technicians. The system ensures accurate request capture, faster resolution through automation, and better visibility with reporting and SLA tracking.