**Automated Network Request Management in ServiceNow**

**Project Description:**

This project aims to design and implement a streamlined, automated solution for managing network-related service requests within ServiceNow. It enables end users to submit requests for network services through a user-friendly self-service portal.

The system leverages ServiceNow’s workflow engine, catalog items, and approval processes to ensure requests are properly captured, validated, and routed for fulfillment. Upon submission, requests trigger automated notifications, task assignments, and—where applicable—integration with network automation tools or scripts to fulfill standard requests without manual intervention.

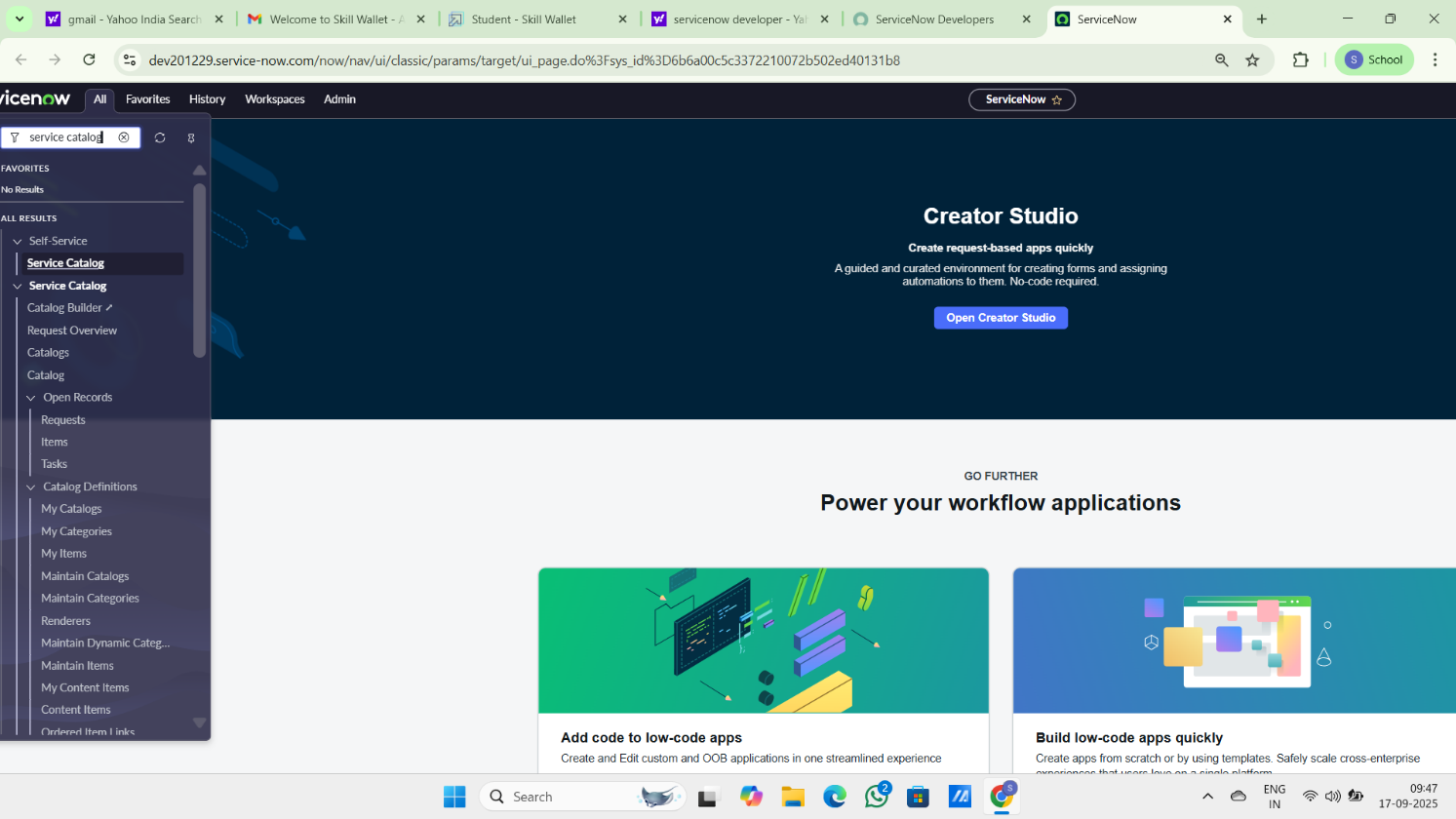
**Key Features:**

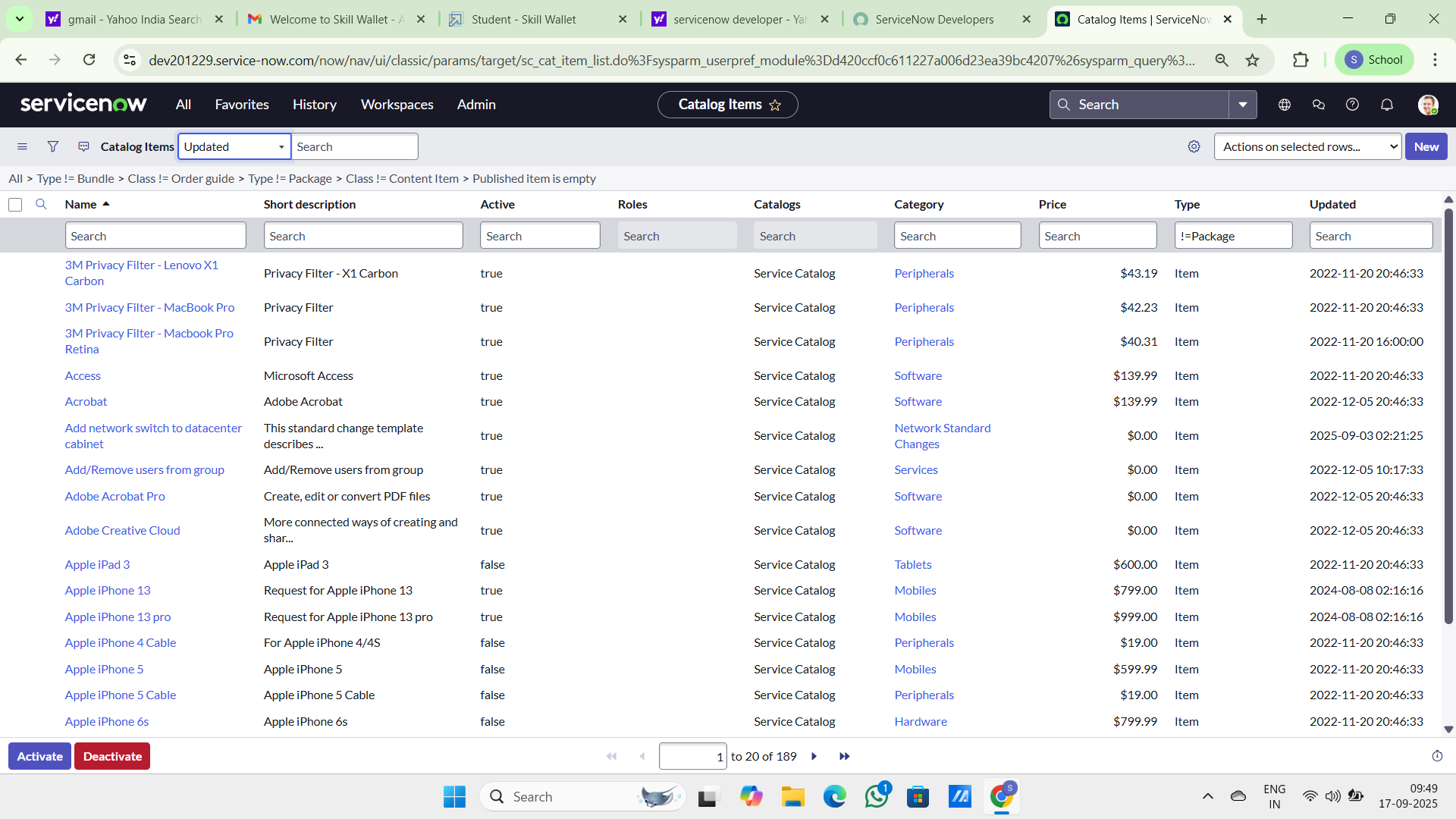
* Custom service catalog for common network requests
* Dynamic forms to capture relevant request details
* Automated approval workflows based on request type and sensitivity
* Integration with infrastructure management or orchestration tools (optional)
* Real-time status updates and notifications to requesters and technicians

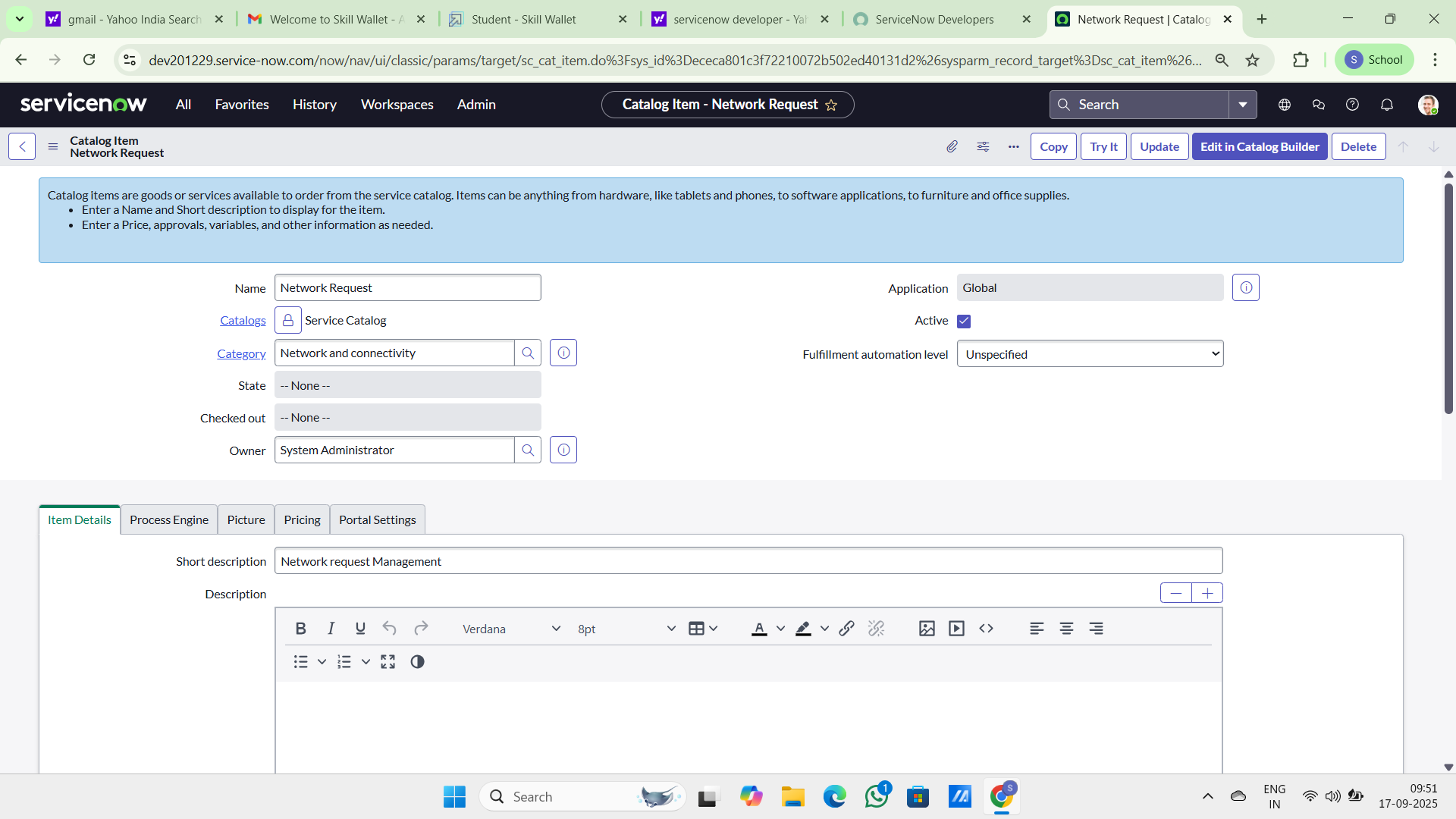
Reporting and analytics on request volume, resolution time, and SLA adherence

**Step1: Service Catalog Creation**

1. Navigate to Application navigator
2. Click on All >> search for Service Catalog
3. Under Service Catalog>> Maintain items
4. Click on New
5. Fill the details >> Name– Network Request
6. Select Catalog>> Service Catalog
7. Select Category>> Network
8. Fill the Short Description as Network request Management
9. Click on Save.



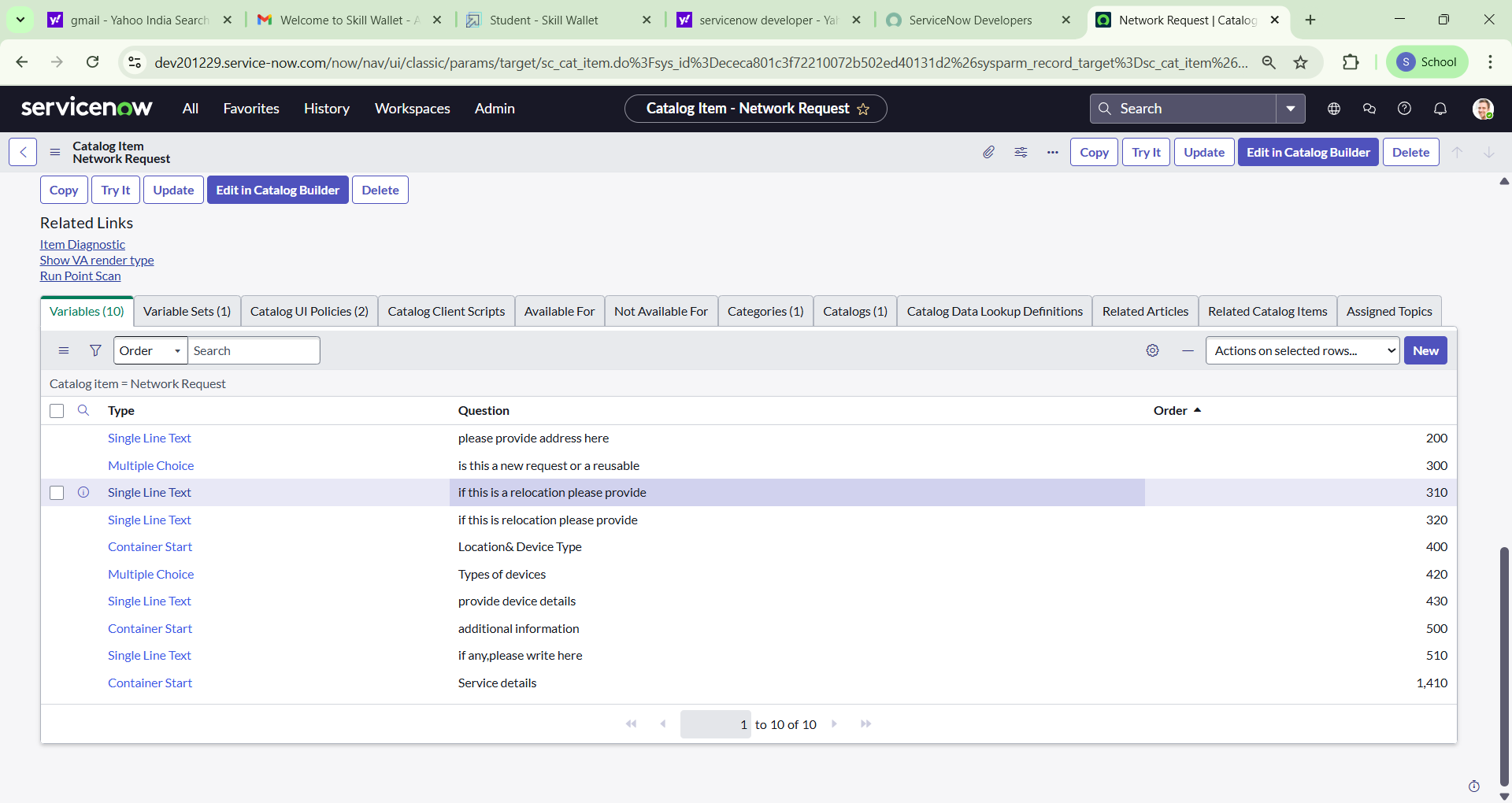


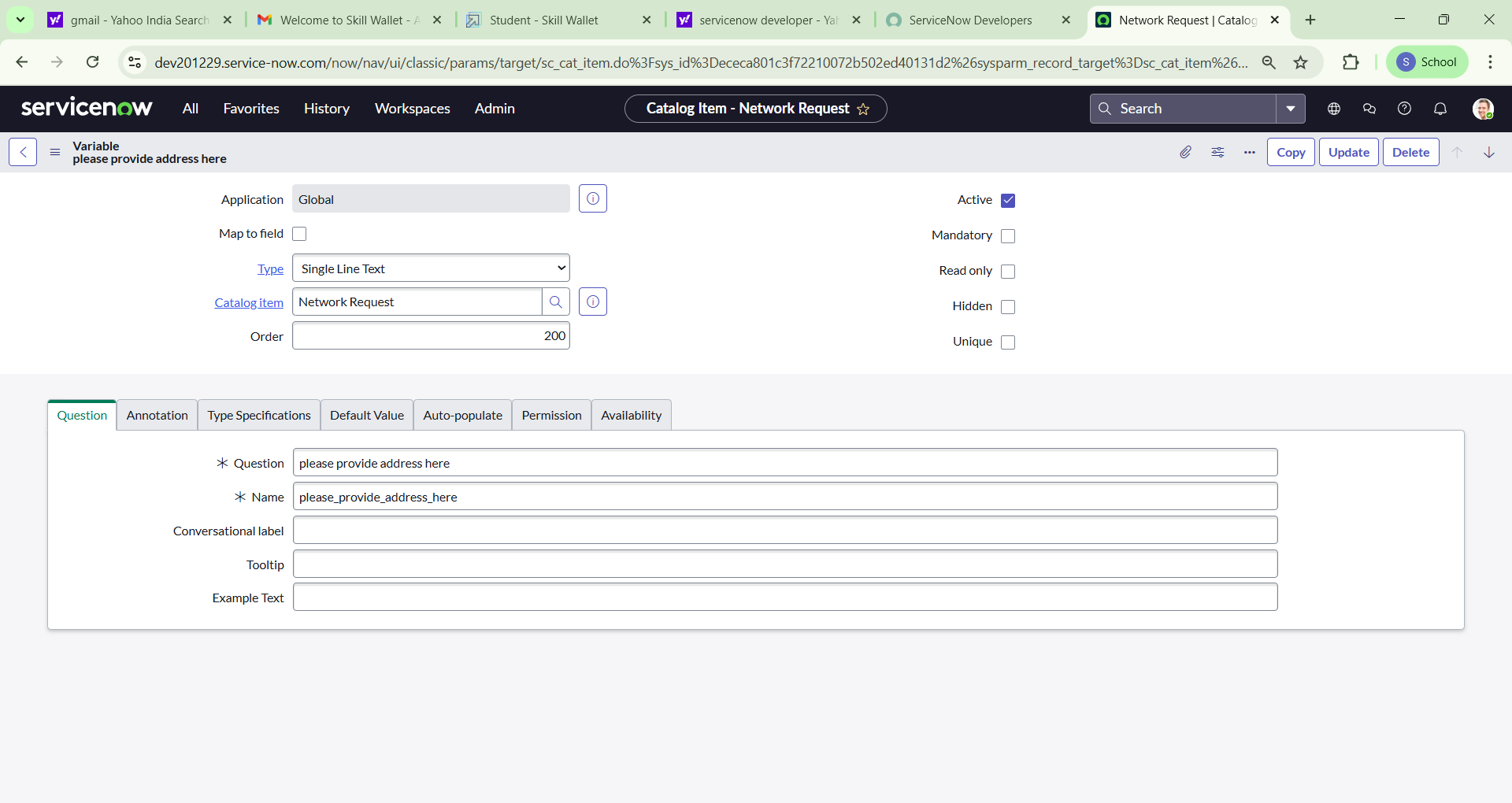


**Variables Configuration**

Open the catalog item just created.  
Scroll down to the **Variables** related list and click **New** to create form fields.

1. Select Variables type as Single, Multi line text, reference, choices etc as per requirement
2. Catalog item–  Network Request
3. Order–100,200,300,,,,
4. Question– provide the variable label
5. Name–provide the variables name(used for scripting)
6. Tooltip– this will appear when cursor overed on the field
7. Example text – this will suggest what we need to enter on the field.
8. Mandatory, Read-Only– need to configure on demand
9. Auto populate– need to select dependent variable, apply dot walking to get selected value.
10. Click on Save or Submit.





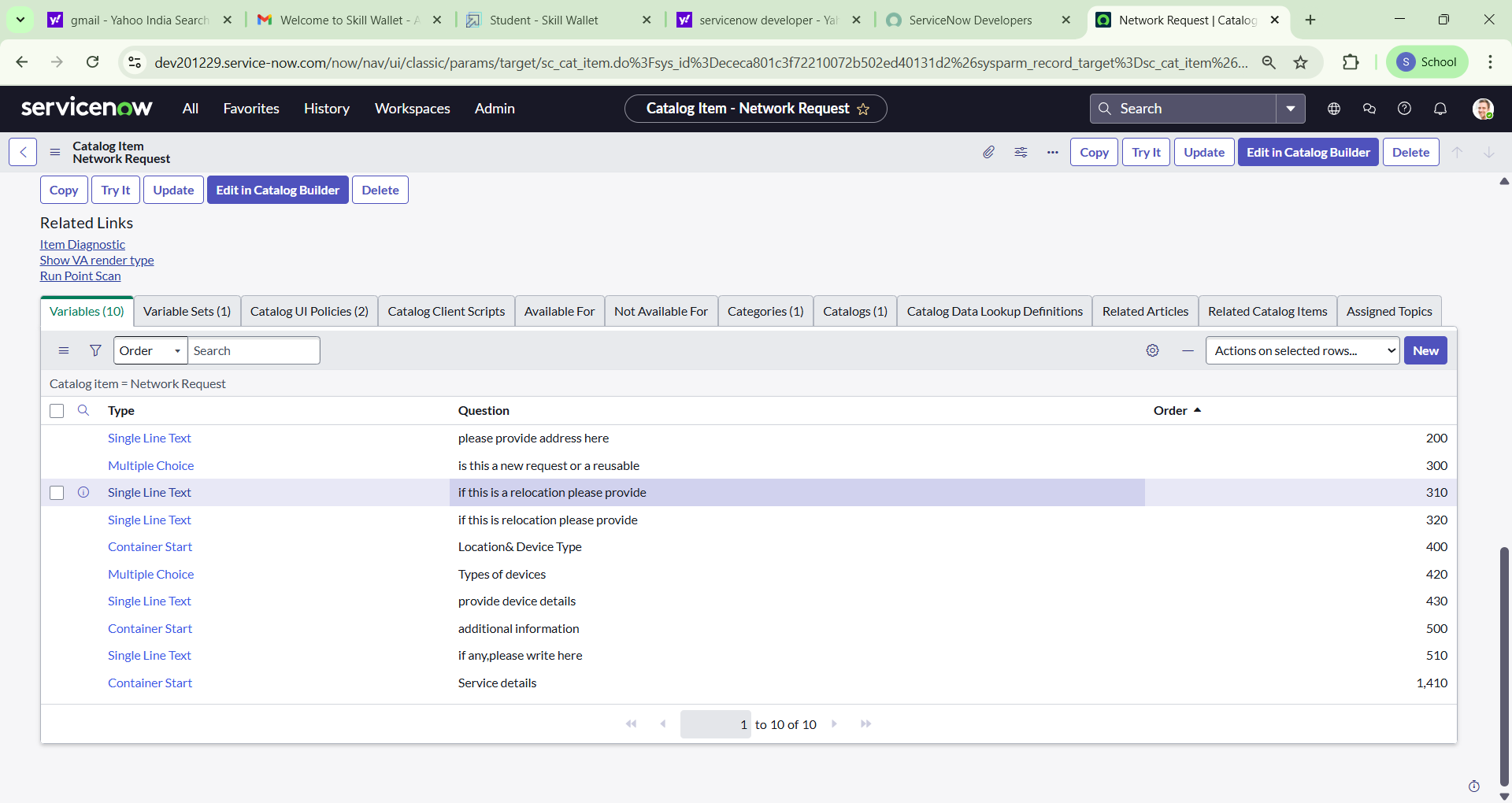
**Variables Types**

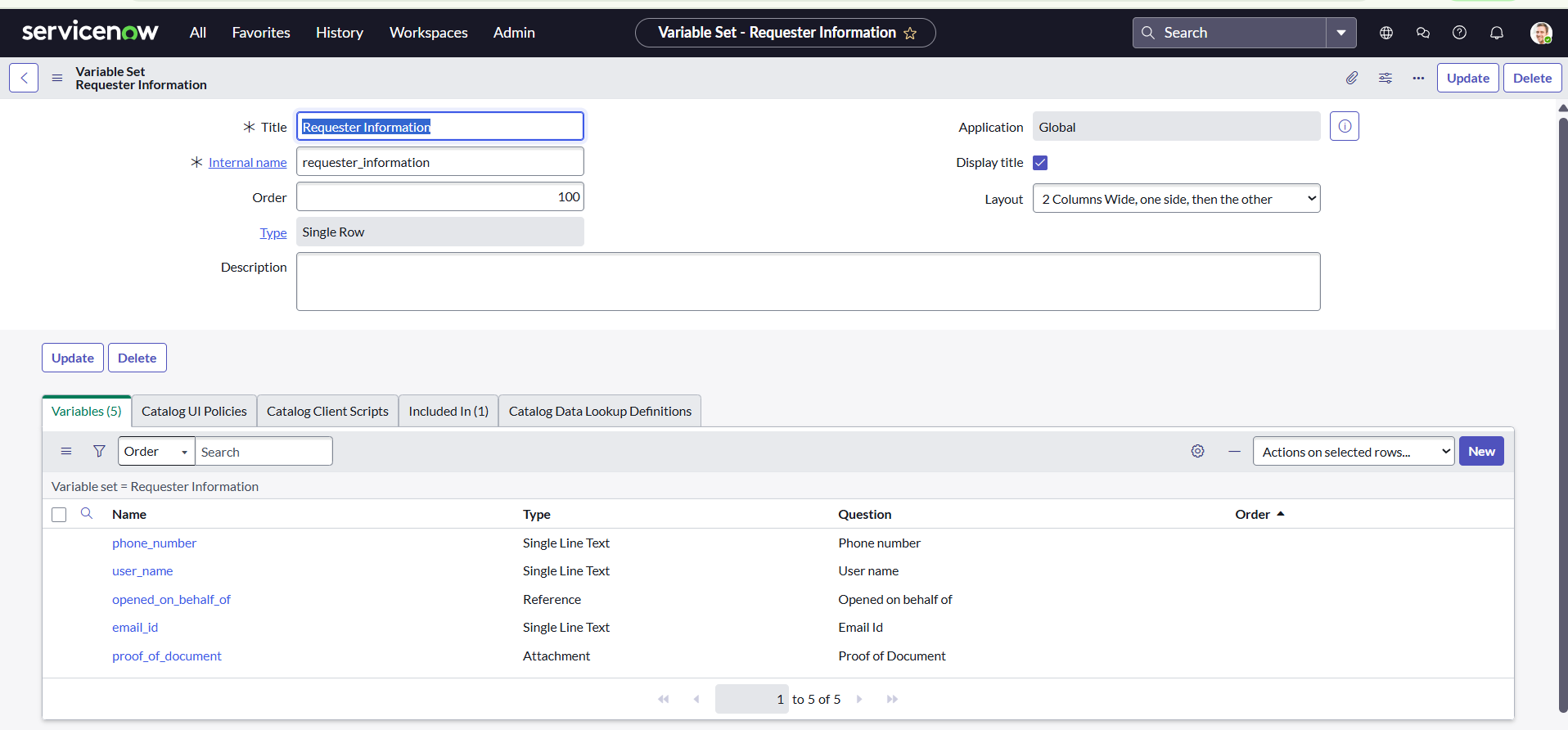
1. Is this a New connection or Relocation? >> **Choice** >> **New/ Relocation/None**
2. If this is a relocation, Please provide your relocated address here>>**String**
3. Types of devices>> **Choice**>> **Laptop/Mobiles/Others**
4. Please provide address here**>>String**
5. Provide device details here>> **String**
6. If anything else, please specify>> **String**

**Variable Set Configuration**

* To enhance form usability:
  + Navigate to the **Variable Sets** (optional).
  + Follow the same procedure as we used for Variables Creation, for the variable set as well.
  + Apply variable sets to the catalog item.

**Variables Types**

1. Opened on behalf of >> Reference>> reference to user table
2. Email Id >> Single line text >> Auto populate by Opened on behalf of variable.
3. User name >>Single line text >> Auto populate by Opened on behalf of variable.
4. Phone Number >>Single line text >> Auto populate by Opened on behalf of variable.
5. Proof of Document >> Attachment

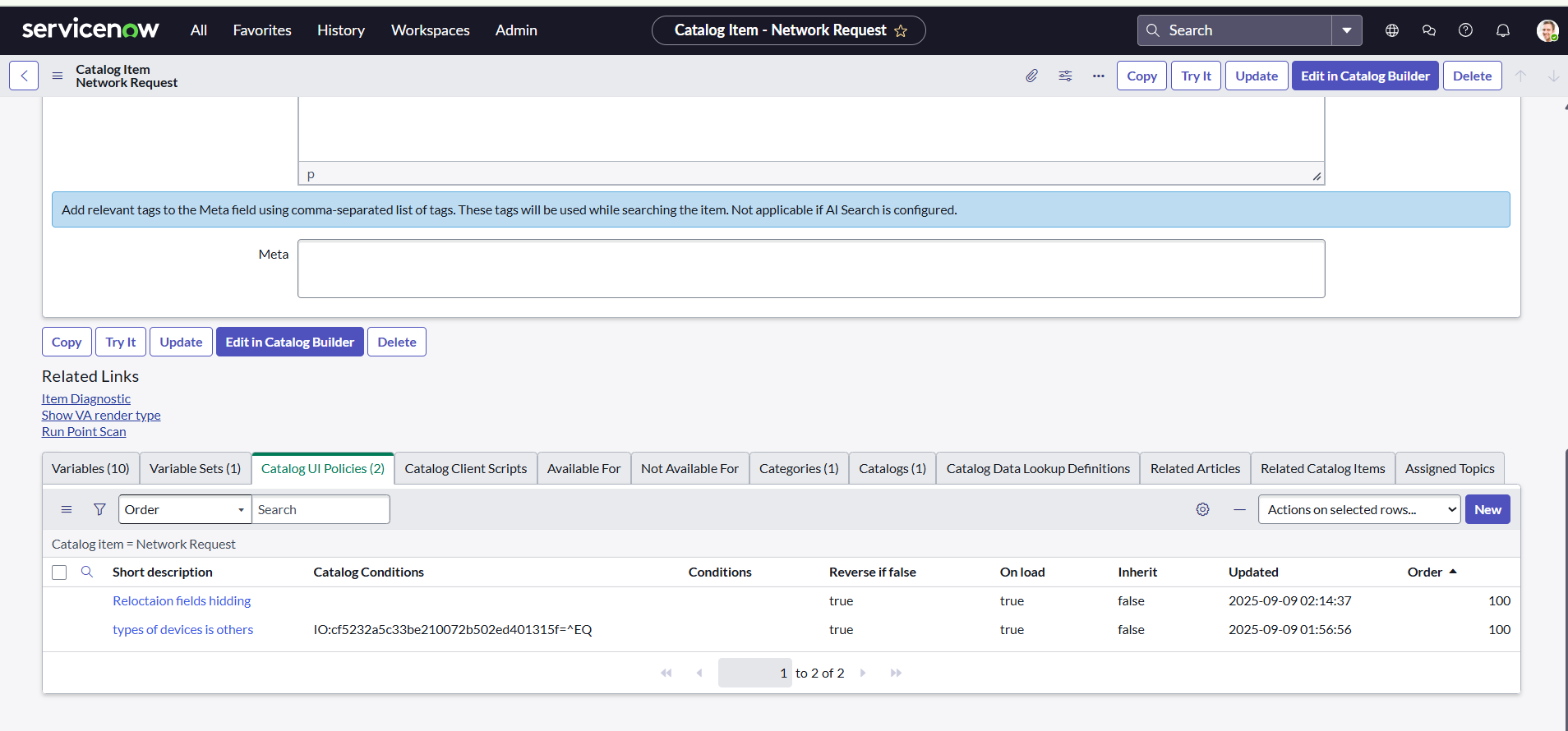


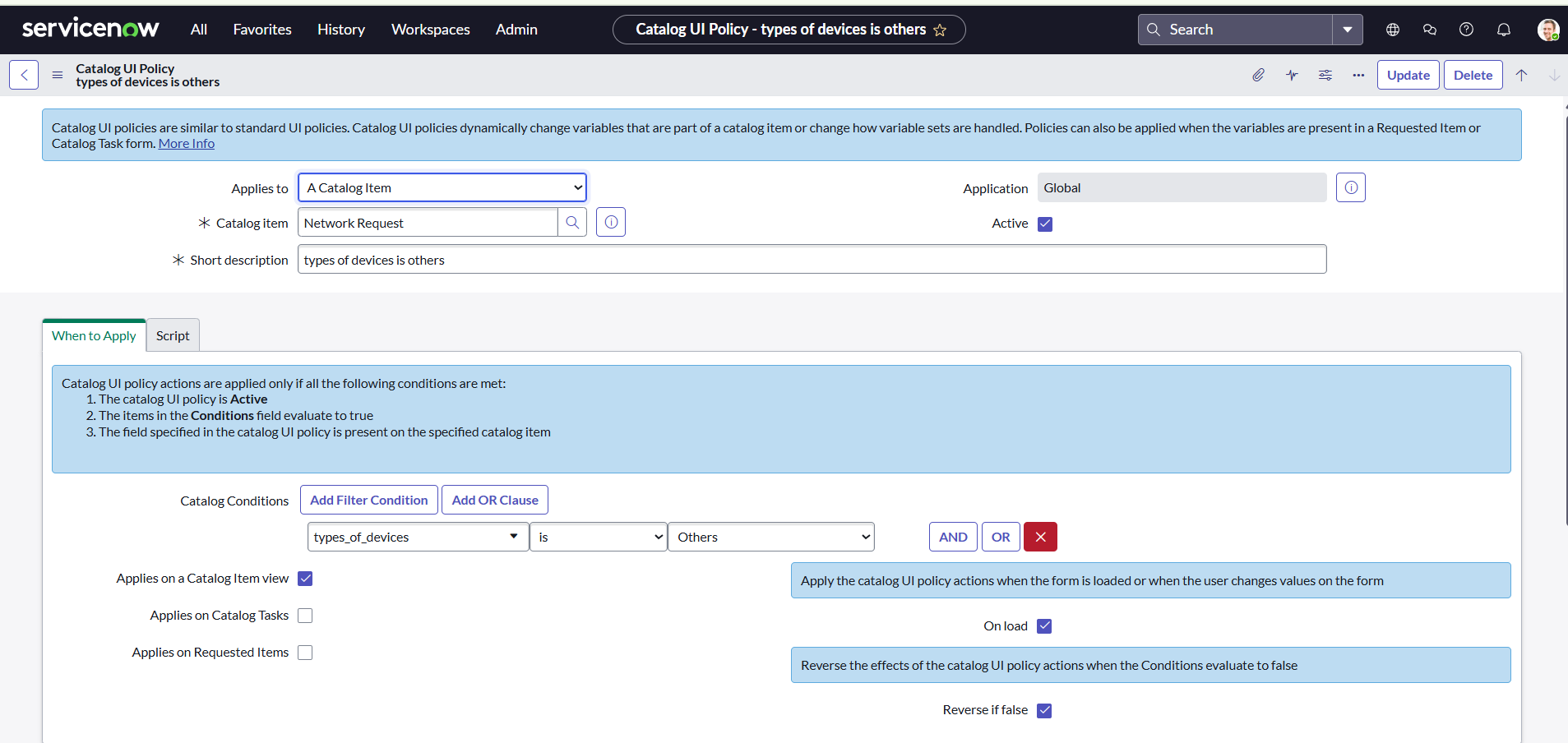
**Catalog UI Policy Configuration**

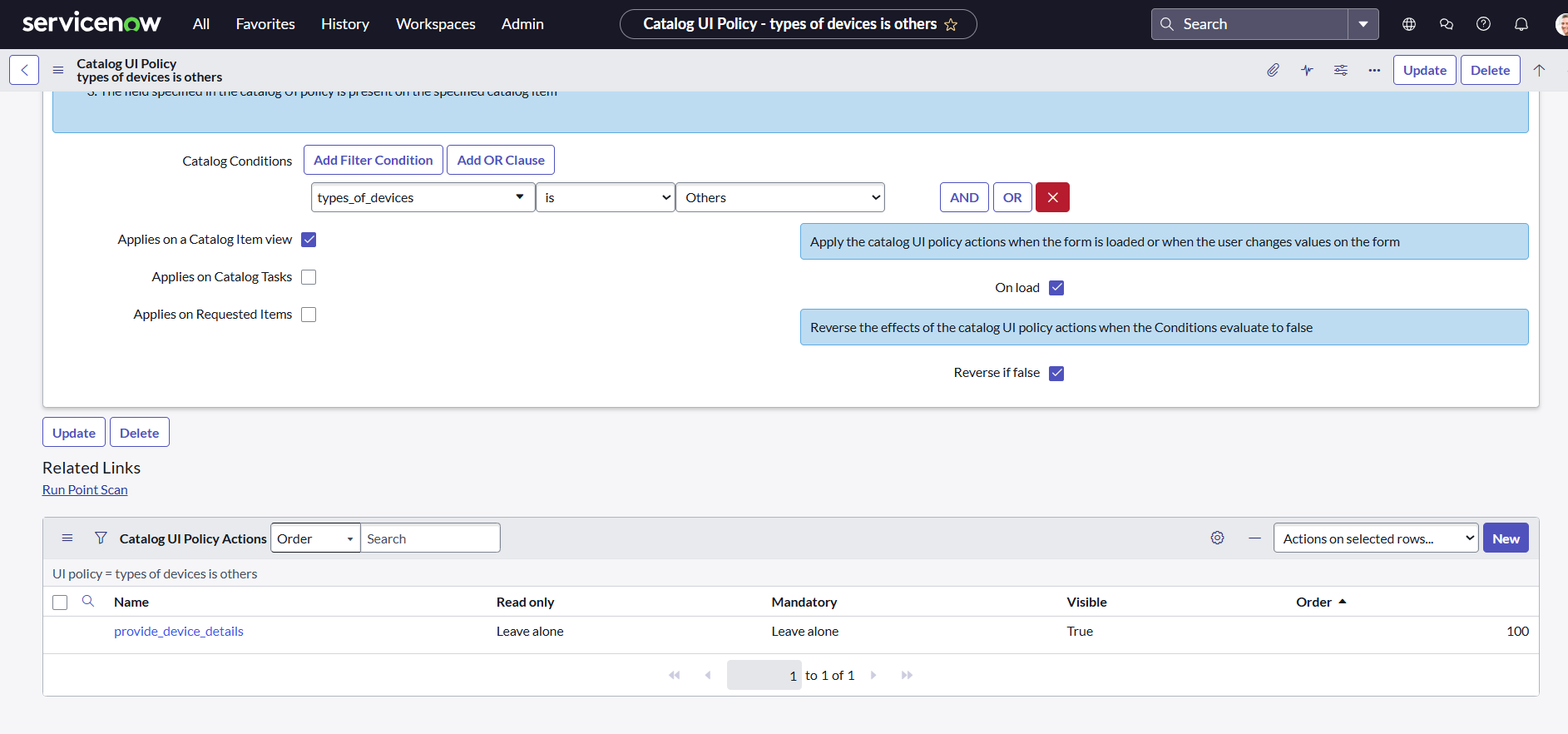
Scenario: If user selects types of devices is **Others,** then Please specify field should populate.

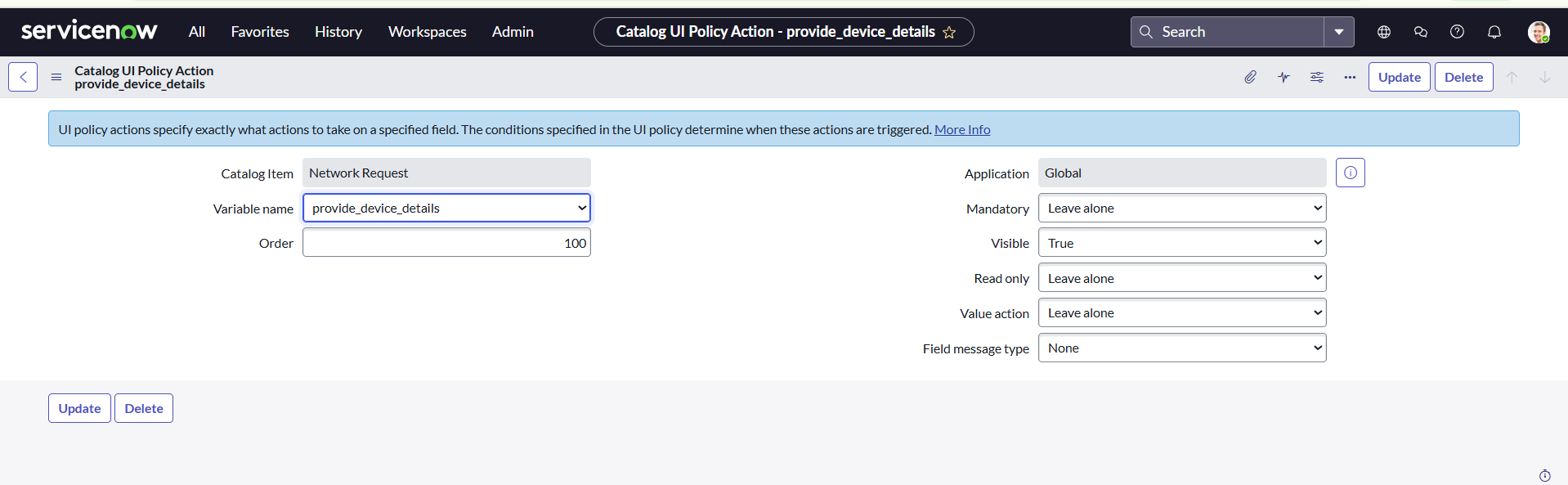
**Procedure:**

1. Navigate to catalog items
2. Open Network Request item
3. In related list, we have Catalog UI policy
4. Click on New button to configure New UI policy
5. Select Applies to as Catalog item
6. Select catalog item as Network Request
7. Provide short description, if required
8. Apply condition>> **types of devices** is **others**
9. Clickon save, after saving the form will get UI policy actions in the related list
10. Click on New button to configure new UI Policy action, and Select the variable which we want to display on condition
11. Make Visible True as per our requirement
12. Update the UI Policy and Test the same on Catalog form.









**Step2: Creation of Table**

**Creation of Table**

· **Navigate to**: System Definition > **Tables**.

· Click **New** to create a new table.

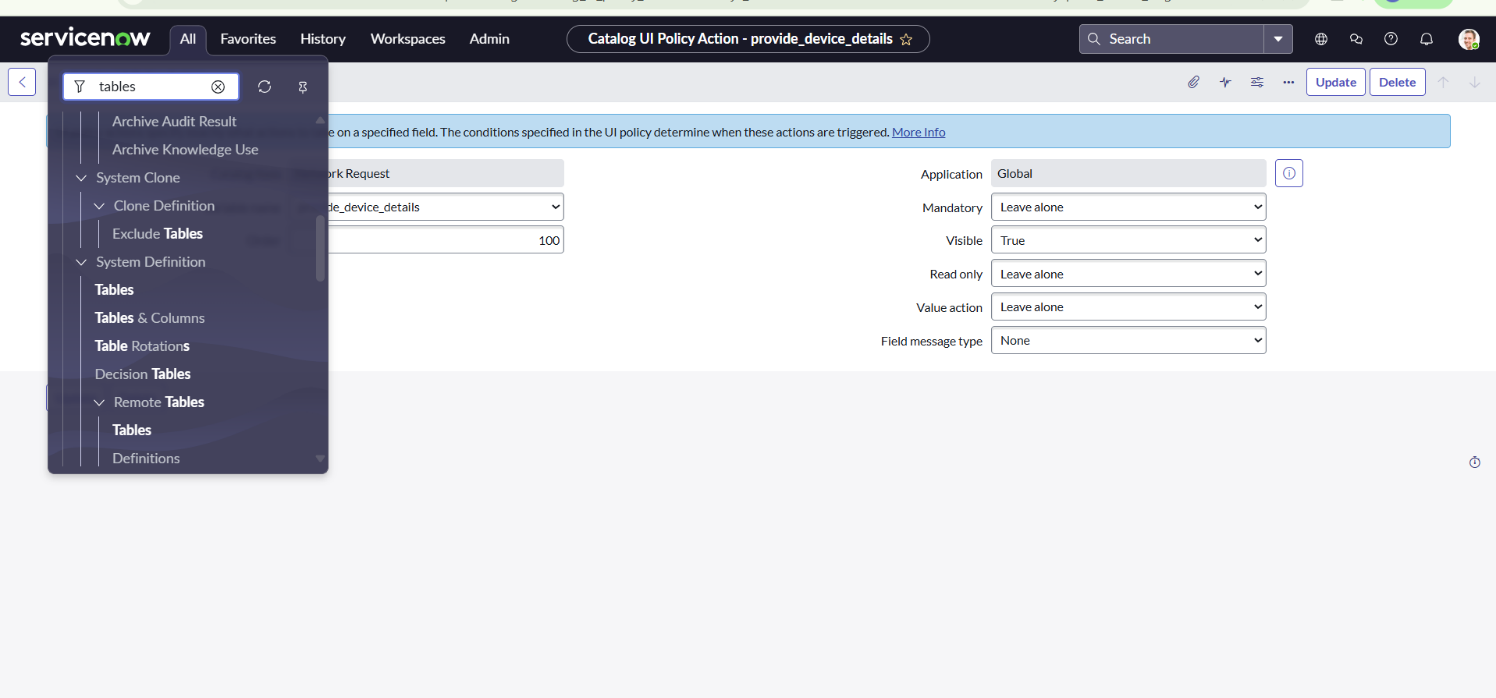
· Fill **in Table Information**:

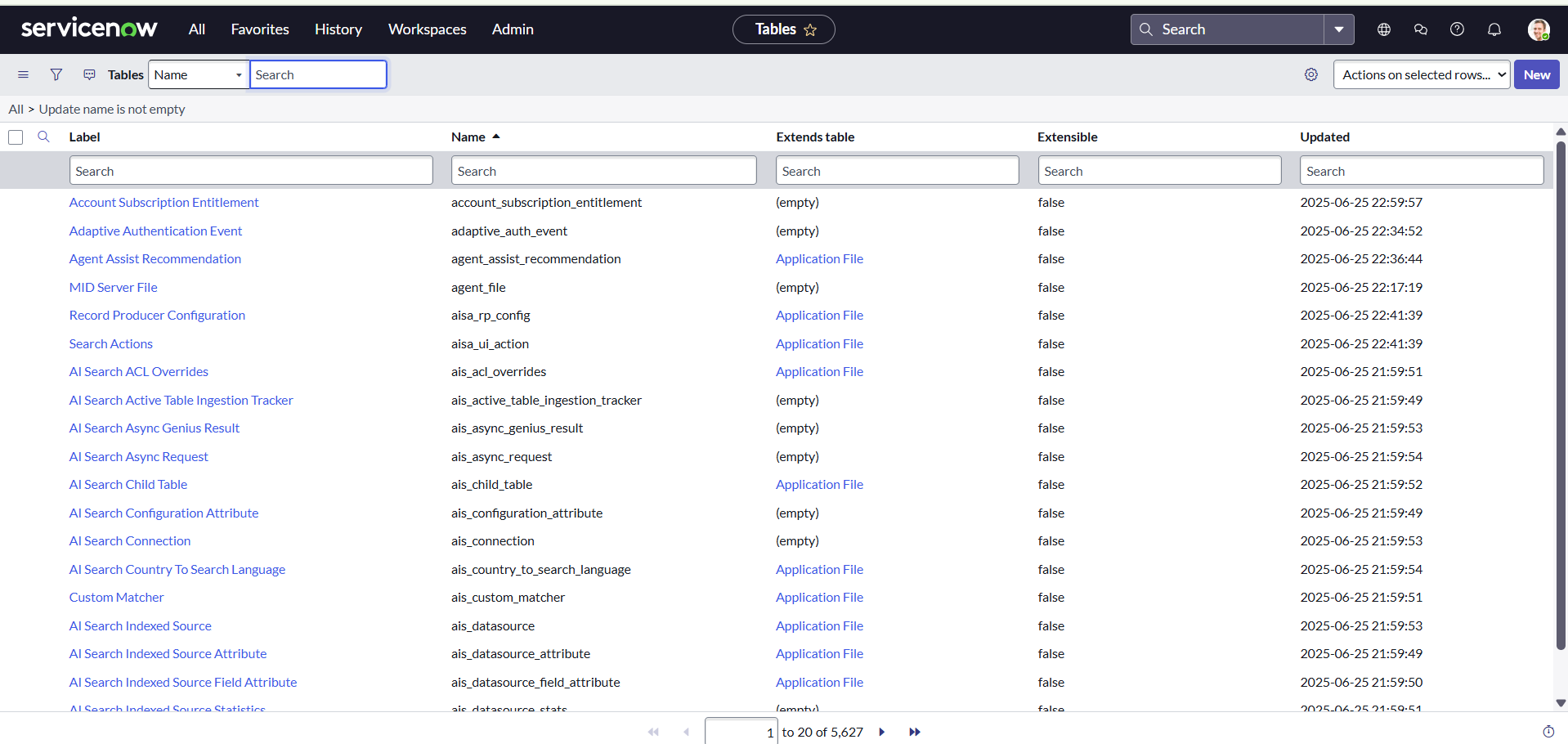
·   **Name**: Name of the table ------

·   **Label**: Backend name of the table------

·   **Auto-generate schema**: Leave it checked if you'd like ServiceNow to auto-generate schema fields.

· Click **Submit** to create the table.







**Creation of fields**

In ServiceNow, fields are created at the **table** level. To create a field, you first need to identify the table where the field will reside.

1. In the **Application Navigator** (left-side panel), type **Tables** in the search bar.

2. Under **System Definition**, click **Tables**. This will take you to a list of all tables in the system.

**Select the Table to Add the Field**

·   From the list of tables, search for and select the **table** you want to add a field to. For example, if you want to add a field to the **Network database**  table:

1. Type "**Network database**" in the search box or scroll through the list.

2. Click on the **Network database** table name. You’ll now see a list of all fields (columns) associated with the **Network database** table.

**Open the Table's Columns**

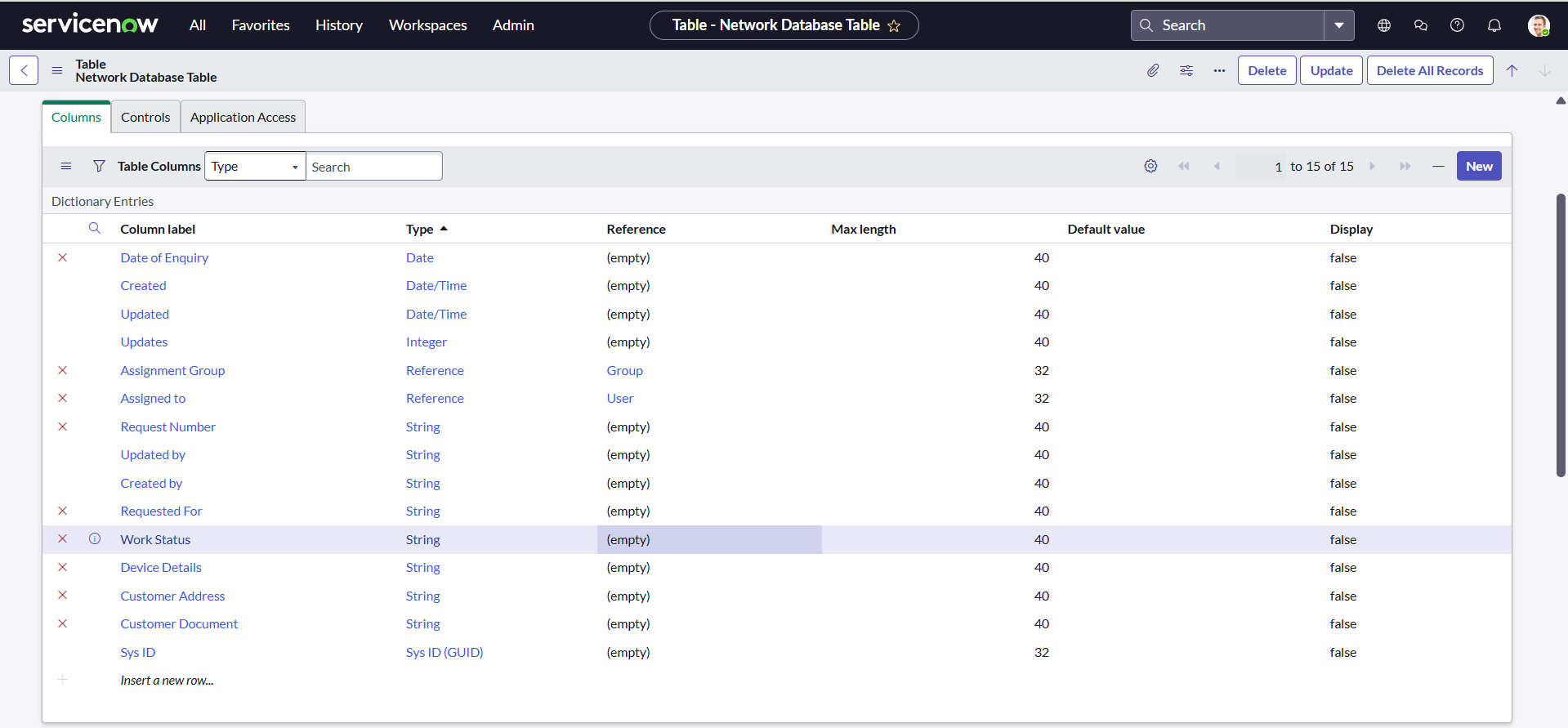
·   After selecting the table, you'll be brought to a view that lists all the columns (fields) that currently exist on that table.

·   To create a new field (column), go to the **Columns** tab (this is where all fields for the selected table are listed).

**Create a New Field**

1. In the **Columns** tab, click the **New** button located at the top-right corner of the page to create a new field.

2. You’ll now be prompted with a form where you need to define the new field. The following fields need to be filled out:



**Define Field Properties**

Fill in the following details for your new field:

**1. Column Label (Field Label)**

·   **Description**: This is the name that will be displayed on the forms, lists, and records.

·   **Example**: Customer Name

**2. Column Name**

·   **Description**: This is the internal name of the field and is auto-generated based on the column label. It should be unique for each field. Do not manually edit this unless necessary.

·   **Example**: customer\_name

·   **Description**: The type of field determines the kind of data it will store. You need to choose the correct type based on the data you want to store (e.g., text, number, date, etc.). Some of the most common types include:

o   **String**: For short text values (e.g., name, description).

o   **Integer**: For numbers without decimals (e.g., age, number of items).

o   **Choice**: A dropdown list of options.

o   **Reference**: A field that links to another table (e.g., linking to a User table).

o   **Boolean**: A true/false checkbox.

o   **Date**: For a date picker field.

o   **Date/Time**: For both date and time.

·   **Example**: String, Choice, Reference

**3.  Max Length (Optional)**

·   **Description**: If you are creating a string-type field, you can specify the maximum length of the text allowed.

·   **Example**: 255 characters (default length for a string field).

**4. Mandatory**

·   **Description**: Check this box if the field should be required when creating or updating records.

·   **Example**: For a "Customer Name" field, this might be required.

**5. Default Value (Optional)**

·   **Description**: You can set a default value for the field if desired. This value will appear automatically when creating a new record.

·   **Example**: Set the default value to "New Customer" for a "Customer Name" field.

**6. Read-Only**

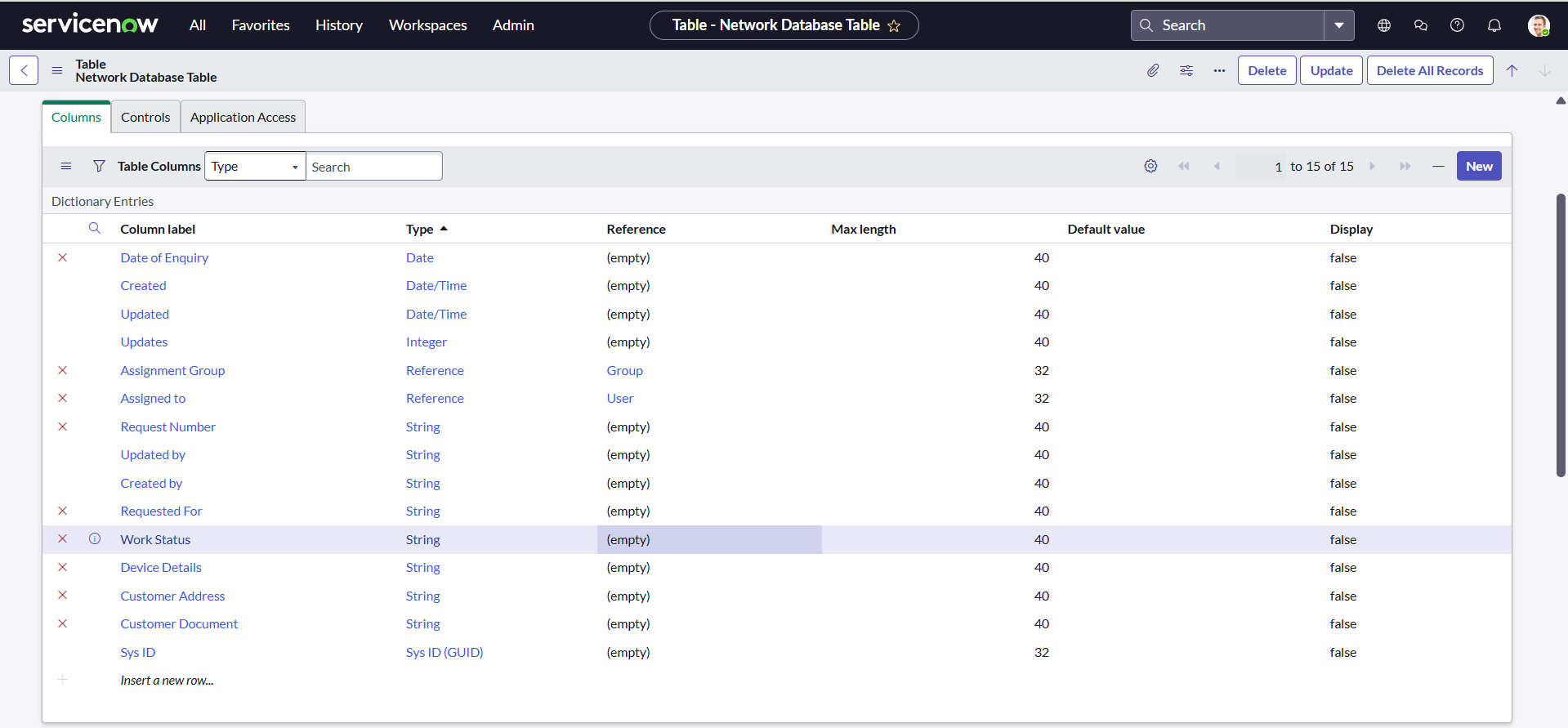
·   **Description**: Check this box if the field should be read-only (users cannot modify its value). This is commonly used for calculated or system-generated fields.

·   **Example**: "Created Date" or "Record Number".

**7: Save the Field**

·   Once you’ve configured all the necessary field properties, click **Submit** or **Save** to create the field.

·   After saving, ServiceNow will create the new field and add it to the list of columns for the selected table.



**Add the Field to a Form (Optional)**

After creating the field, you may want to add it to a form so that users can view or update it.

1. To do this, navigate to **System UI > Forms** in the application navigator.

2. Select the **form** you want to modify (e.g., Incident form).

3. Open the **Form Designer** (click on the "Design" icon).

4. From the **Field Navigator** on the left side, search for the new field you created.

5. Drag the field onto the form layout where you want it to appear.

6. Click **Save** or **Publish** to apply the changes.

**Test the New Field**

·   Go to a record in the table where the field was added (e.g., create a new incident or record).

·   Check if the new field appears on the form.

·   Verify the field behaves as expected (e.g., required, read-only, etc.).

**Key Field Types in ServiceNow:**

·   **String**: Short text input (e.g., a name, description).

·   **Integer**: Whole numbers.

·   **Choice**: Dropdown list with predefined options.

·   **Reference**: A reference field to another table (e.g., referencing an **User** table).

·   **Date**: A date picker.

·   **Date/Time**: A combination of date and time.

·   **Boolean**: Checkbox (True/False).

·   **Currency**: Currency field with monetary values.

**Additional Tips:**

·   **Field Data Types**: Make sure you choose the correct field type based on the type of data you want to store (e.g., Text, Integer, Date).

·   **UI Policies/Client Scripts**: These can be used to make fields visible, read-only, or mandatory based on certain conditions.

·   **Naming Conventions**: Follow proper naming conventions for field labels and column names to maintain consistency.

**Step3: Request Approvals Creation(Related List)**

**Creation of Related List**

Navigate to **System Definition > Relationships**.

·   Click **New** to create a new relationship.

·   Fill in the following details:

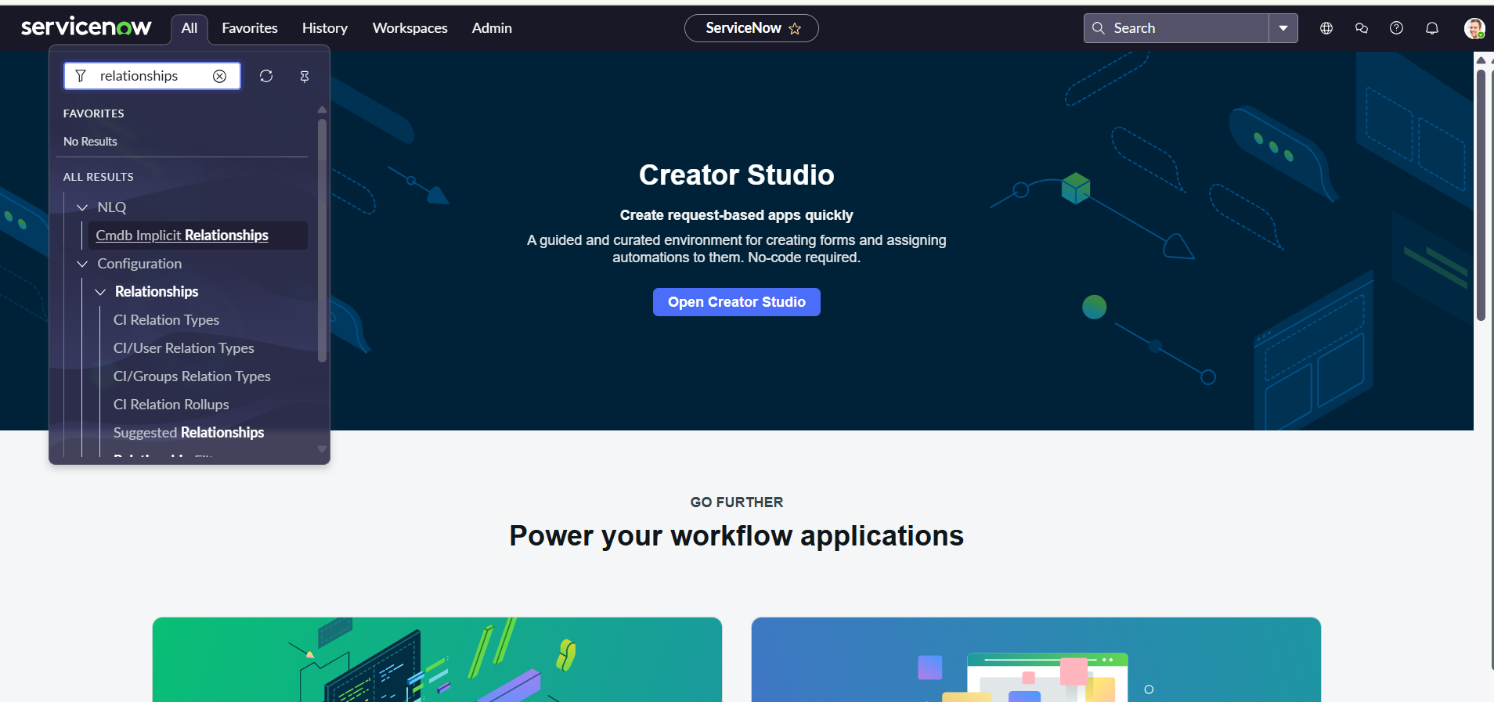
o   **Name**: Approval Request

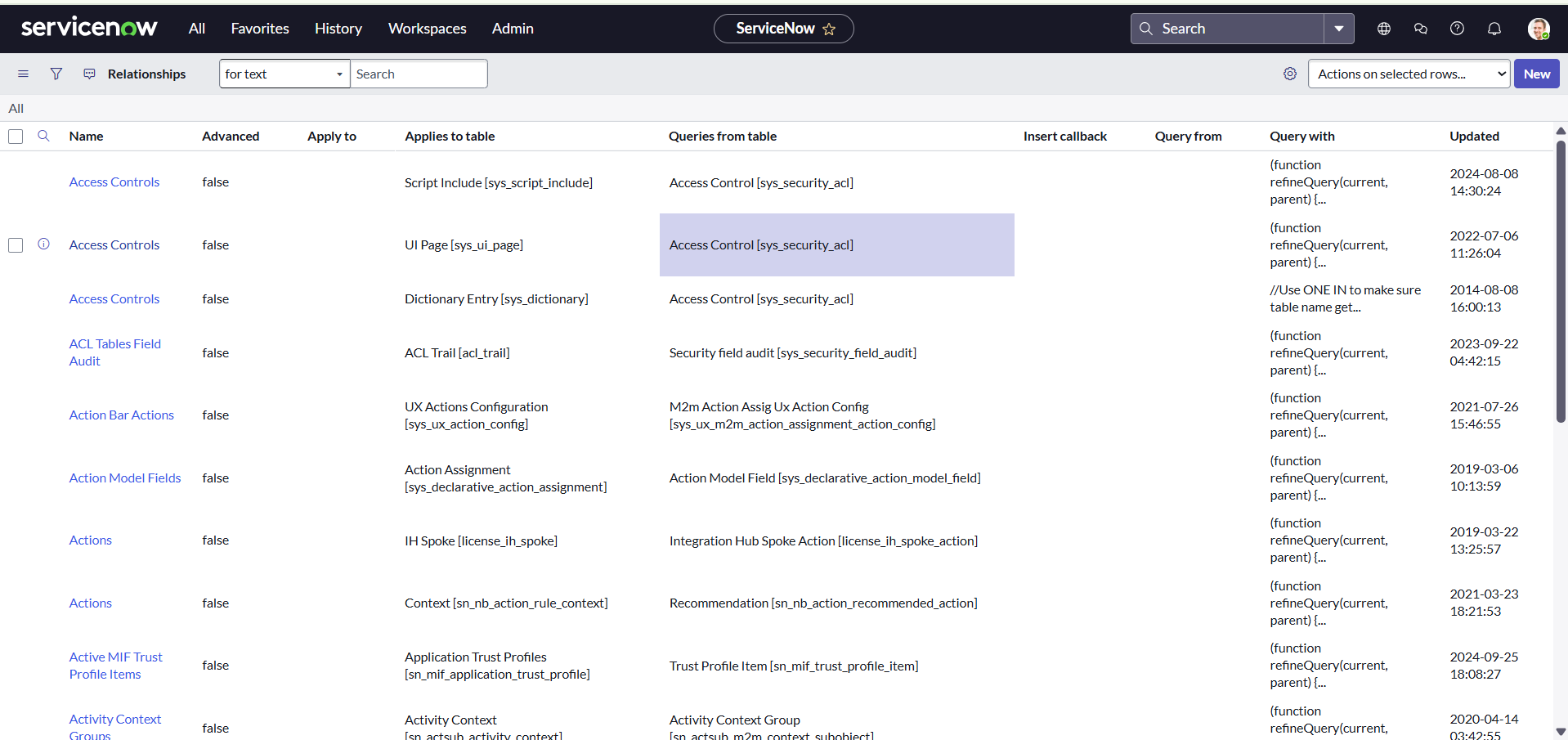
o   **Applies to Table** : Network Database table.

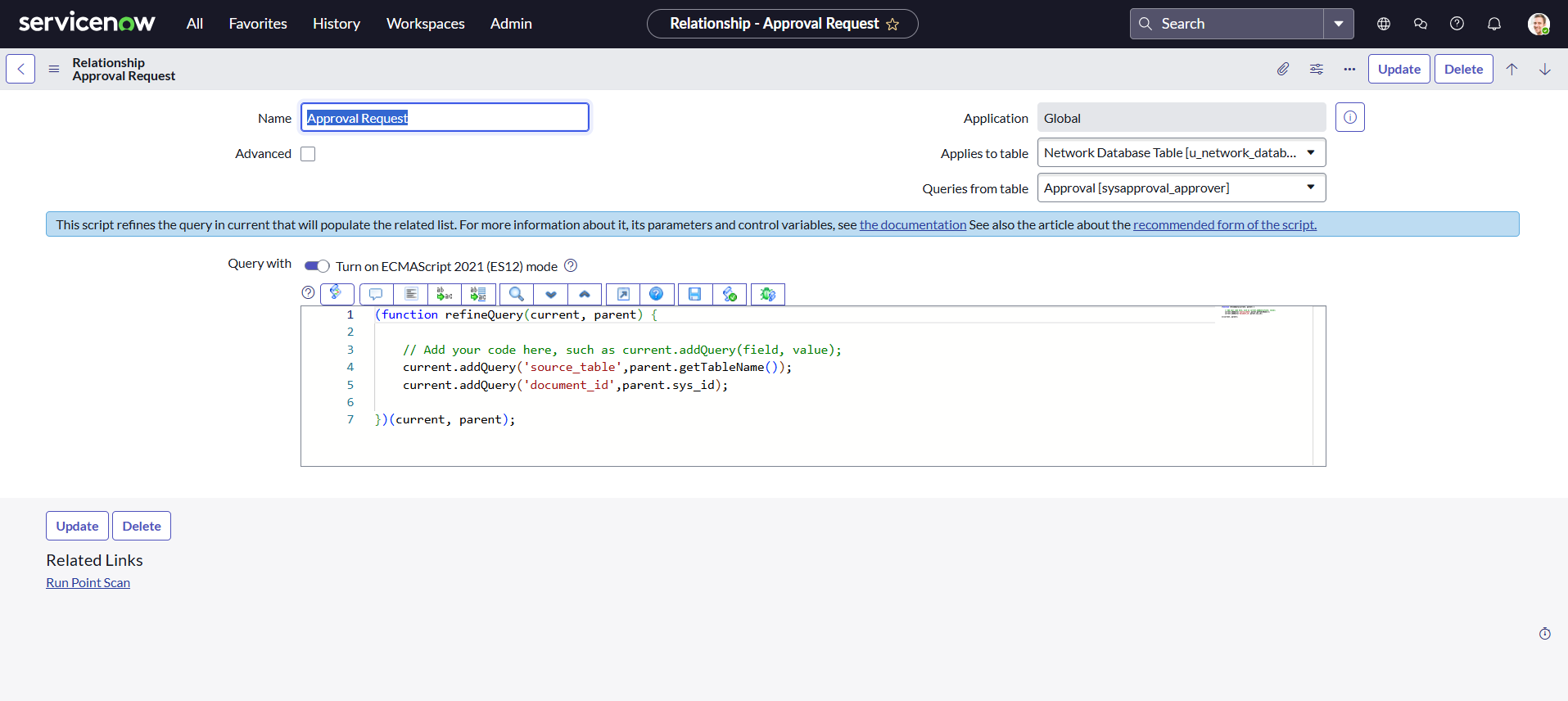
o   **Queries from Table** : Sysapprovals table.

o   **Active**: Make sure it's set to **True**.

·   Save the relationship.







**Adding Related List to the Table**

You can create a **Related List** on a form to display the related records. This helps in easily viewing the relationships between records.

·   Navigate to **Form Designer** for the table where you want to show related records.

·   Add a **Related List** widget to the form.

·   Select the **Related List** you want to show

**Step4: Overview of flows,Actions in Flow Designer**

**Flow Designer Overview**

Flow Designer allows you to automate business processes by designing, testing, and implementing flows that automate tasks, approvals, notifications, and more, across different ServiceNow applications.

**Key Features:**

·   No-code interface for building automation.

·   Reusable Flow Actions to create modular components.

·   Ability to automate processes across multiple tables and integrate with other systems.

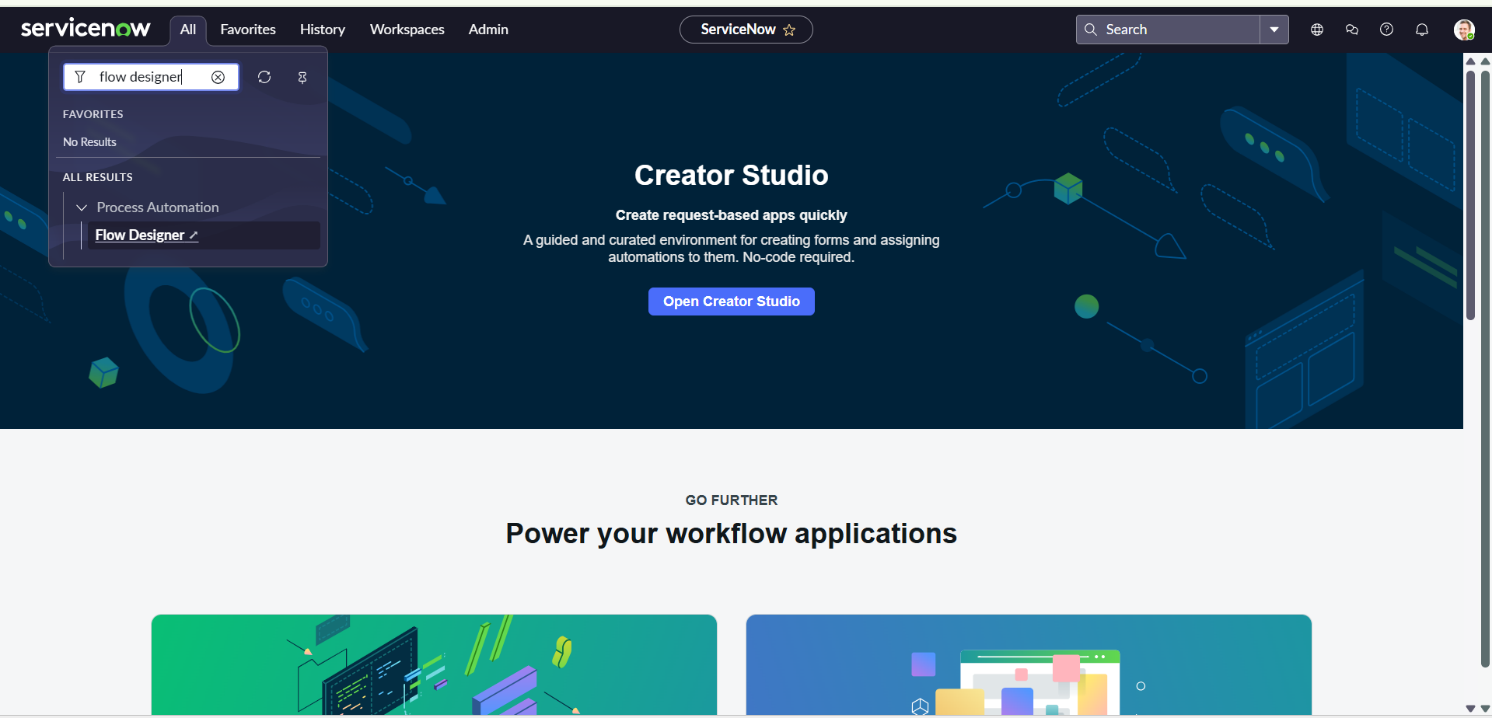
·   Conditional logic, approvals, notifications, and integrations can be easily included in your flows.

·   Full integration with ServiceNow Orchestration for complex automation.

**Navigating to Flow Designer**

To access **Flow Designer**:

·   Go to **Flow Designer** by typing Flow Designer in the left-hand application navigator, or navigate through **All > Flow Designer**.



**Flow Designer Components**

**Key Components in Flow Designer:**

1. **Triggers**:

o   **Record Trigger**: Runs when a record is created, updated, or deleted in a specific table.

o   **Scheduled Trigger**: Runs at a specific time interval or on a schedule.

o   **Custom Event**: Triggered by custom events.

2. **Actions**:

o   Actions define what happens when the flow is triggered. Common actions include:

§  **Create Record**: Create a new record in a table.

§  **Update Record**: Modify an existing record.

§  **Send Notification**: Send an email, SMS, or other notifications.

§  **Run Script**: Execute custom scripts for advanced logic.

3. **Data Pills**:

o   These are dynamic references to data from records or previous steps in the flow. They are used to populate action inputs.

4. **Conditions & Decisions**:

o   Conditions help in making decisions in a flow based on data, which can control the flow’s behavior (e.g., send an approval notification if a specific condition is met).

5. **Flow Logic**:

o   Includes decision points, loops, and waits for conditions to add complex logic to the flow.

**Creating a Flow in Flow Designer**

**Steps to Create a Flow:**

1. **Open Flow Designer**:

o   Go to **Flow Designer > Flows**.

2. **Click on New**:

o   This will start the process of creating a new flow.

3. **Define Flow Properties**:

o   **Name**: Provide a name for your flow - Network Request.

o   **Table/Application**: Choose the target table/application for the flow –Application–Service Catalog.

o   **Trigger**: Define when this flow should run when a request is created.

o   **Description**: Optional but helpful for understanding the purpose of the flow.

4. **Set a Trigger**:

o   The **Trigger** defines when the flow is initiated. Common triggers include:

§  **Record Created**: When a record is created in a specific table.

§  **Record Updated**: When a record is updated.

§  **Scheduled**: When a flow should run on a schedule.

§  **Custom Event**: Triggered by a custom event (e.g., a certain event happening in the system).

* Select the relevant trigger for your flow (when a **Request** record is created).

**Adding Actions**

* After defining the trigger, you can add **actions** that will be executed when the flow is triggered. Some

common actions include

1. **Get Catalog Variables:**

* In ServiceNow Flow Designer, the "Get Catalog Variables" action is used to retrieve values from a catalog item or record producer request. This action is especially helpful when you need to use user-inputted variables (from a catalog item) within a flow
* **How to Use “Get Catalog Variables” in Flow Designer**

1. Open Flow Designer
2. Navigate to: Flow Designer ? Create or open a flow.
3. Ensure the Trigger is Catalog-Based
4. Choose a trigger like Catalog Item Requested or Catalog Task Created.
5. Add Action ? Get Catalog Variables
6. Click + under your trigger or previous action.
7. Choose Action ? **"Get Catalog Variables"**.
8. Select the record input (Requested Item [sc\_req\_item]) from the Data pill.
9. Set the Catalog item– Network Request
10. You’ll typically input the Requested Item Record from the trigger.

**Use Output Variables**

The output will include all the catalog variables submitted with the request**.**

**2. Create Record:**

* In ServiceNow Flow Designer, the "Create Record" action is used to create a new record in any table (e.g., Incident, Task, Custom Table, etc.) during the execution of a flow.
* This is one of the most powerful and commonly used actions in Flow Designer, allowing you to automate the creation of tasks, incidents, change requests, approvals, and more.

**How to Use "Create Record" in Flow Designer**

1. **Open or Create a Flow**
   * Go to **Flow Designer** (Flow Designer > Designer)
   * Open an existing flow or click **New**.
2. **Add a Trigger (if needed)**
   * e.g., **Record Created**, **Catalog Item Requested**, etc.
3. **Add Action ? "Create Record"**
   * Click the **"+"** under the trigger or another action.
   * Select **Action** ? Search for **"Create Record"**.
4. **Configure the Action:**
   * **Table**: Select the table where you want the new record created (Network database table).
   * **Fields**: Set the field values you want on the new record using static values, data pills, or dynamic inputs.

**3. Send Email Action:**

* The "**Send Email**" action in ServiceNow Flow Designer allows you to send customized emails as part of an automated flow. It’s commonly used to notify users, groups, or stakeholders based on triggers like catalog submissions, record changes, task updates, and more.

**How to Use "Send Email" in Flow Designer**

1. **Open or Create a Flow**
   * Go to **Flow Designer** (Flow Designer > Designer)
   * Open an existing flow or click **New**.
2. **Add a Trigger**
   * Examples: **Catalog Item Requested**, **Record Updated**, **Incident Created**, etc.
3. **Add Action ? "Send Email"**
   * Click the **"+"** button under the trigger or previous action.
   * Select **Action** ? Search for and select **"Send Email"**.
4. **Configure Email Details**
   * **To**: Choose one or more recipients (Users, Groups, Emails). You can use:
     + Data Pills —>Requested For.Caller.email)
     + Static email addresses
   * **Subject**: Write a subject line. Your Request has been Created
   * **Body**: Enter the message body using:
     + Plain text
     + HTML formatting
     + Dynamic data pills (like variables, record fields)
5. **(Optional) Add CC or BCC**
   * Available in the action settings if needed.
6. **Save and Test the Flow**

**4. Ask for Approval**

* In ServiceNow Flow Designer, the “**Ask for Approval**” action is used to request approval from one or more users or groups as part of an automated process. It's commonly used in flows for change requests, catalog items, onboarding, and custom workflows where decisions are required.

**How to Use “Ask for Approval” in Flow Designer**

**1. Open a Flow**

* Go to: **Flow Designer** ? Open or create a flow.

**2. Add Action ? Ask for Approval**

* Click **+** and select **Action**.
* Search for and select **“Ask for Approval”**.
* Select Table/Record– Network Database table.

**3. Configure the Approval**

**A. Who Needs to Approve?**

* **Users**: Select specific users (static or from data pills like Requested For, Manager, etc.)
* **Groups**: Assign to a group. The first responder usually determines the outcome unless changed. I.e group manager.

**B. Approval Record**

* You must associate the approval with a record, typically the trigger record like:
  + Custom Table Record

**C. Approval Details**

* **Short Description**:The approvers will see the request."

**4. Use the Outcome**

* The action outputs an **Approval State** will  be like approved, rejected, or skipped.

**5. Flow Logics:**

* In **ServiceNow Flow Designer**, **Flow Logic** actions are used to **control the flow’s path** based on conditions, iterations, or specific structure. They help you make decisions, loop through data, wait for conditions, and handle errors.

**Using of If Condition:**

* Click the "+" below your previous step (like the approval).
* Choose "Flow Logic" ? "If".

1. **Set the Condition in the If block:**
   1. Click "Add Condition"
   2. Choose a data pill (such as Approval State, variables.reason, or any field).
   3. Set your condition.
2. **Add Actions Inside the If Block**
   1. Inside the **If (true)** block, add actions like:
   2. Create a New table record/Update an existing record
3. **Save and Test**:
   1. After configuring the flow, click **Save** and then **Test** the flow to ensure it behaves as expected.
   2. You can test the flow using sample data or by triggering it manually.
4. **Activate the Flow**:

Once you’ve tested the flow and everything looks good, you can **activate** the flow so that it starts running based on the defined trigger.

**Testing and Debugging Flows**

**Steps to Test and Debug a Flow:**

1. **Testing**:

o   After creating a flow, you can test it by triggering the flow manually or creating a test record that matches your trigger conditions.

2. **Debugging**:

o   Use the **Flow Execution Logs** to debug and track the flow’s execution.

* Go to **Flow Designer > Flows**, select the flow, and review execution logs to identify any issues.

**Best Practices for Flow Designer**

·   **Use Subflows**: For reusable processes, you can create subflows that are called from other flows. This reduces duplication.

·   **Keep Flows Simple**: Avoid overly complex flows. Break up large processes into smaller, more manageable subflows.

·   **Error Handling**: Make sure your flows are capable of handling errors gracefully.

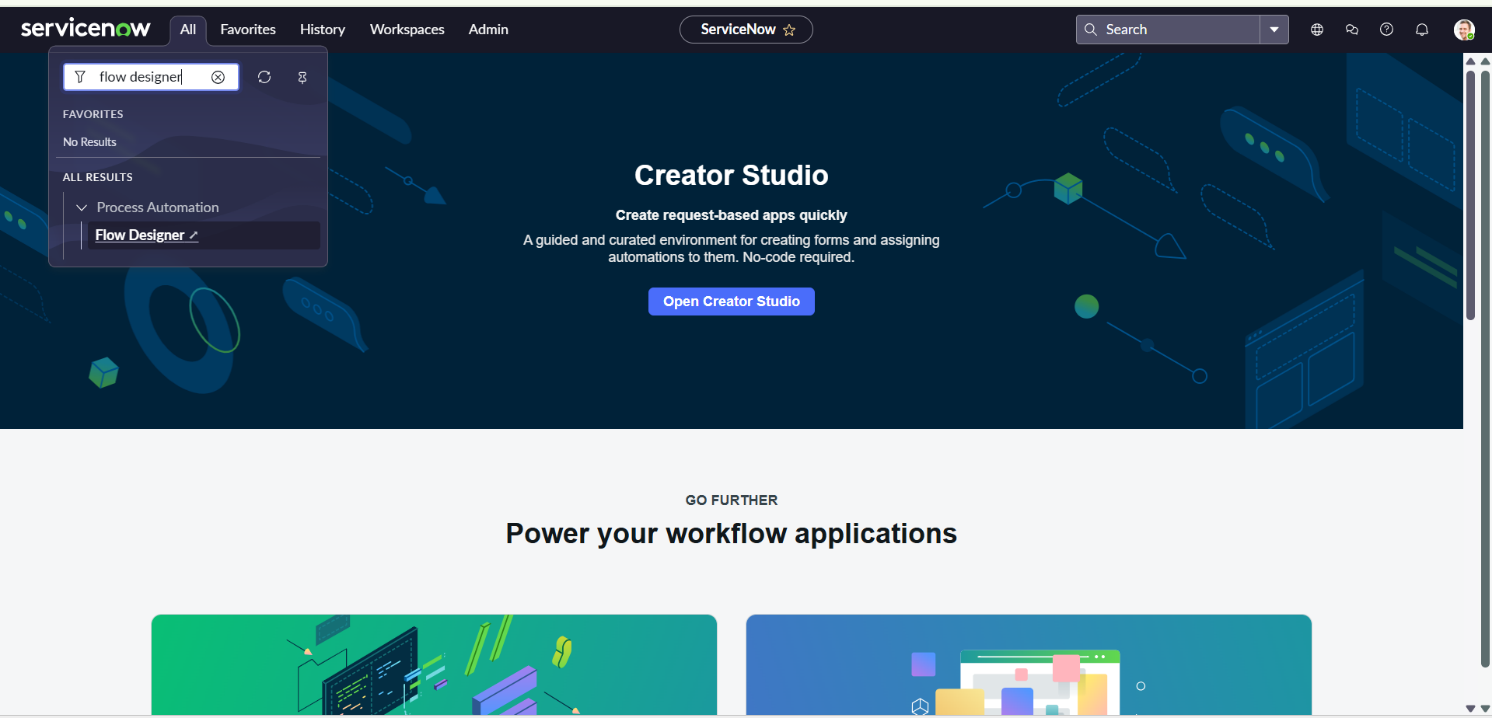
·   **Naming Conventions**: Use clear, consistent naming conventions for flows, actions, and subflows.

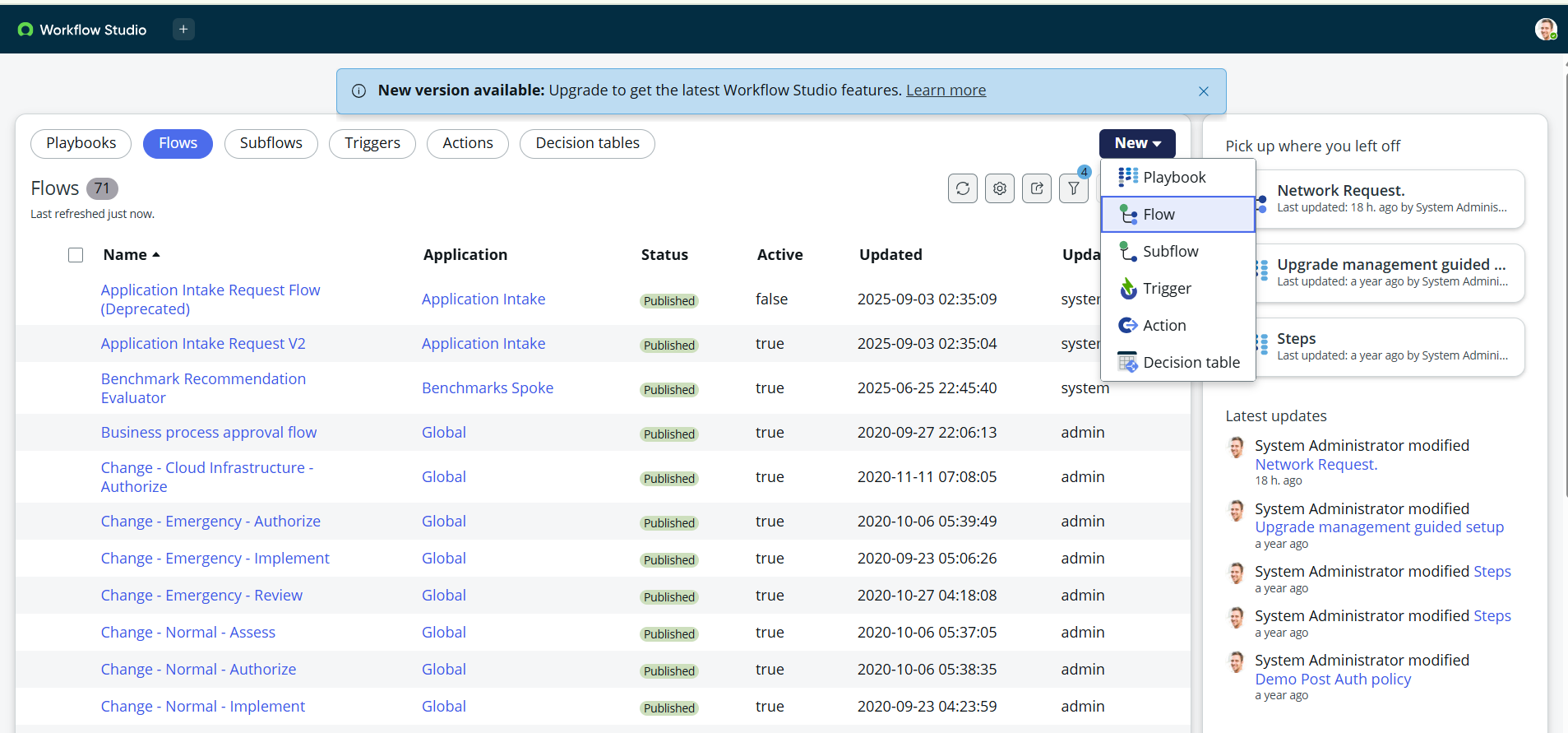
·   **Testing**: Always test your flow thoroughly before activating it in a production environment.

**Step5: Creation & Implementation of flows, Actions in Flow Designer**

**Creation of Flow**

1. Navigate to Flow designer home page
2. Click on New to create  a new flow
3. Provide flow name as Network Request
4. Provide description of flow
5. Click on Build flow.







**Configuring Trigger**

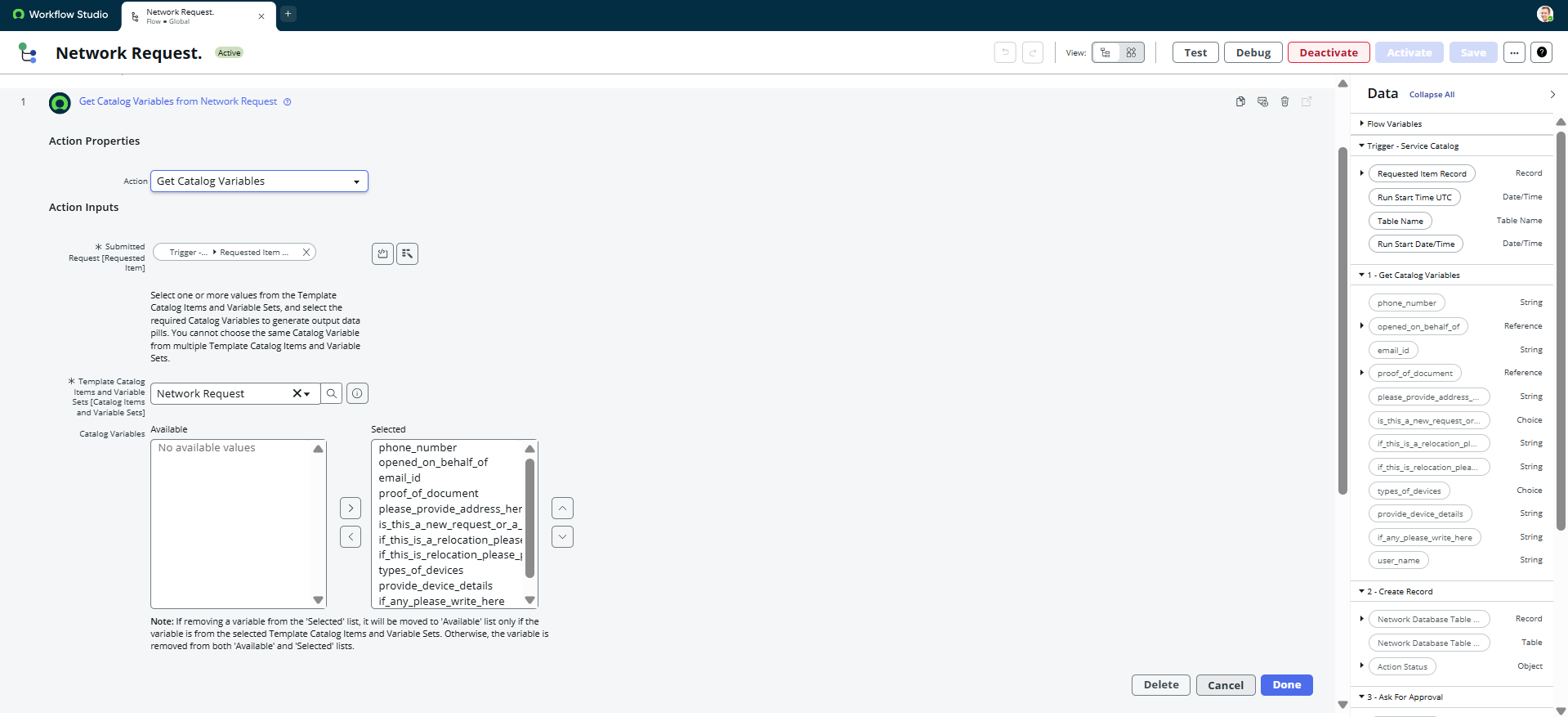
1. Click on (+) Icon to Configure the Trigger
2. Select Trigger as Application >> Service catalog
3. Click on **Done.**

**Configuring Actions**

Click on Actions button to configure new action

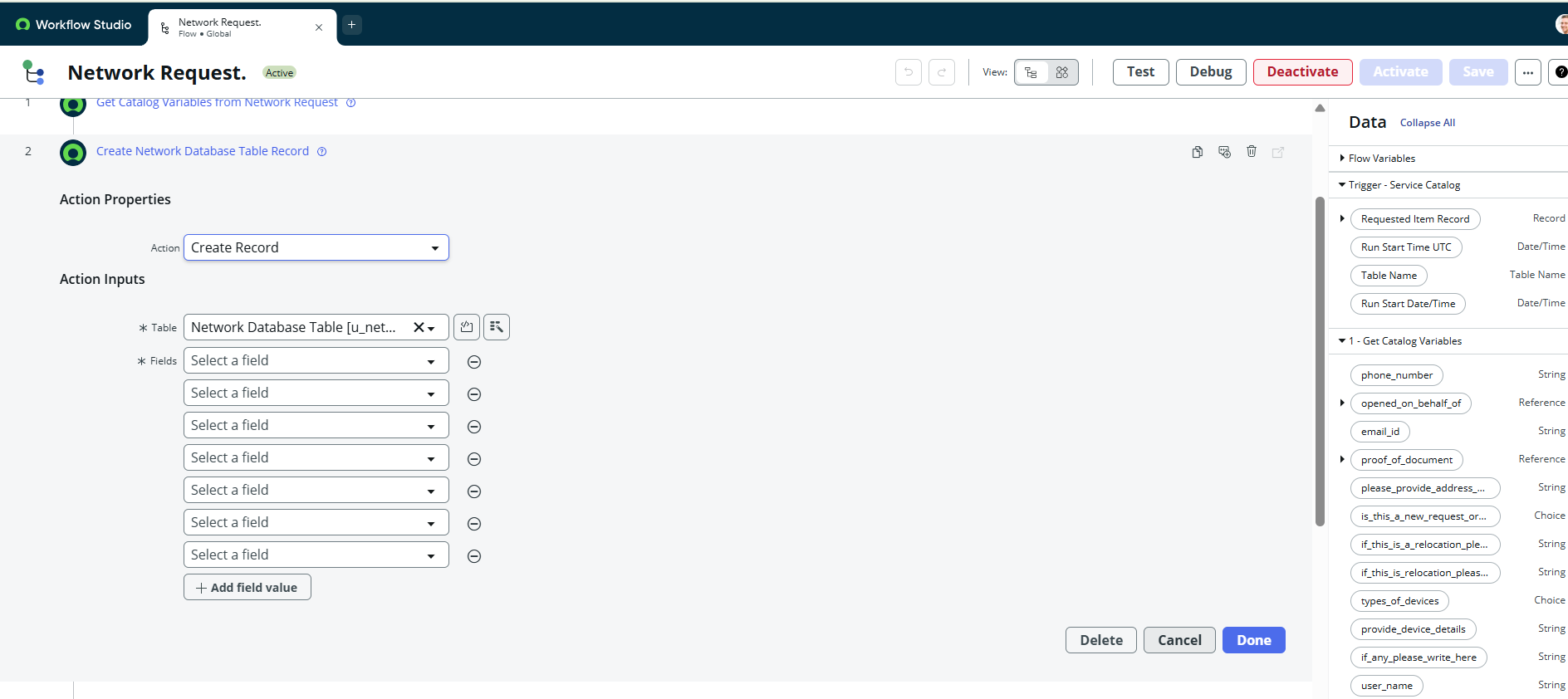
1. **Get Catalog Variables**

* Click on Action, search for Get Catalog Variables
* Select Get Catalog Variables
* Action Inputs>> Trigger>>service catalog>>Requested Item
* Template catalog items >> Select table >> Network Request
* Select the Required Variables and Move to the selected area.
* Click on done



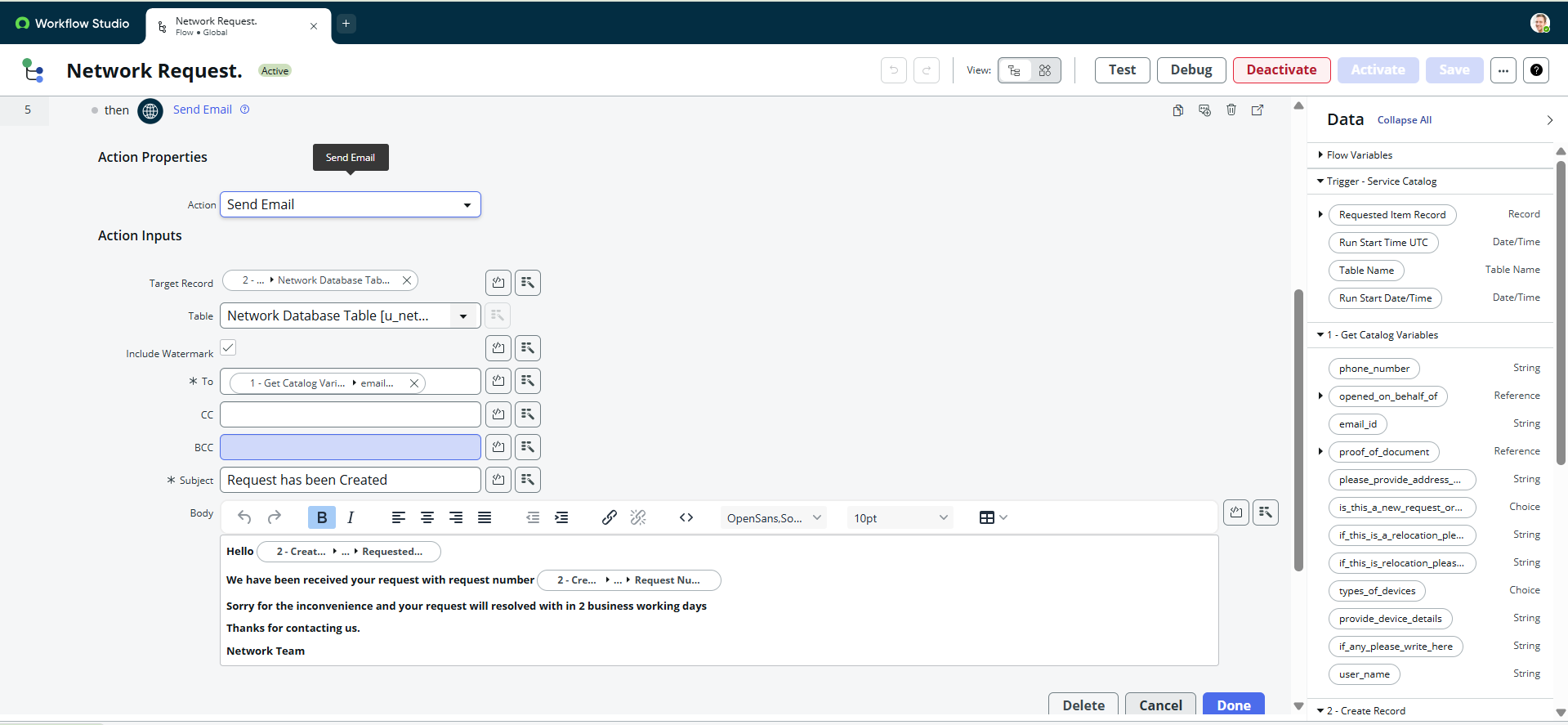
1. **Create Record**

* Select action as Create Record
* Select table as Network Database
* Click on Add fields button to configure the fields
* Configure the Required fields as shown in the below picture
* Click on done



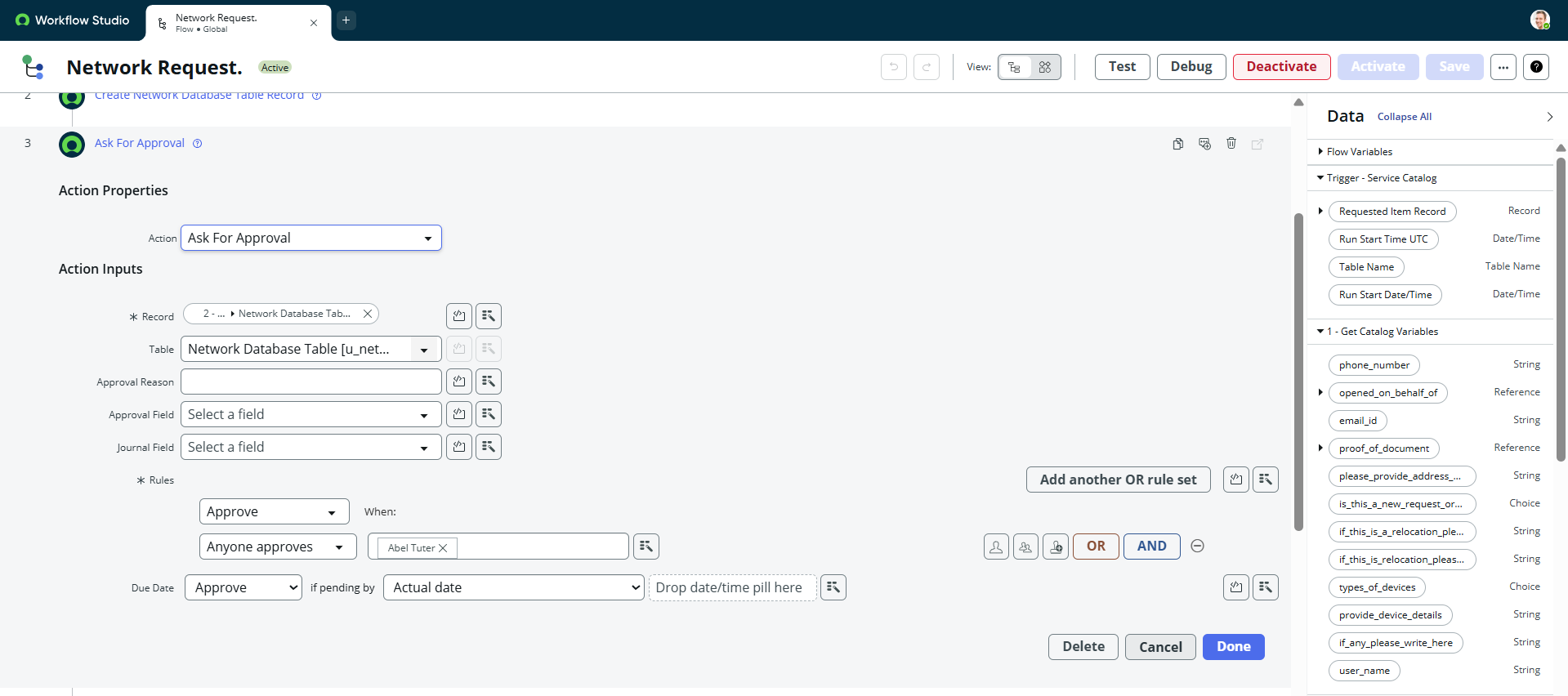
1. **Send Email**

* Select action as Send Email
* Select target record >> Create record>> network database table
* Table will be selected automatically
* Cofigure To, CC, BCC as per our requirements(select static/dynamic)
* Provide Subject & Body as shown in the below picture
* Click on done



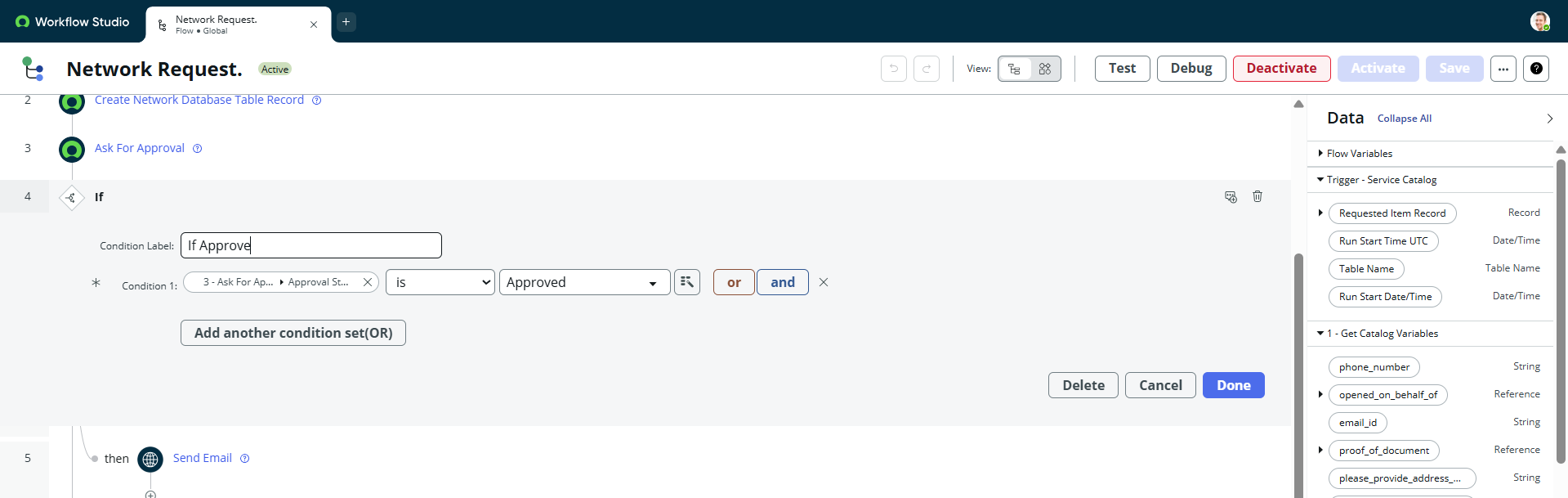
1. **Ask for approvals**

* Select action as Ask for Approval
* Select target record >> Create record>> network database table
* Provide Approval Reason>> Waiting for approval
* Configure approval rules>> Approve, reject, approve/reject
* Select approvals as Anyone approves, everyone approves etc.
* We can select approvals like static/dynamic as shown below
* Click on done



1. **Flow Logic**

* Select action as flow logic and Select If condition
* Apply condition >> Ask for approvals state is **Approved/Rejected** as per requirement
* Click on done

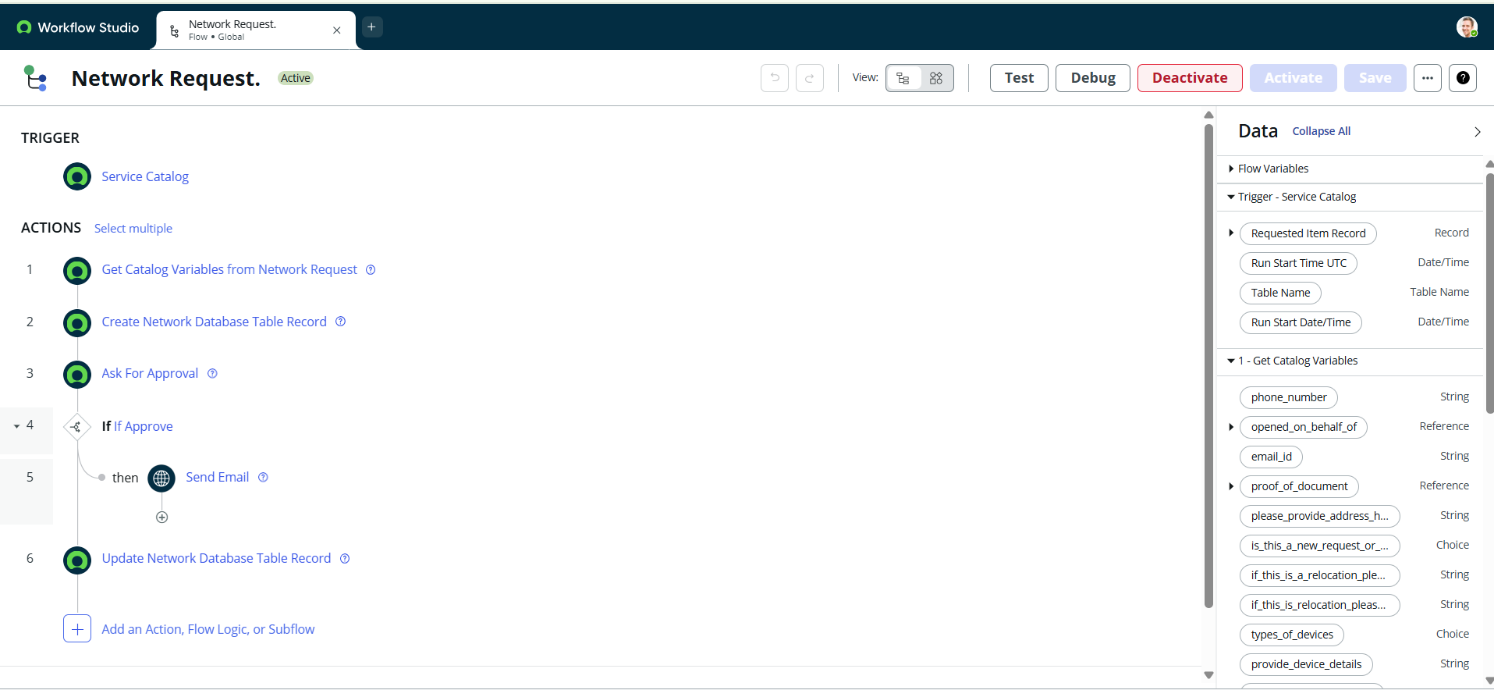


1. **Update Record**

* Select action as Update Record
* Select record as >> create record>> network database
* Table will be selected automatically
* Configure the fields as per requirement, as shown in below
* Click on done



**Flow Chart**

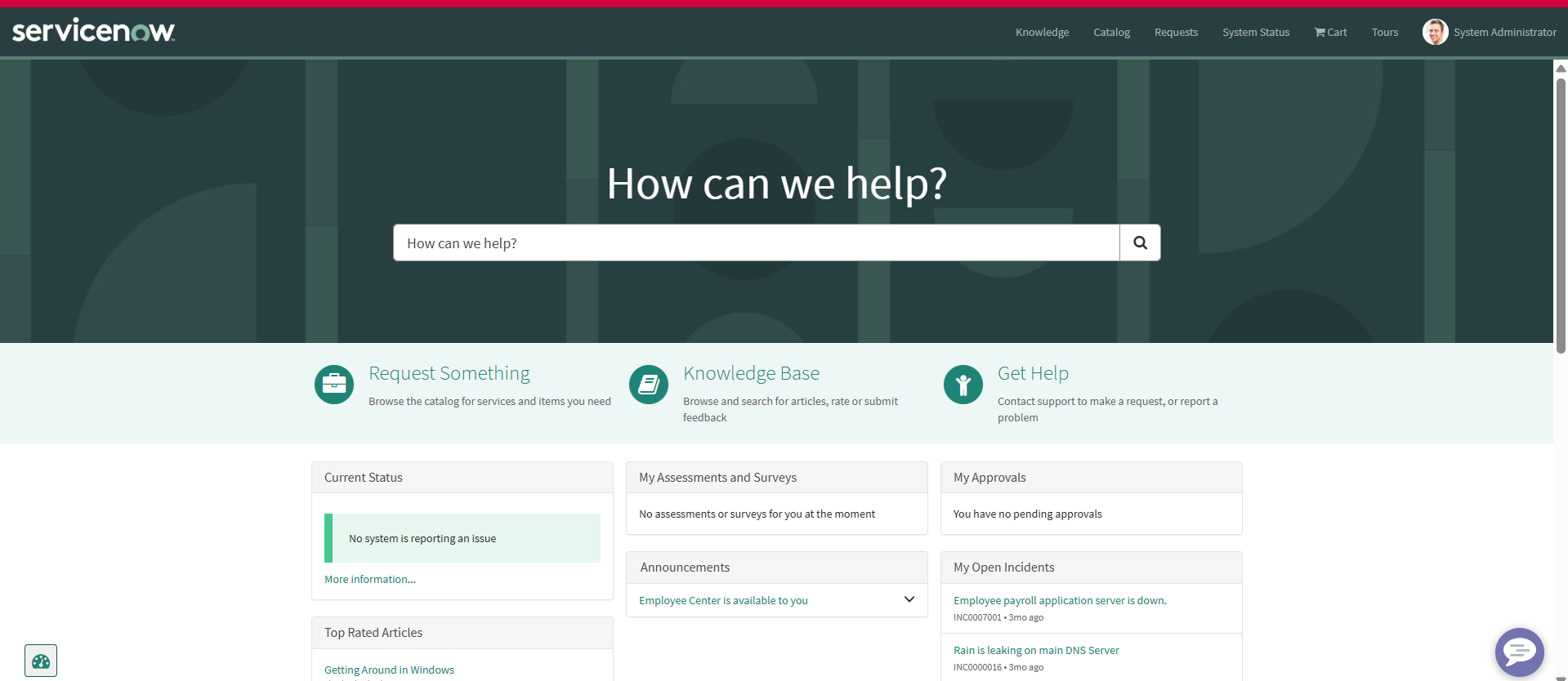
****

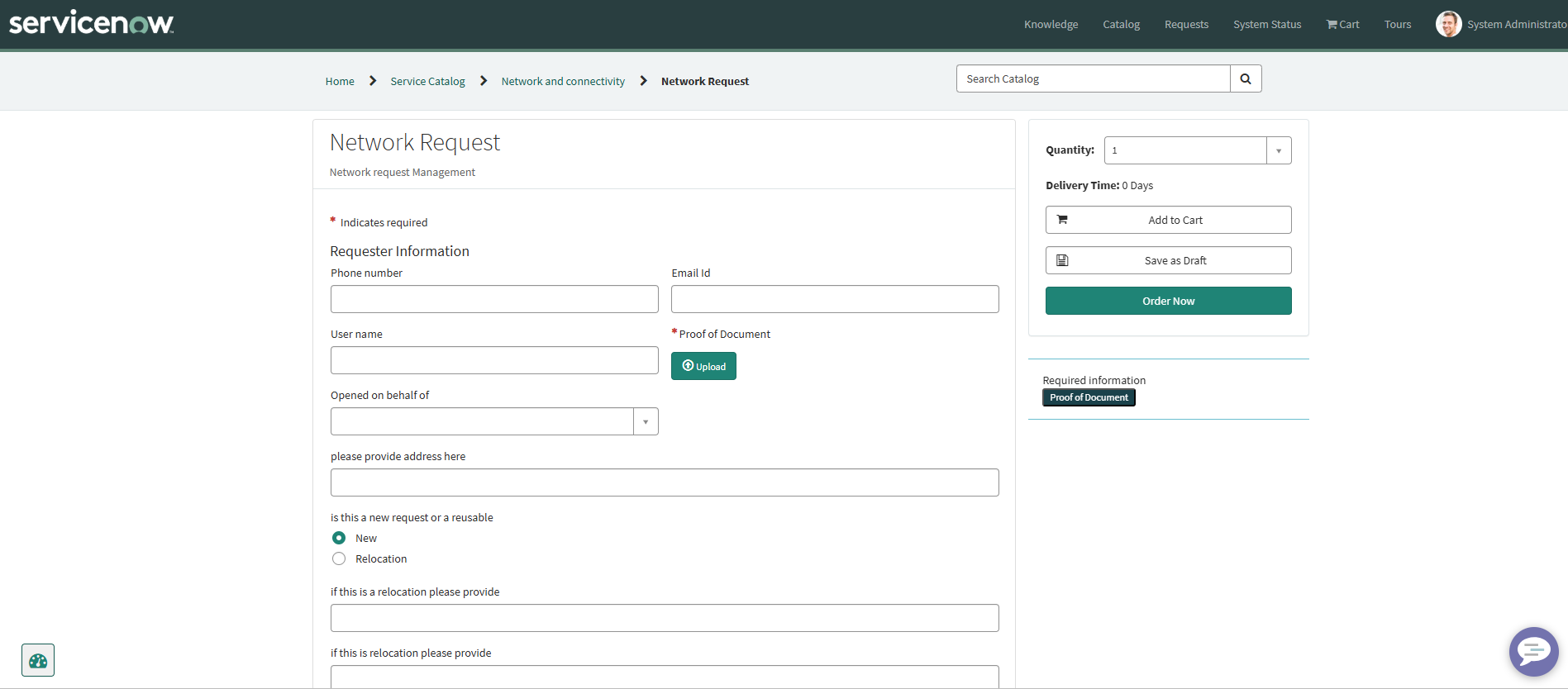
**Step6: Final Testing in End User portal & Instance**

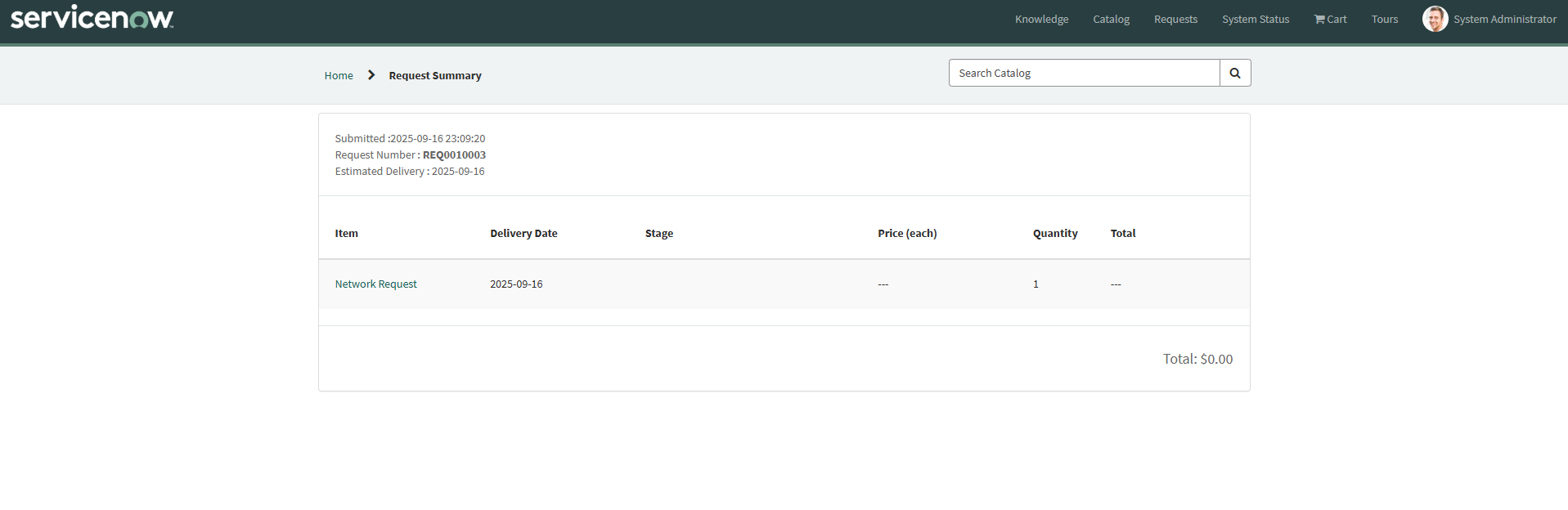
**Testing in Service Portal(End User)**

**Procedure:**

1. Login to ServiceNow PDI
2. Copy the Instance domain ex: [https://dev190678.service-now.com](https://dev190678.service-now.com/).
3. Paste the URL in the Next tab and add Prefix SP to the URL.                                                   ex:<https://dev190678.service-now.com/sp>.
4. Search for Network Requests.
5. Fill the required details and click on submit
6. New Requests will be generated with request numbers and users will get particular emails on the same.



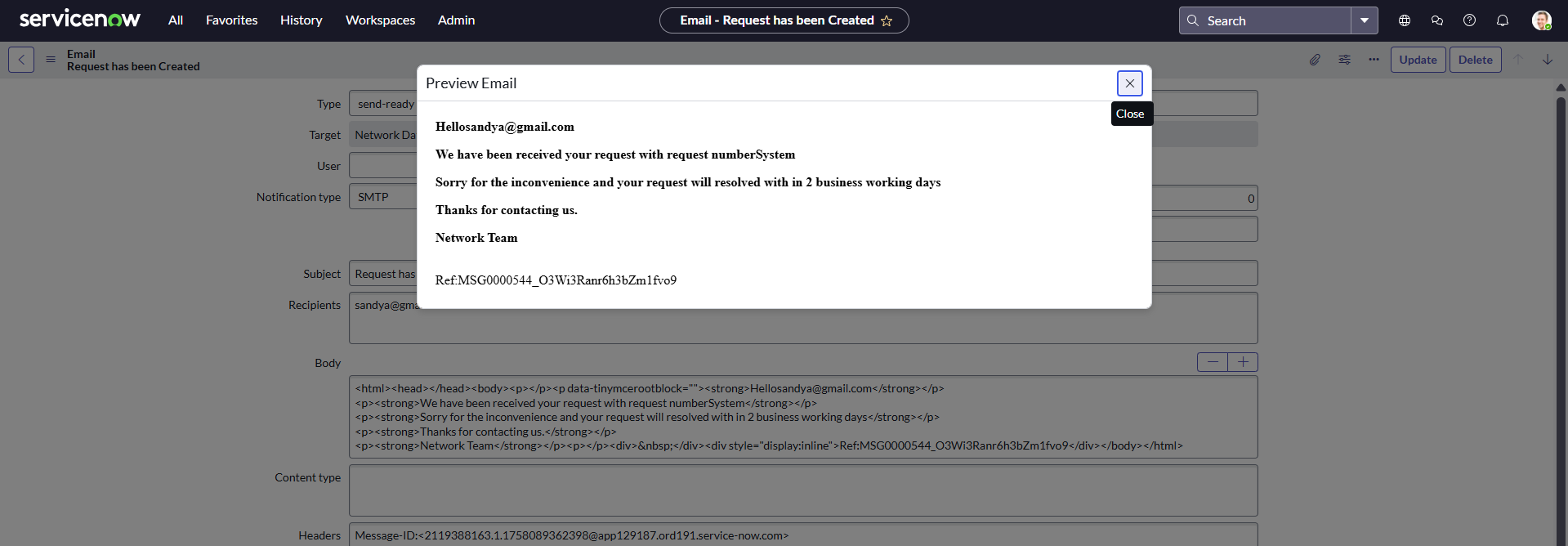




**Testing Emails**

**Procedure**:

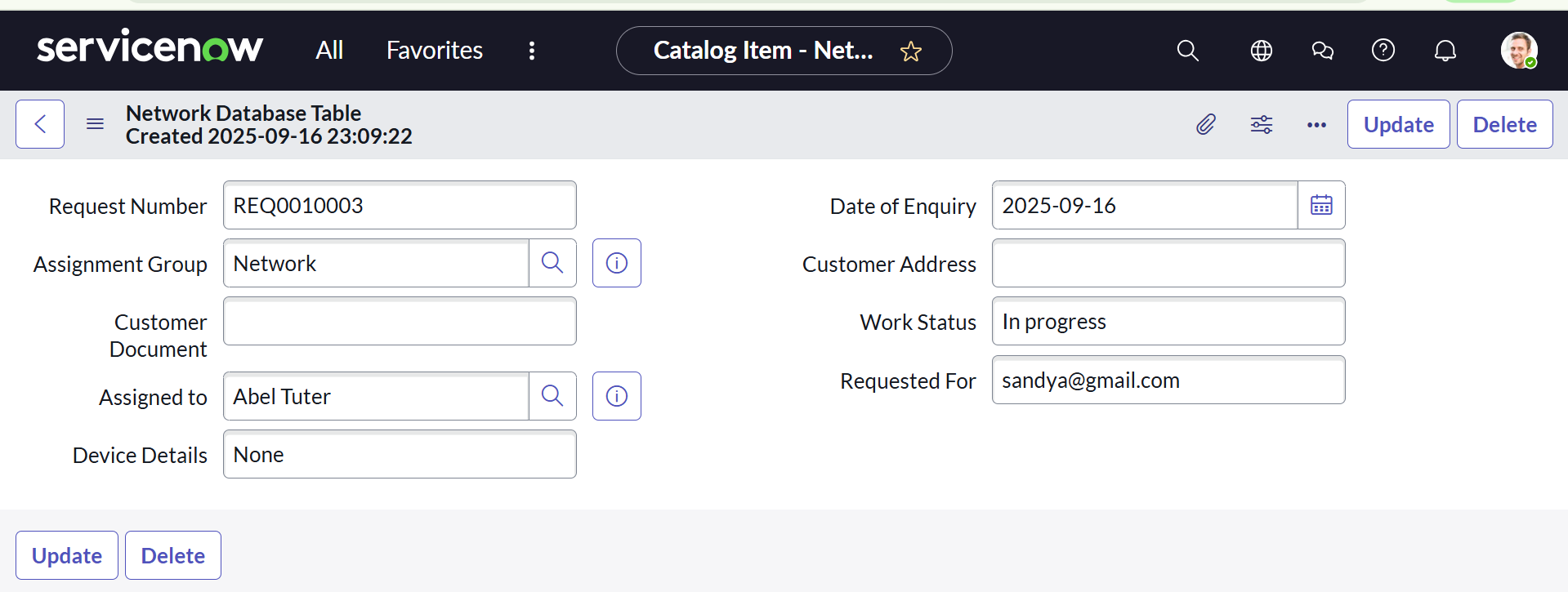
1. Login to ServiceNow PDI
2. System logs>> emails
3. Apply filter>> created on today
4. Search with To, BCC, CC, Subject to get to know what are the emails triggered on the particular request.

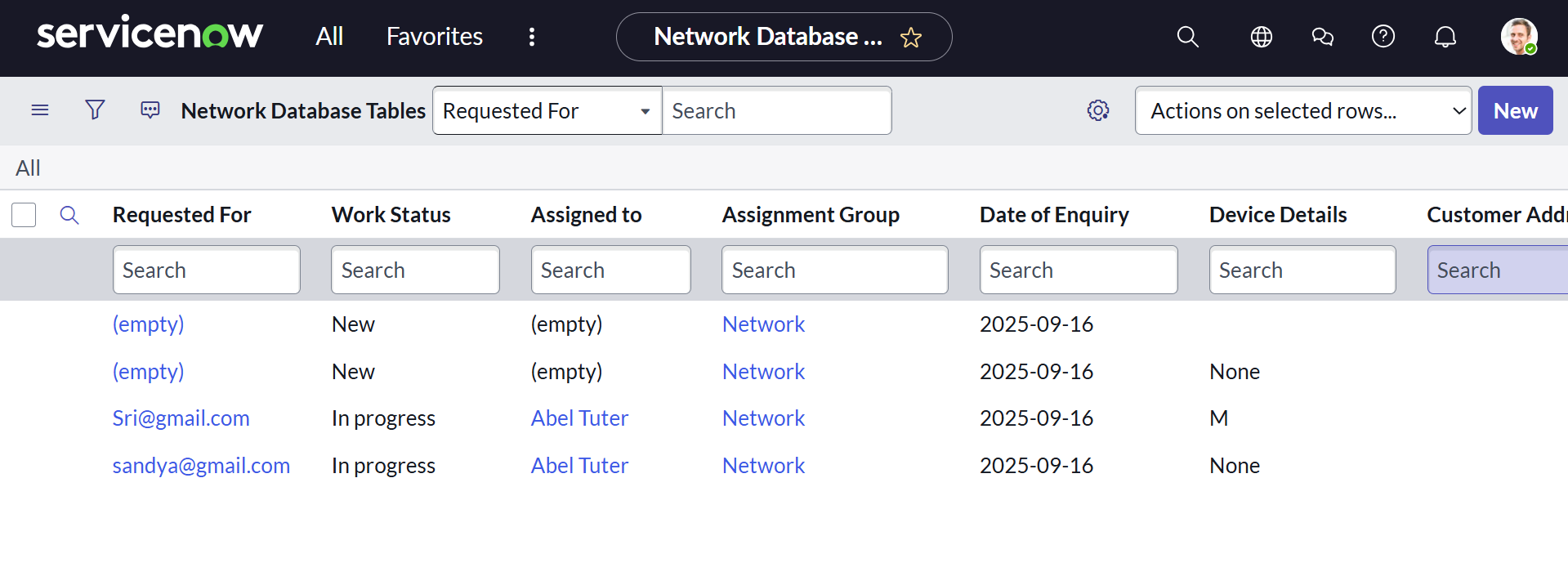


**Testing with Custom Tables**

Procedure:

1. Login to ServiceNow PDI
2. System definition >> Tables>> Network database/Network Task
3. After request is generated, Database and task tables fields are automatically filled by the flow designer configurations
4. Observe the Approvals requests and Changing of States of tables carefully





**Conclusion:**

The Network Request Management system in ServiceNow automates request intake, routing, and fulfillment. Through dynamic forms, role-based approvals, and email notifications, it ensures transparency and efficiency. Optional automation integration further reduces manual workload and risk, enabling faster and more reliable network operations.