

**NAME : TUNGANA RUSHI**

**ROLL NO : 22A51A12D0**

**COLLEGE : ADITYA INSTITUTE OF TECHNOLOGY AND MANAGEMENT**

**PROJECT TITLE :** Automated Network Request Management in ServiceNow

**Automated Network Request Management in ServiceNow**

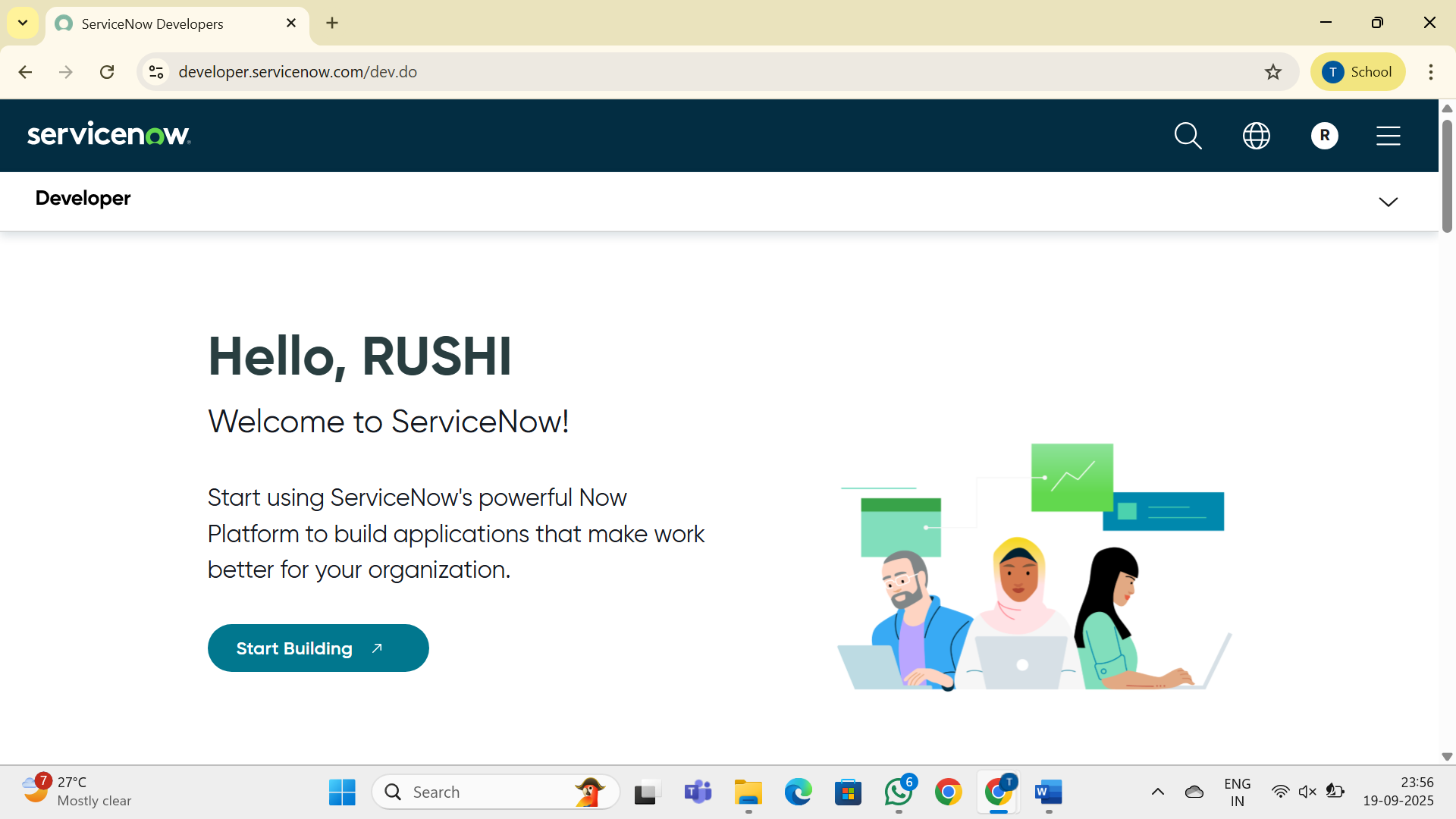
**Project Description:**

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
* **Access the Developer Site**
* Go to **https://developer.servicenow.com/** and sign in with your developer account.
* After a successful sign-in, the landing page shows as below:



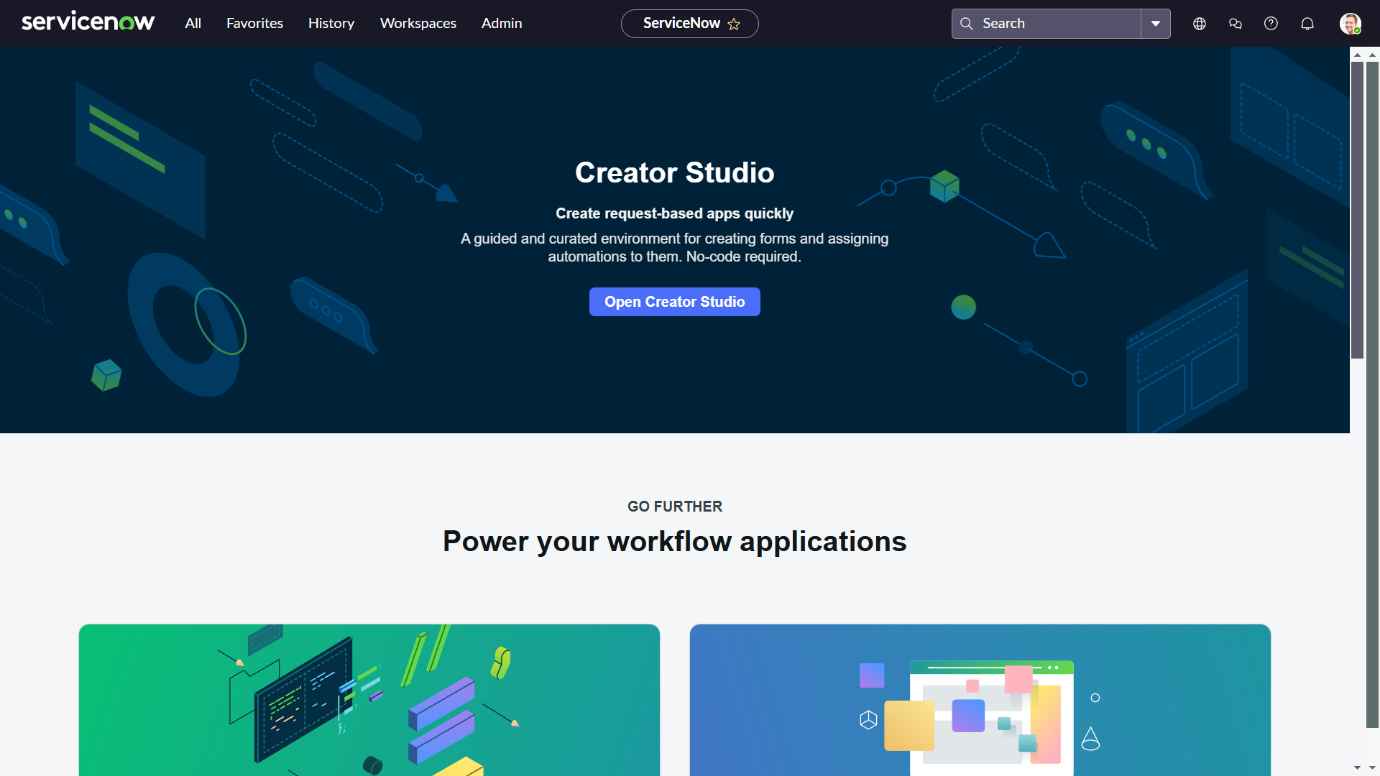
**Start Building an Instance**

On the welcome page, click **Start Building**.

Choose an **Instance location** if prompted (for example: Zurich, Yokohama, or the region closest to you).

Click **Request Instance** or **Start Building** again.

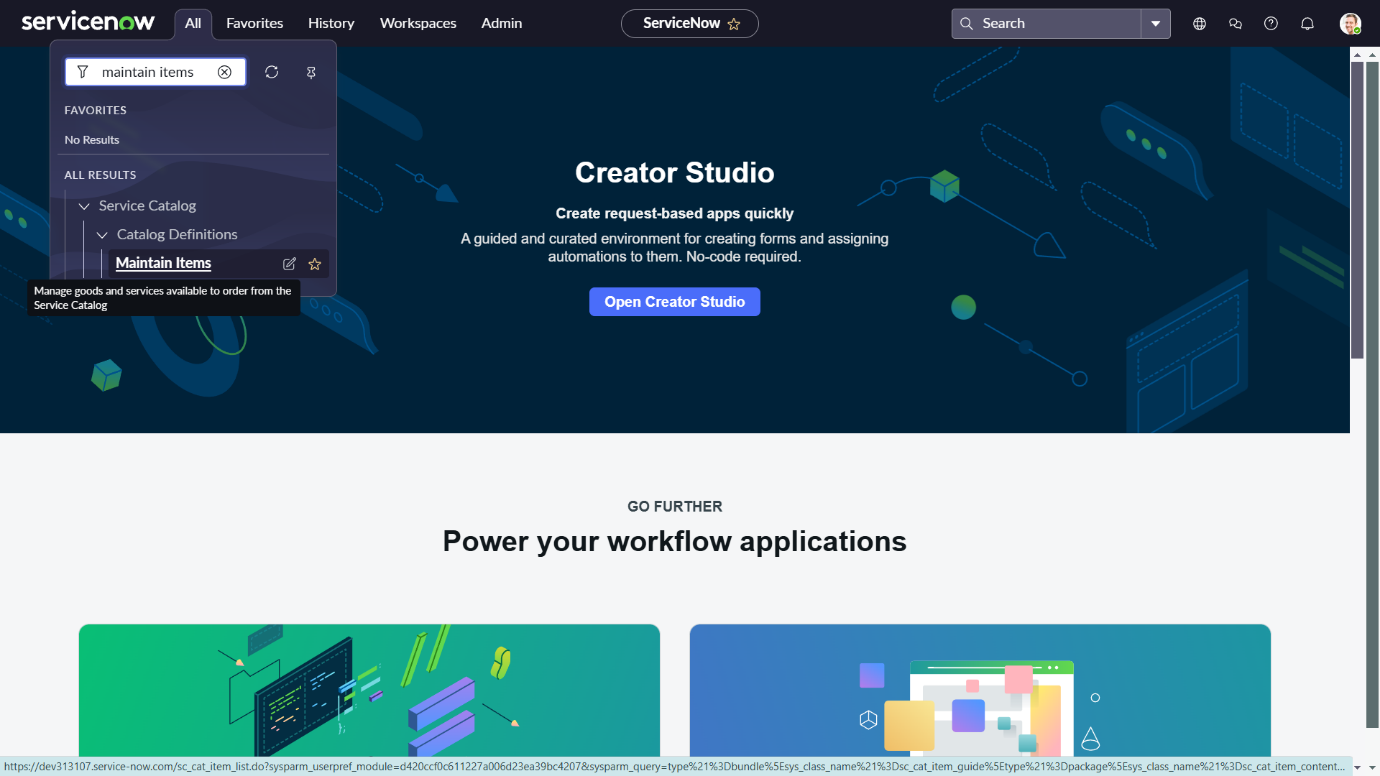
Wait while ServiceNow provisions your **personal developer instance** (this may take a minute or two).

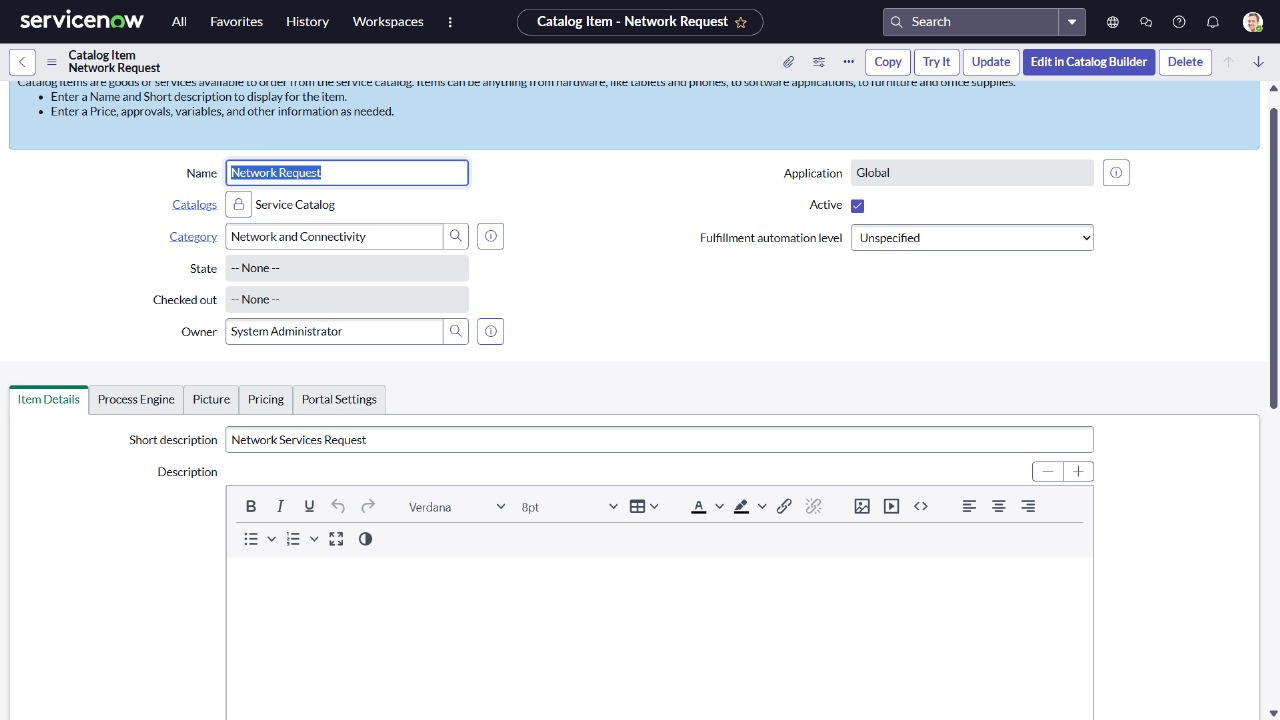


**Service Catalog Creation**

**Creation of service catalog:**

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.

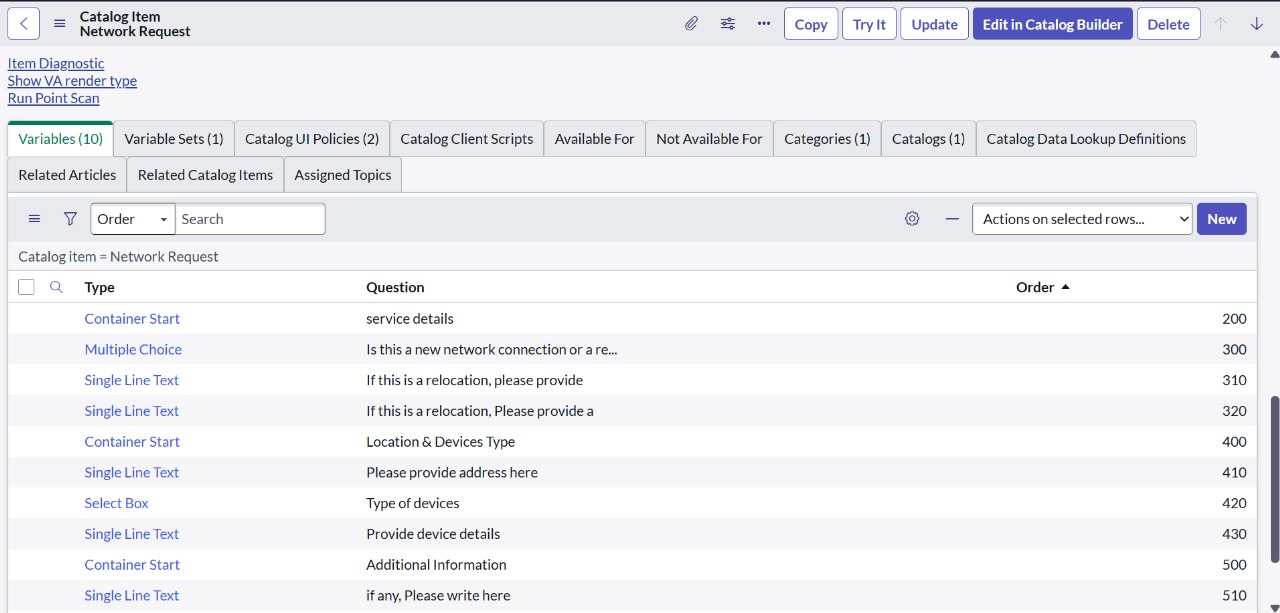




**Variable Configuration:**

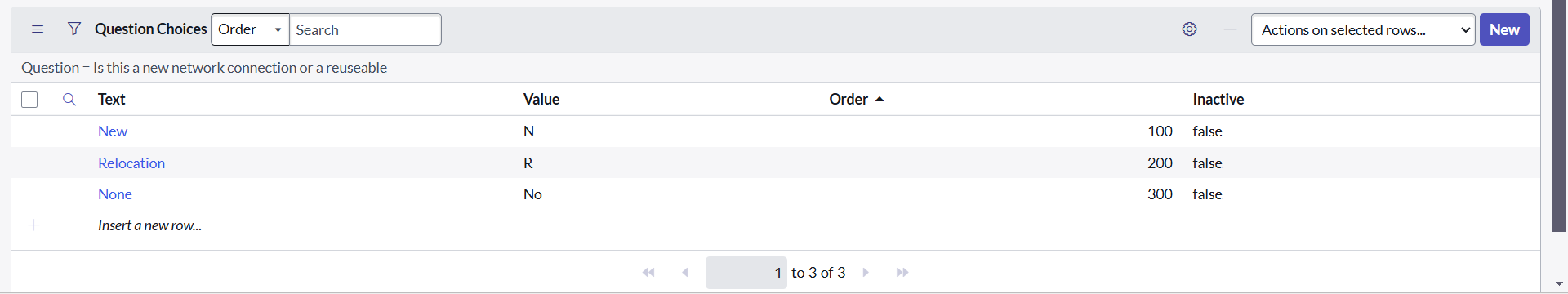
**Go to the Variables Tab:**

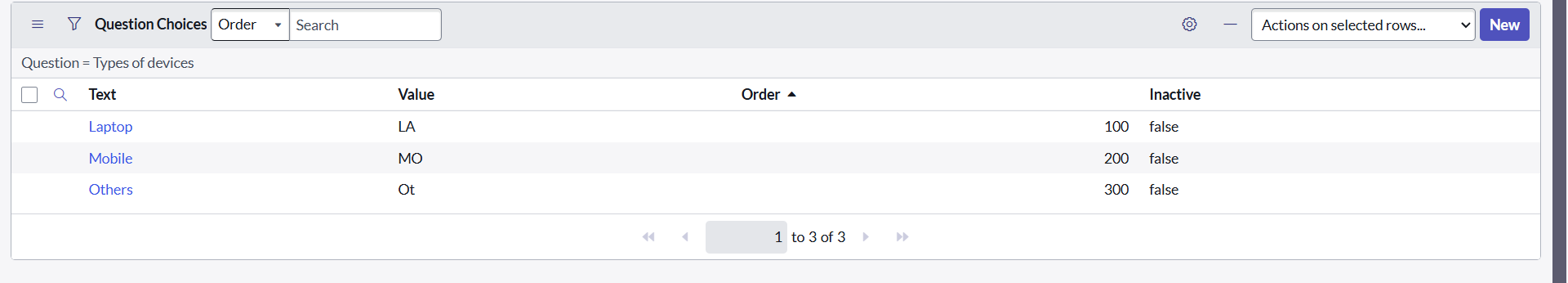
1. Open the catalog item you just created (Service Catalog → Catalog Definitions → Maintain Items).
2. Click the Variables related list at the bottom of the form.
3. Click New to create a variable.



**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





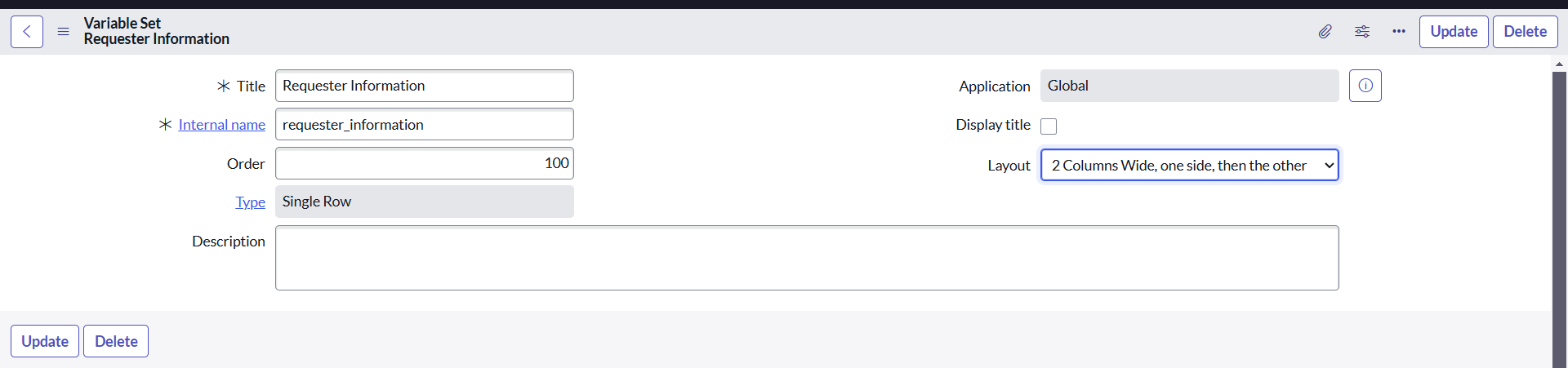
**Variable Set Configuration:**

Variable Sets allow you to reuse common variables across multiple catalog items.

Navigate to Service Catalog → Catalog Variables → Variable Sets.

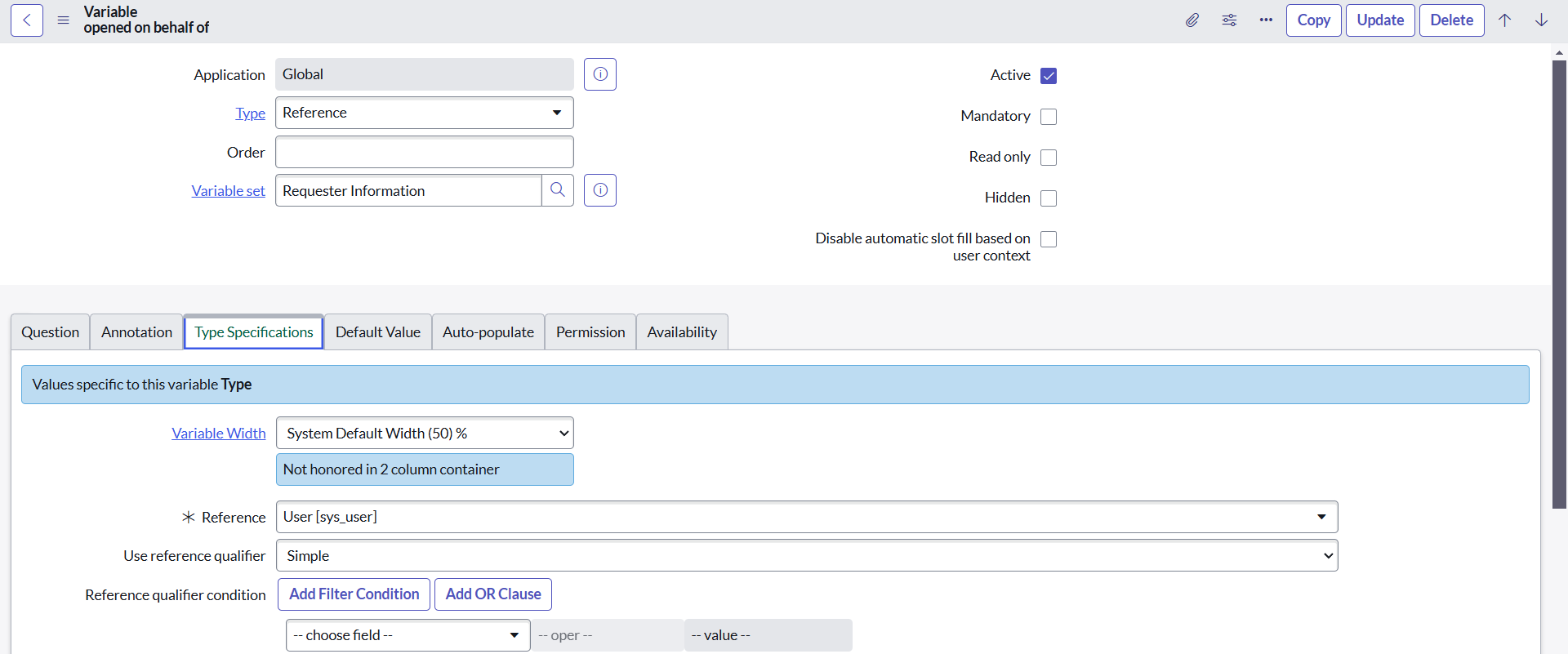
Create a set, add variables to it, then associate the set with multiple items.

Create Title as Requester Information ,Type as Single Row and Layout as 2 Columns Wide,one side then the other

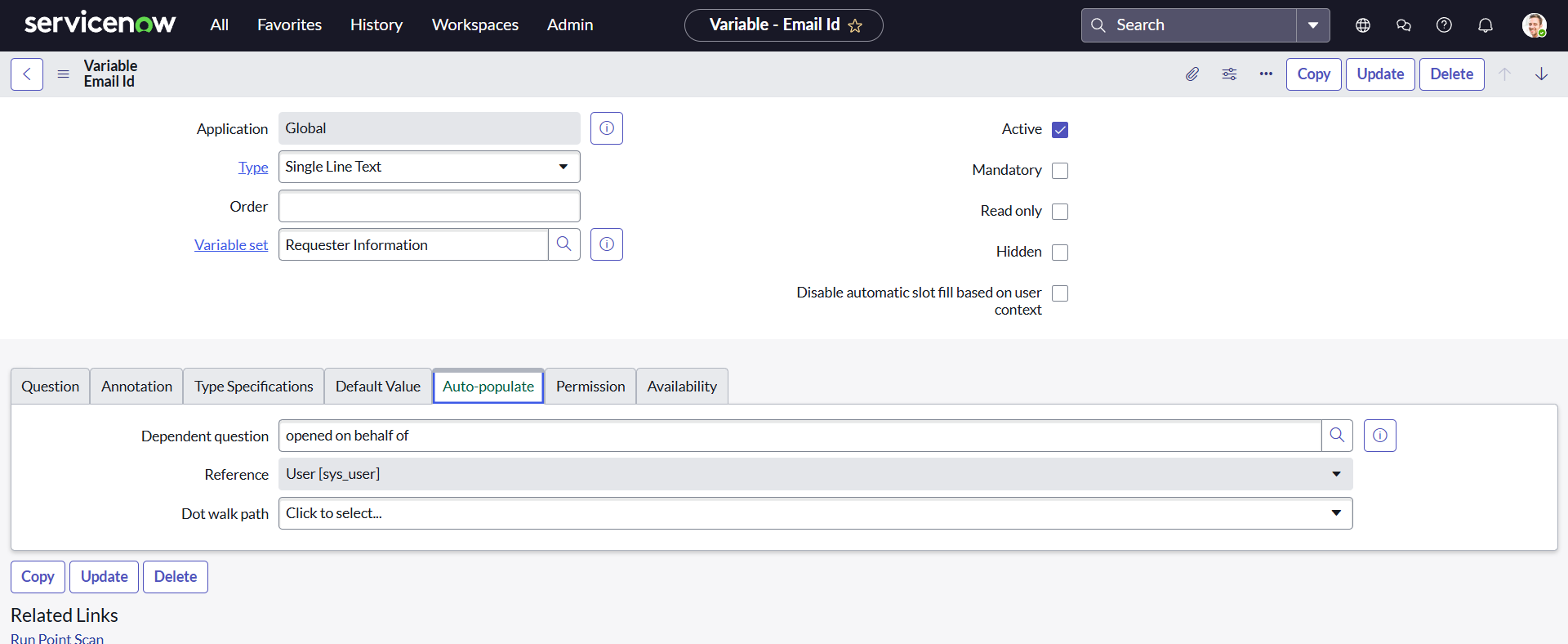


**Variable Types In Variable Set:**

Opened on behalf of >> Reference>> reference to user table



Email Id >> Single line text >> Auto populate by Opened on behalf of variable.

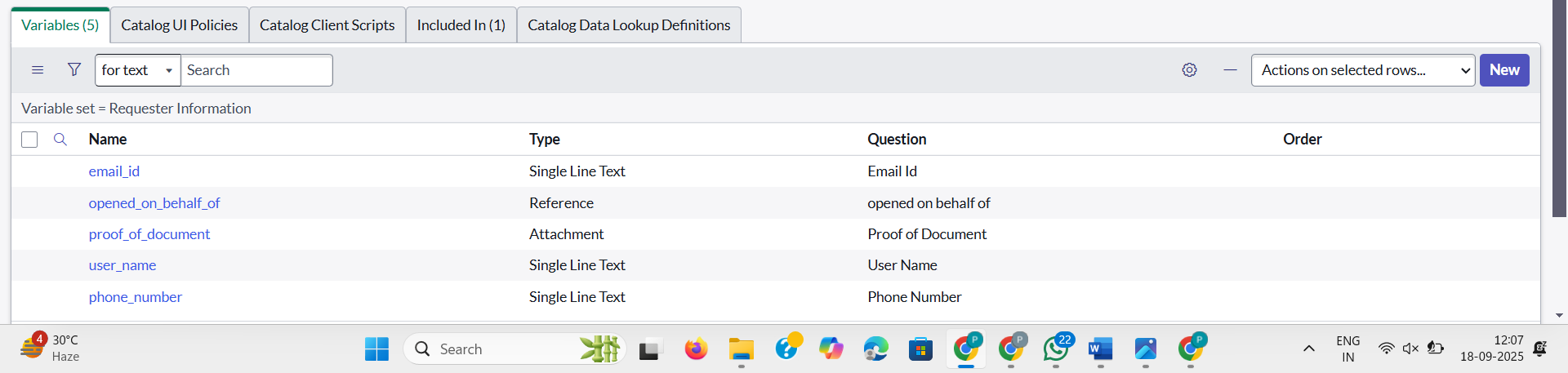


As mentioned in the above figures, all other variables are followed in the same process:

User name >>Single line text >> Auto populate by Opened on behalf of variable.

Phone Number >>Single line text >> Auto populate by Opened on behalf of variable.

Proof of Document >> Attachment

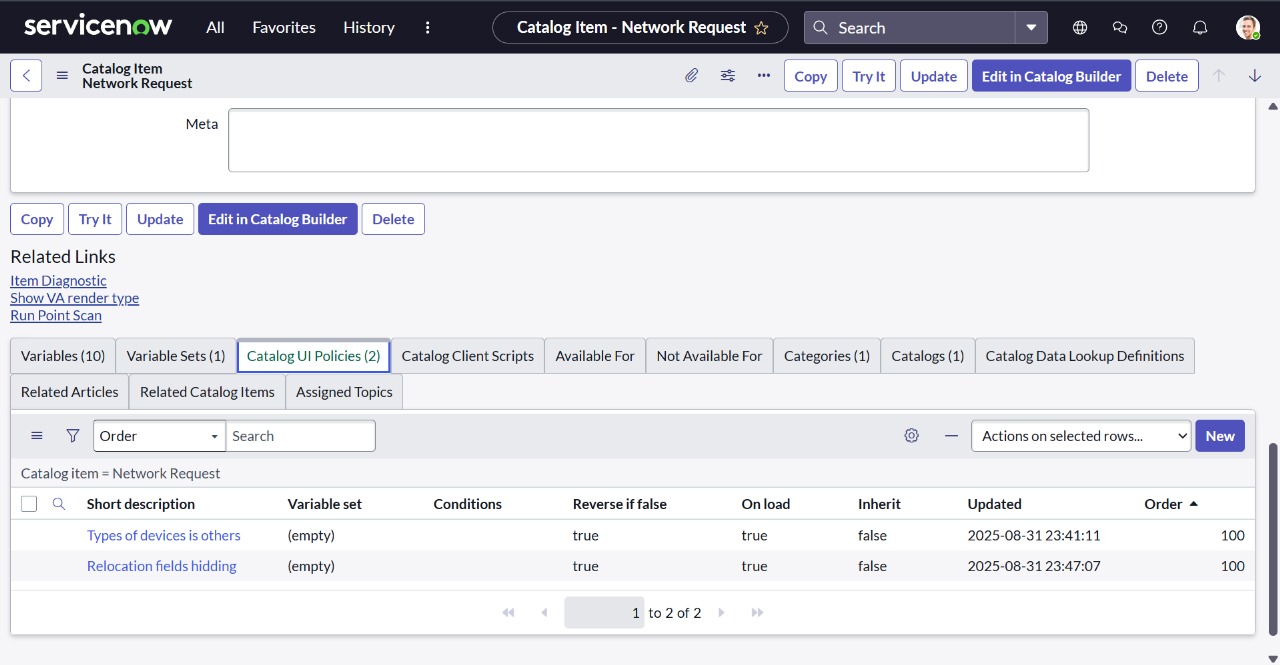


**Catalog UI Policy Configuration**

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**

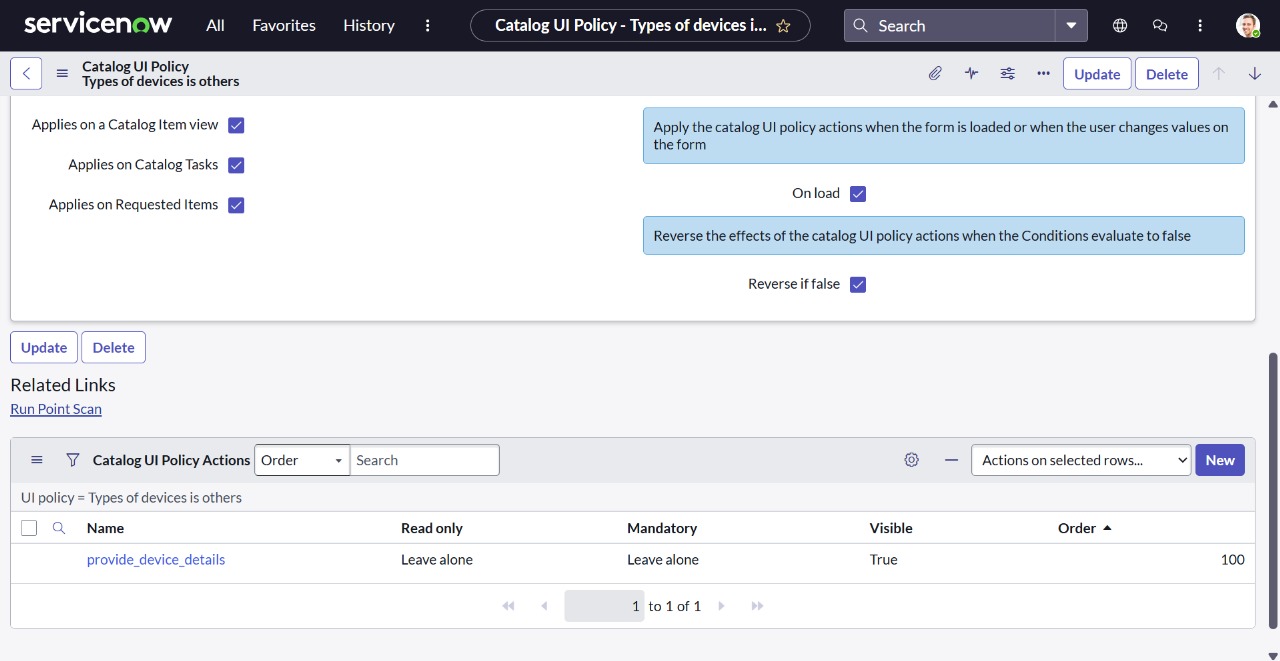


Clickon save, after saving the form will get UI policy actions in the related list

Click on New button to configure new UI Policy action, and Select the variable which we want to display on condition

Make Visible True as per our requirement

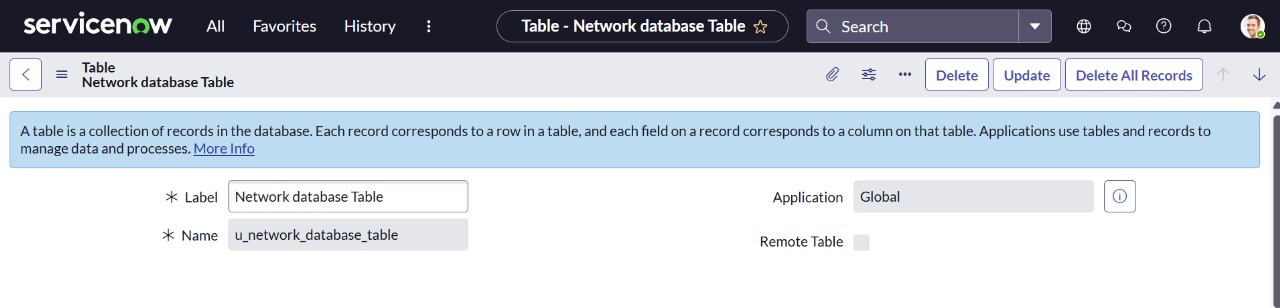
Update the UI Policy and Test the same on Catalog form.



**Creation of Table:**

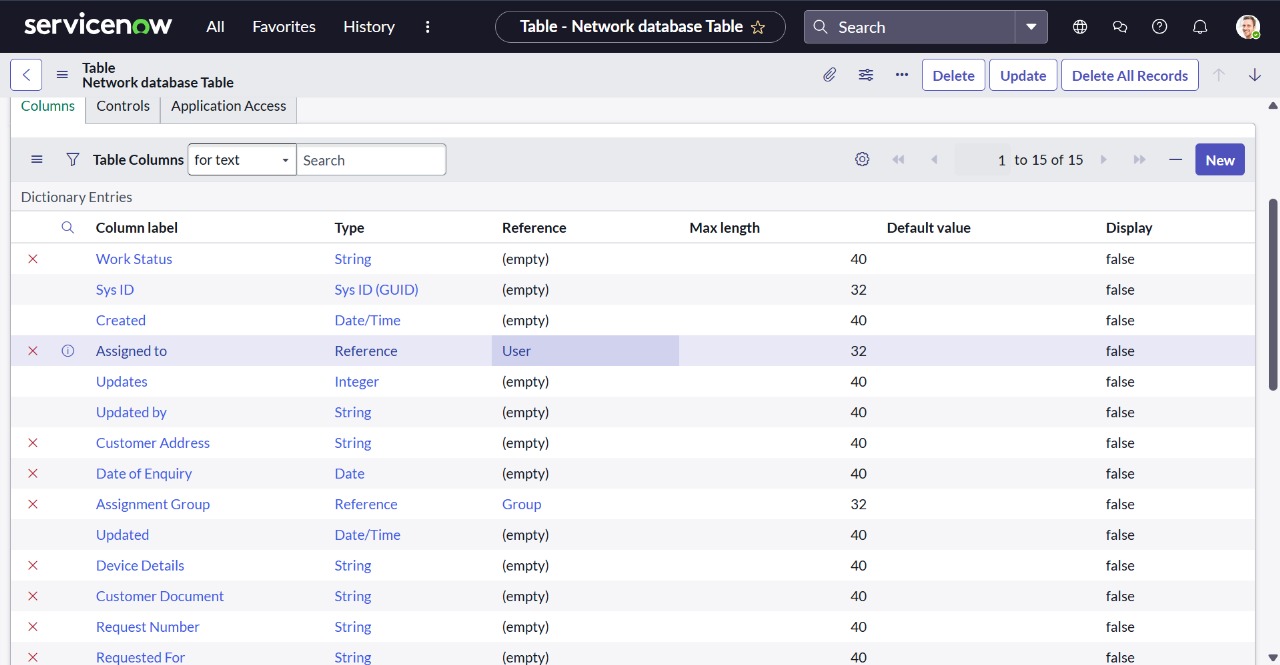
**Create the Network database Table**

* Navigate to **System Definition → Tables**.
* Click **New** to create a new table.
* Fill in **Table Information**:
  + **Name**: **Network database**
  + **Label**: **Network database** (backend name of the table).
  + **Auto-generate schema**: Leave it checked if you want ServiceNow to auto-generate schema fields.
* Click **Submit** to create the table.



**Creation of Fields:**

**We had created the columns as mentioned in the below figure:**



**Navigating to Flow Designer:**

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



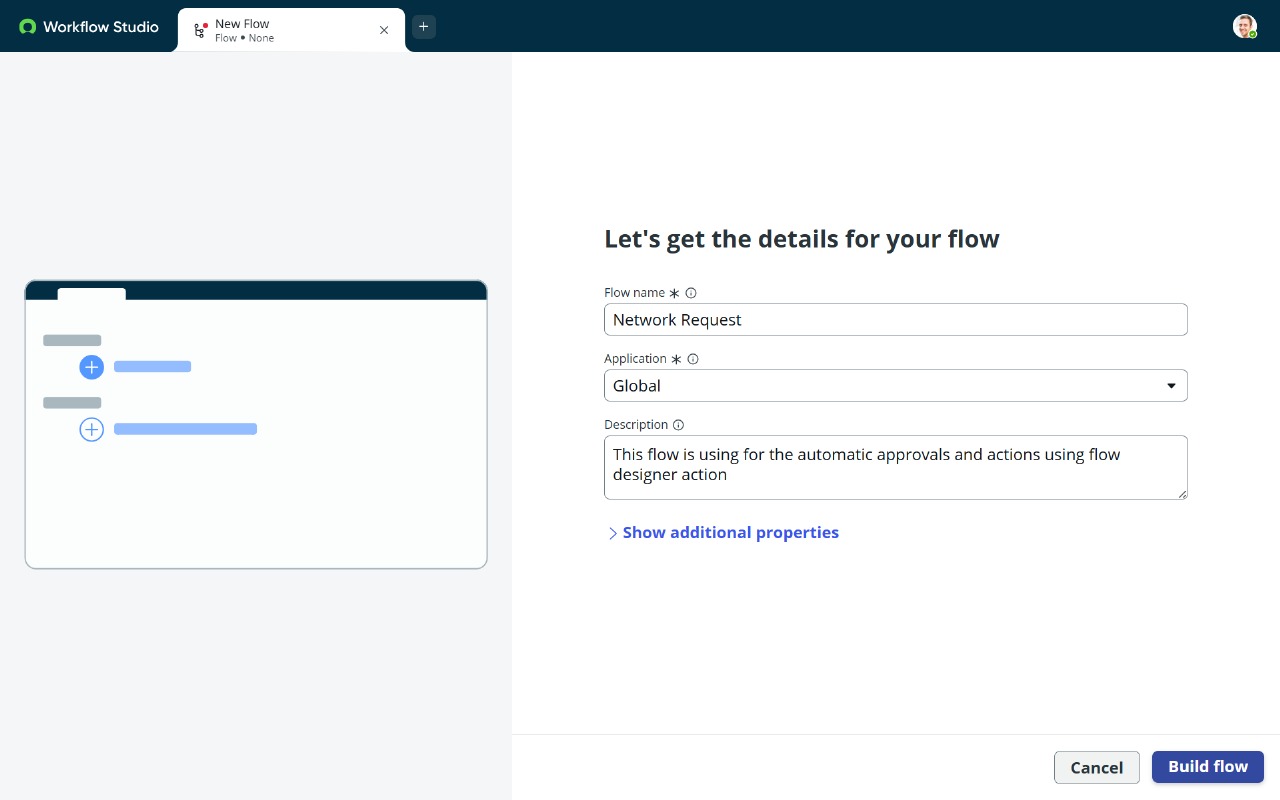
**Creation Of Flow**

After clicking flow designer, it redirects to a new page .

1.Click on New

2.Flow Name >Network Request and give the description

3.Click on Build flow

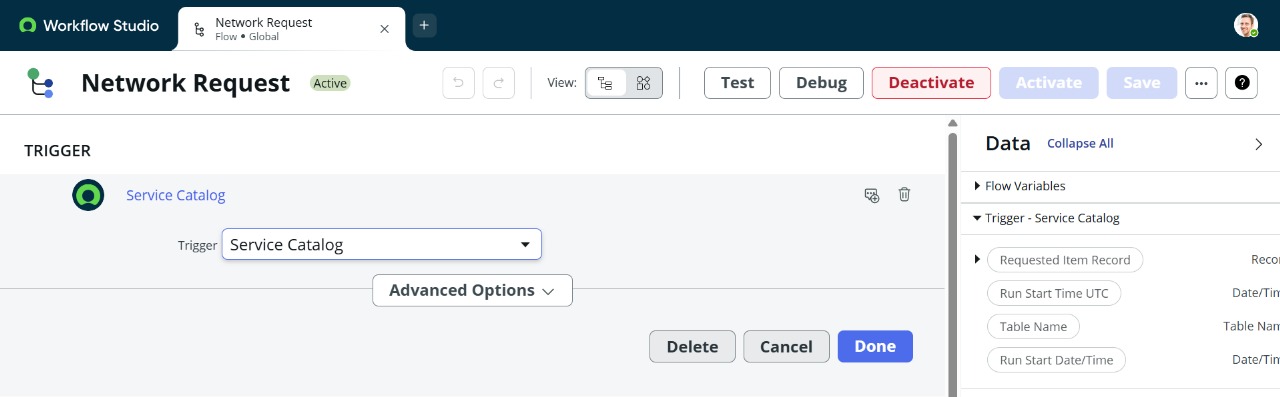


**2.Configuring Trigger**

1.Click on (+) Icon to Configure the Trigger

2.Select Trigger as Application >> Service catalog

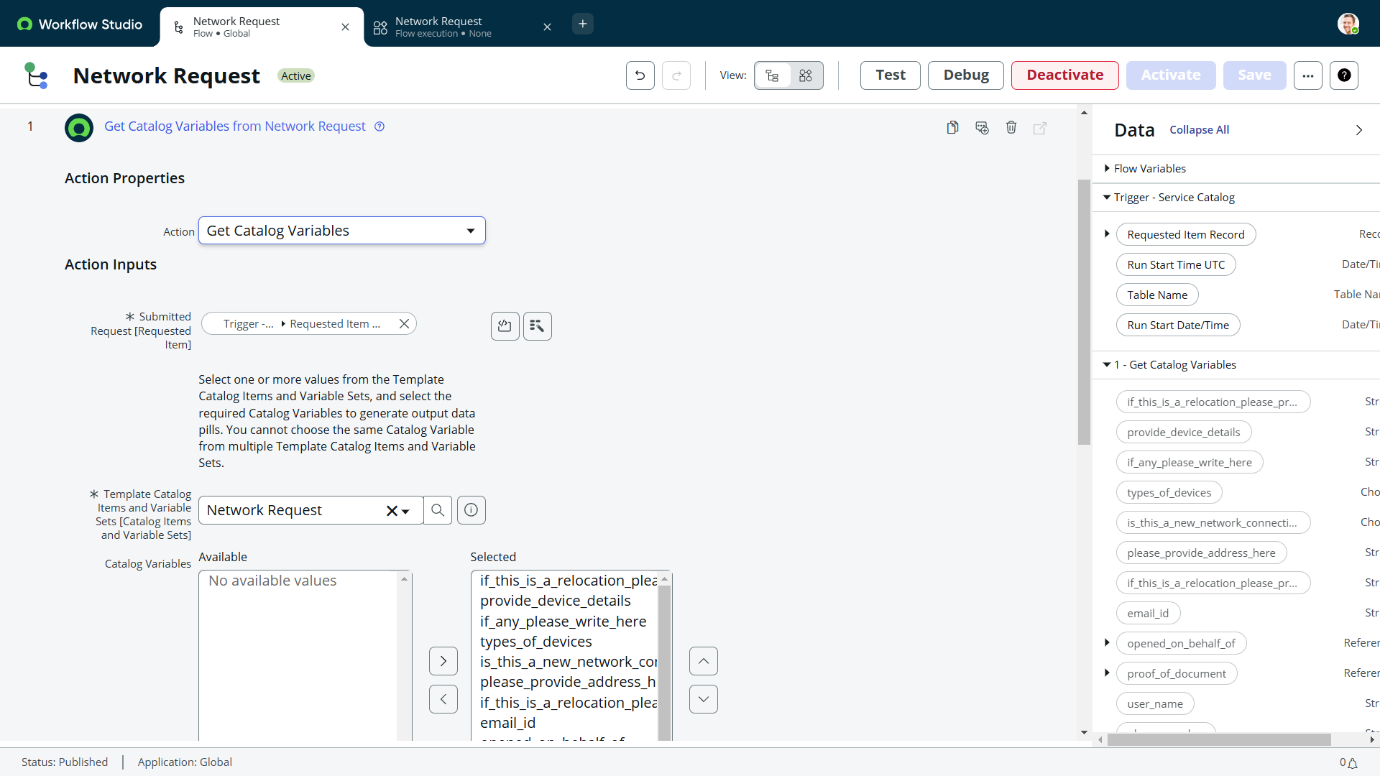
3.Click on Done.



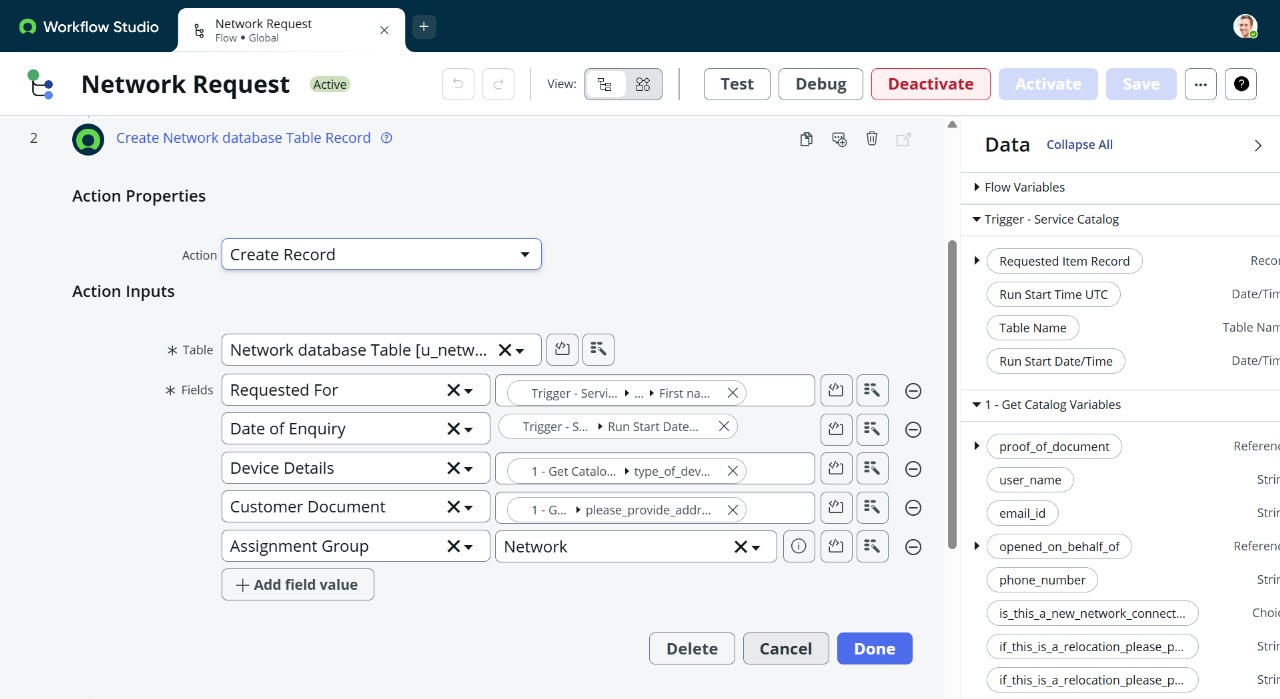
Configuring Action:

We are creating multiple actions to create a Flow.

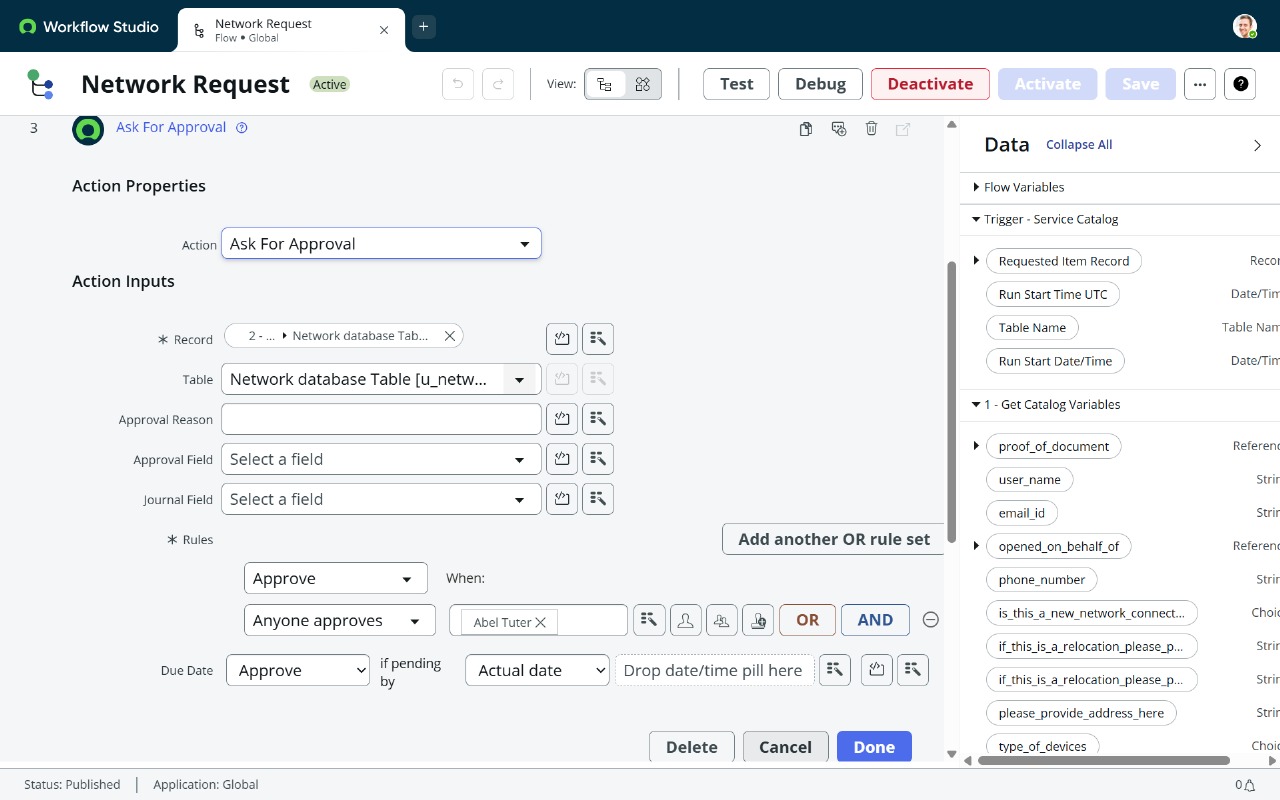
Step 1:We created an Action :Get catalog Variables with several Action inputs.



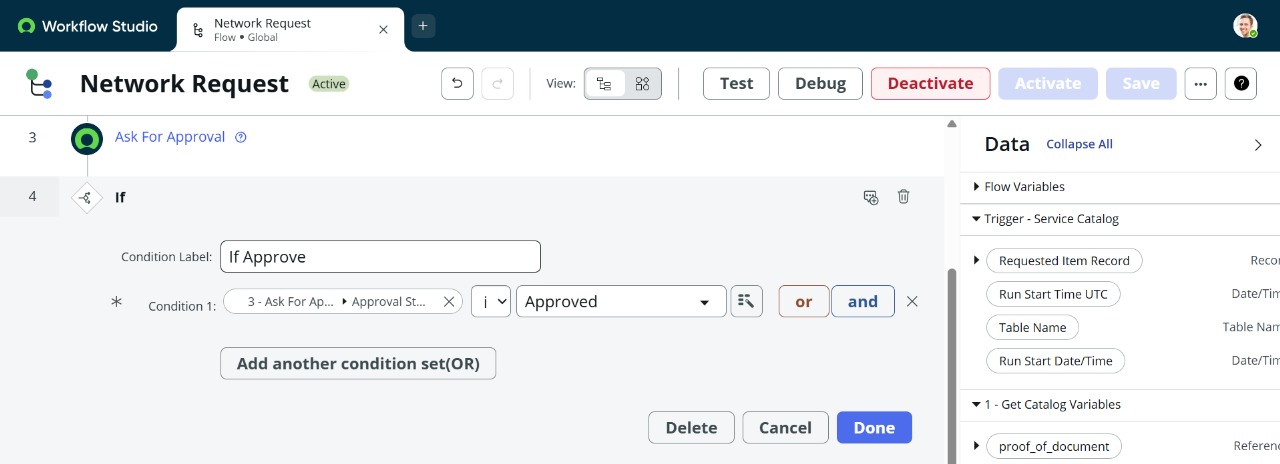
Step 2: We created an Action :Create Record



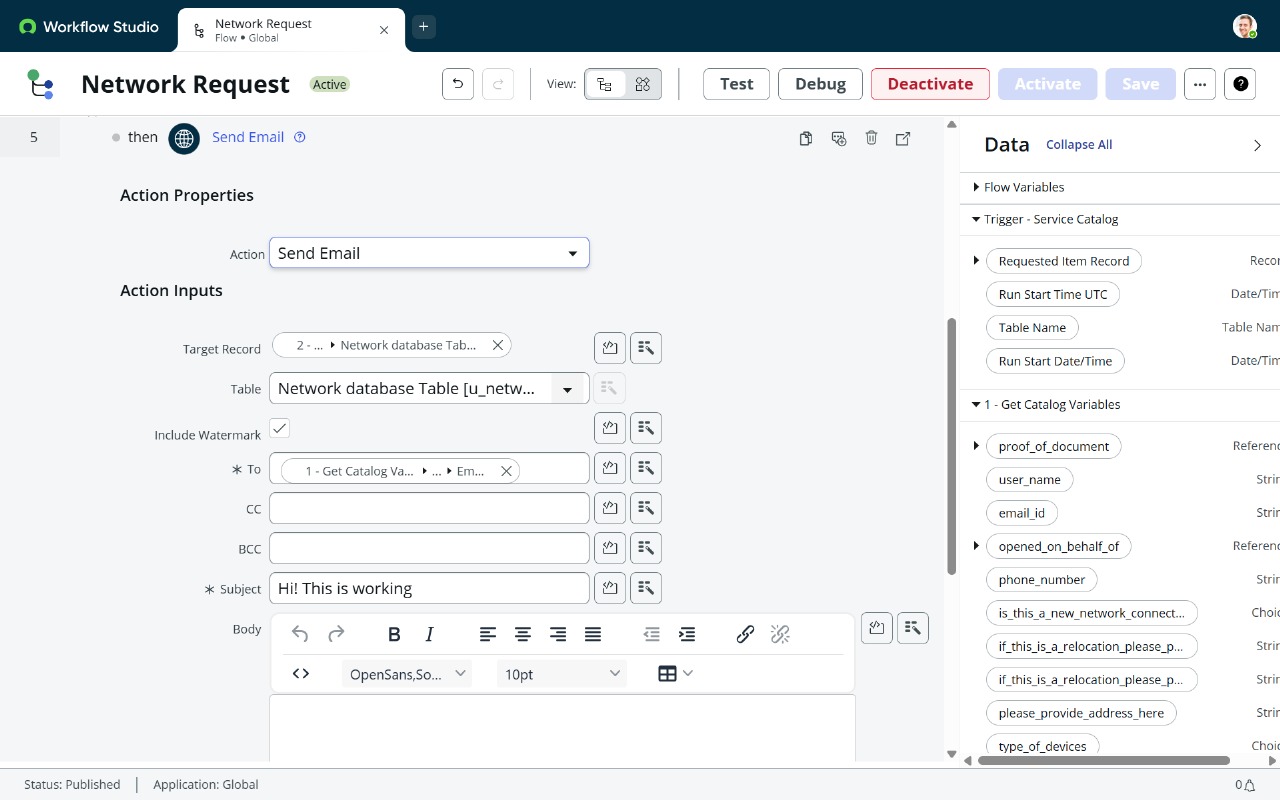
Step 3: We created an Action :Ask For Approval



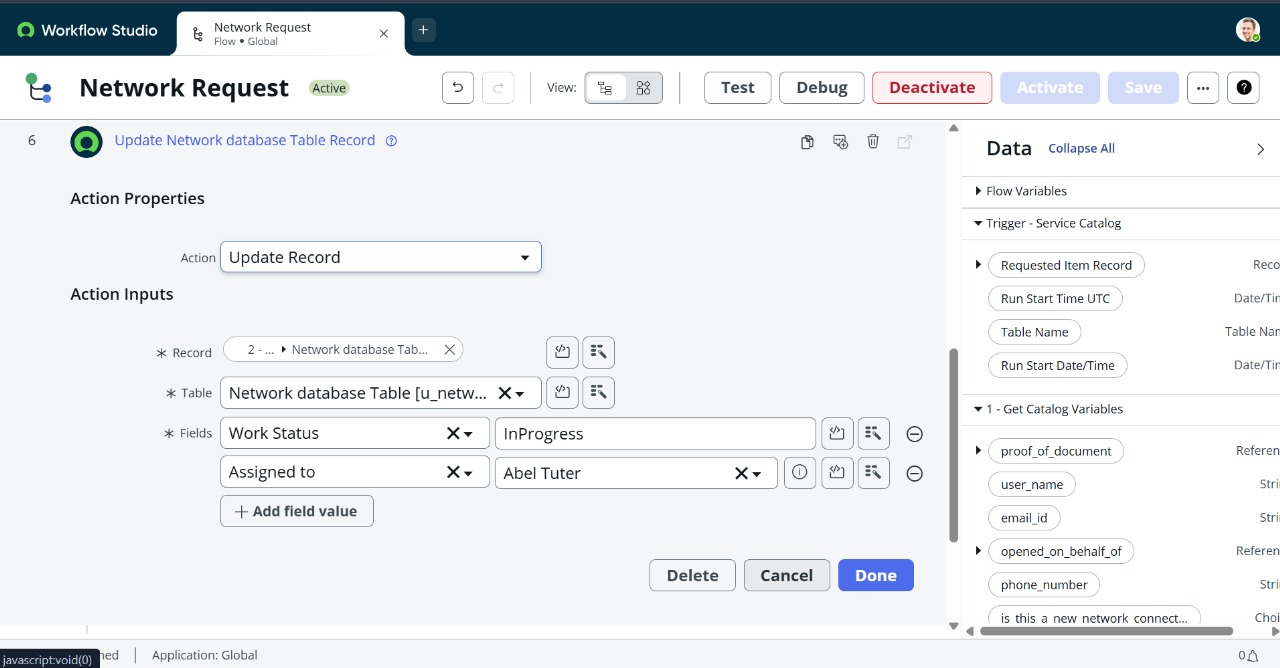
Step 4:We created a Flow Logic

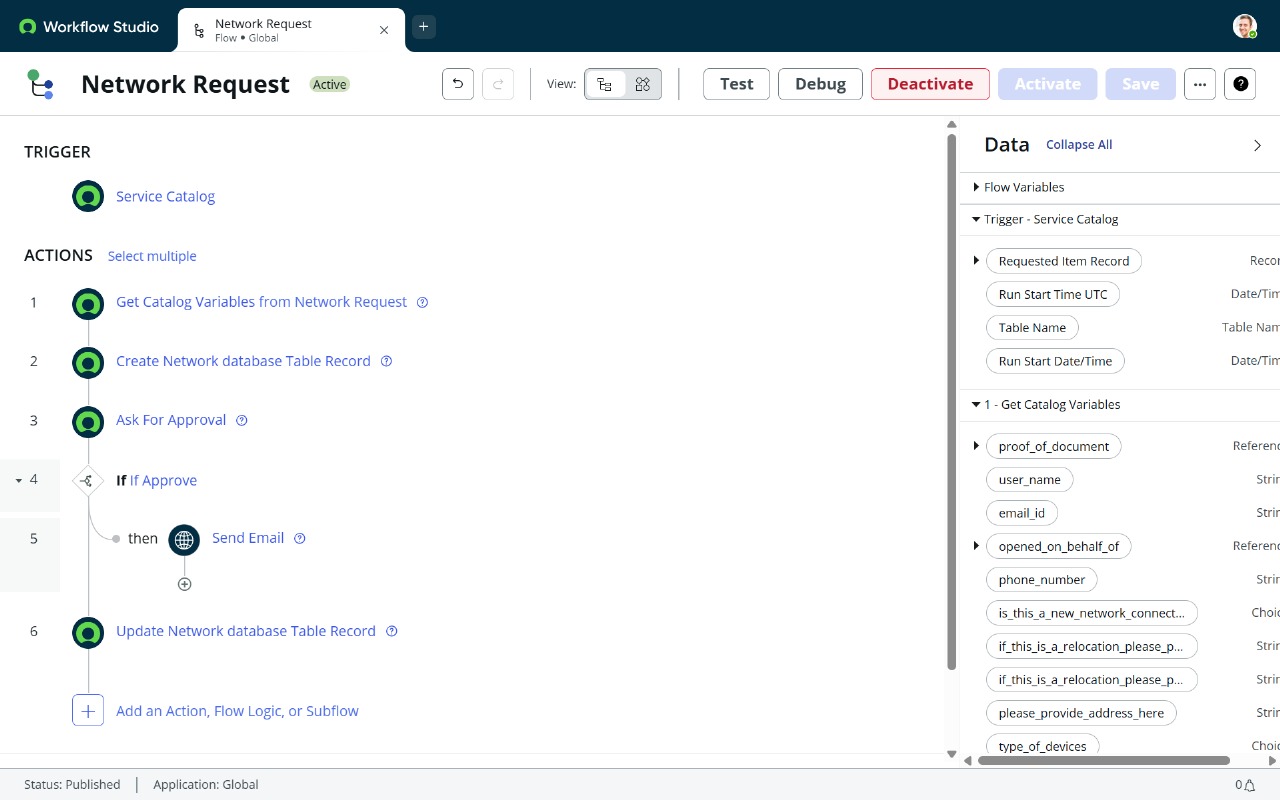


Step 5:We created an action:Send Email

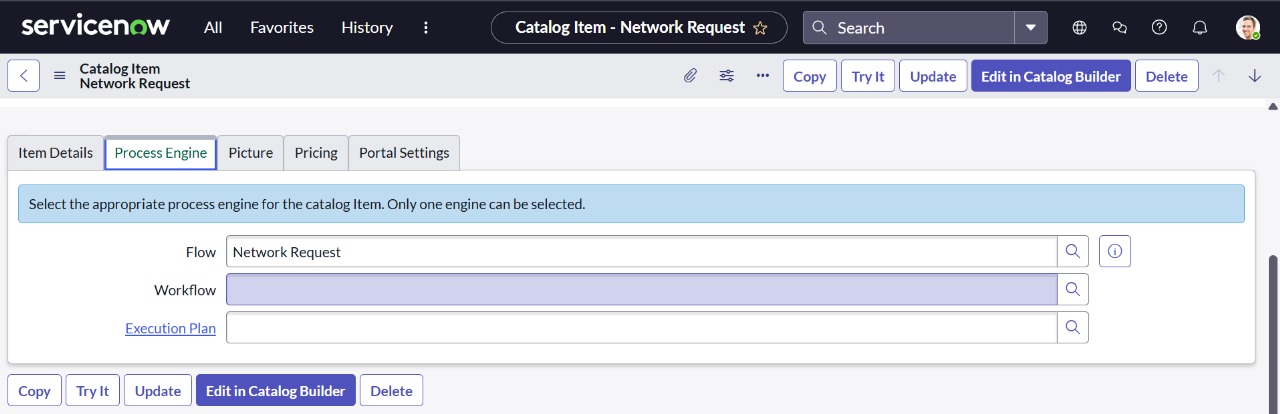


Step 6:we created an action:Update Record

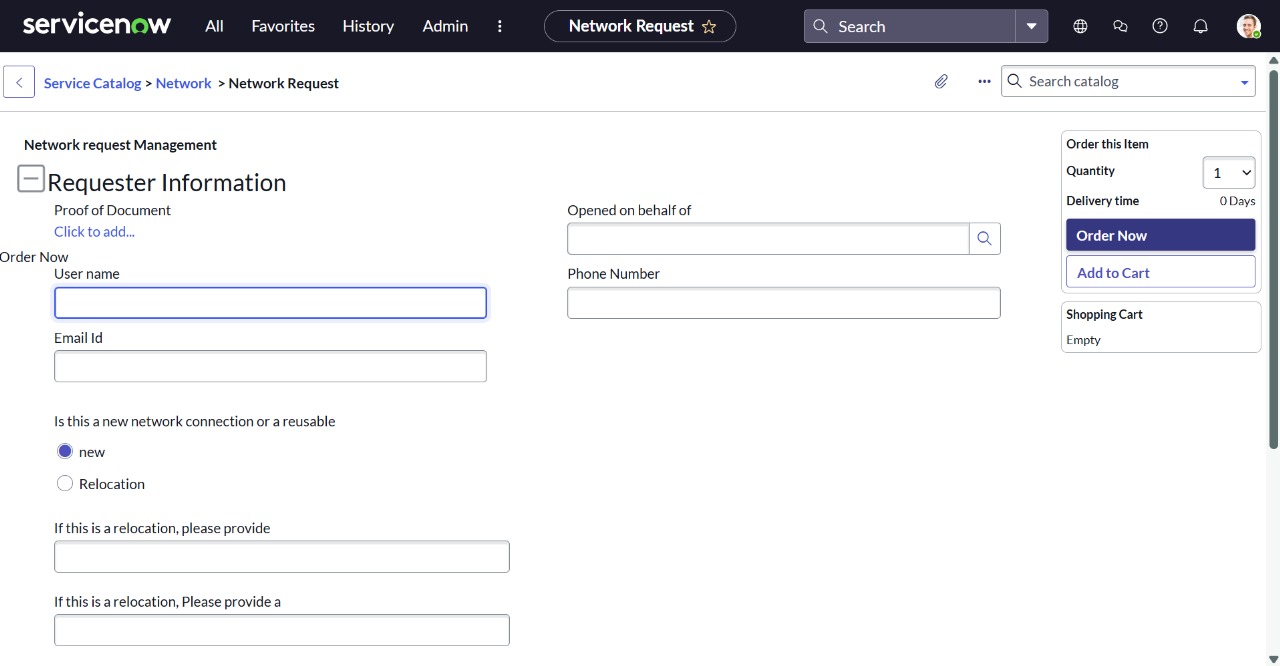


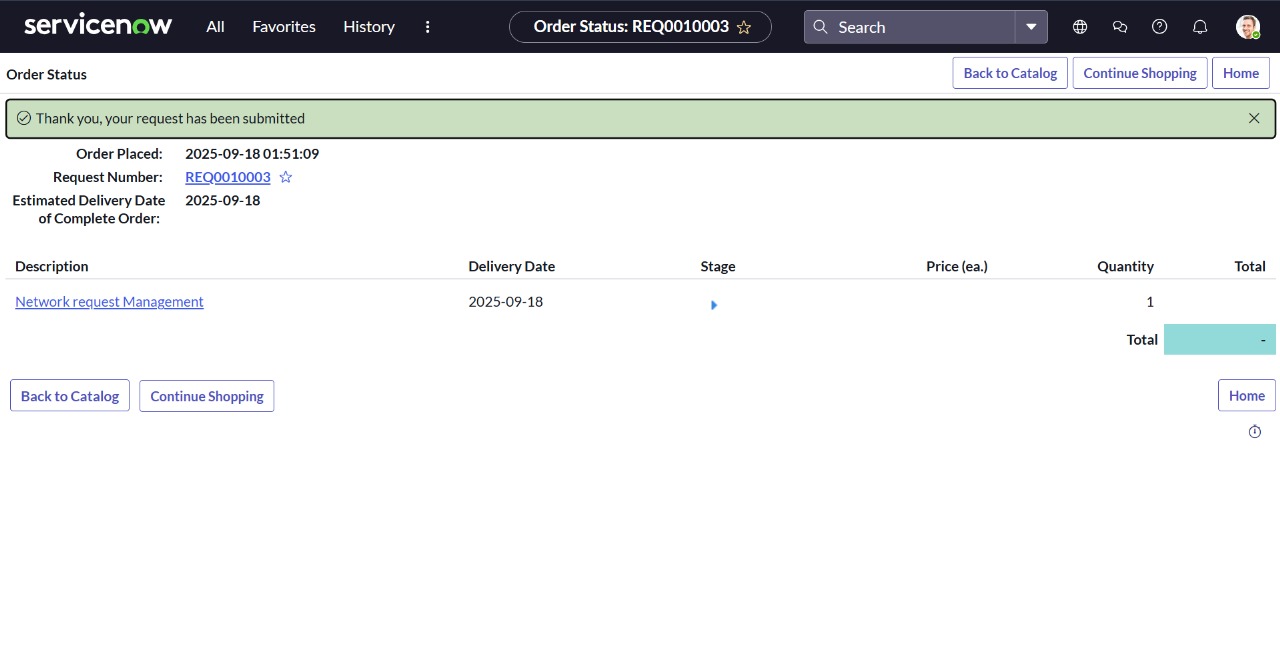
Flow chart:

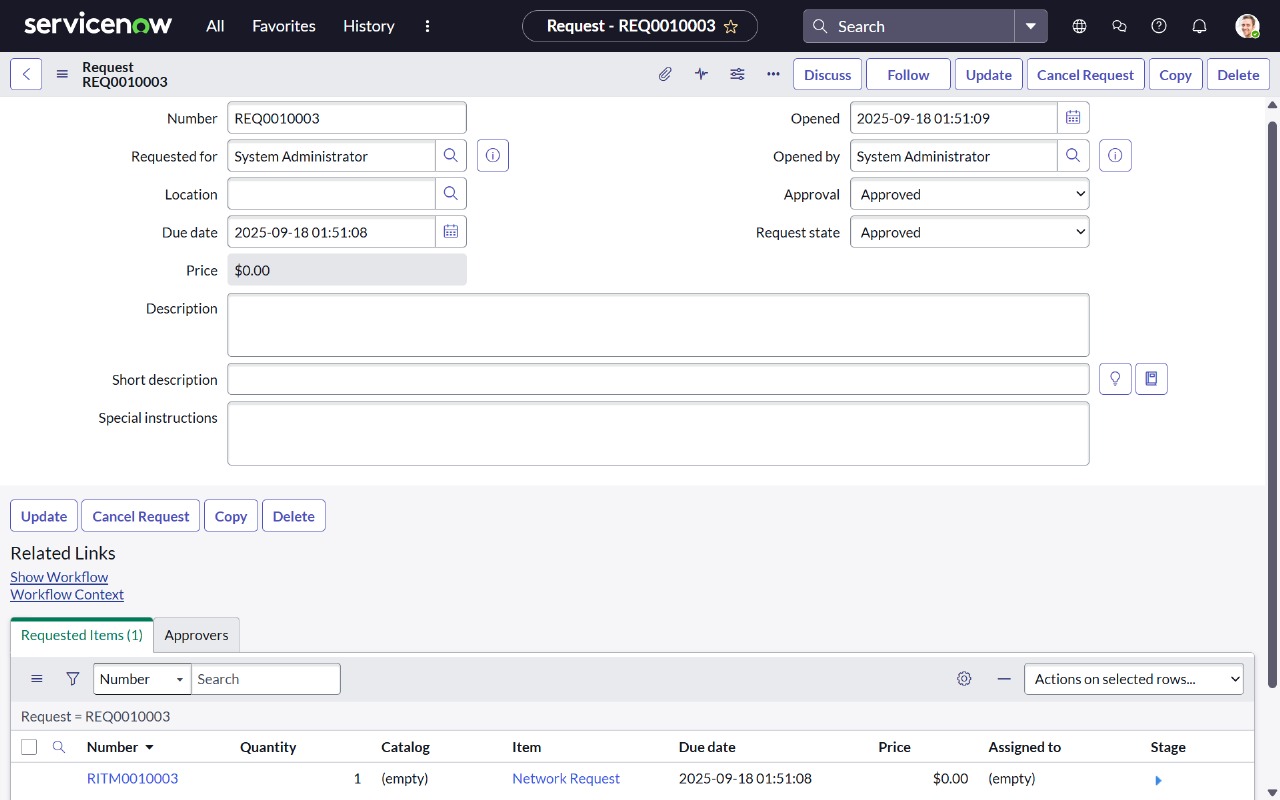
**Final Testing in End User portal & Instance**

In the service catalog->Maintain Items->Network Request->Process Engine->Flow->Network Request

Then click on Try it and choose order now :

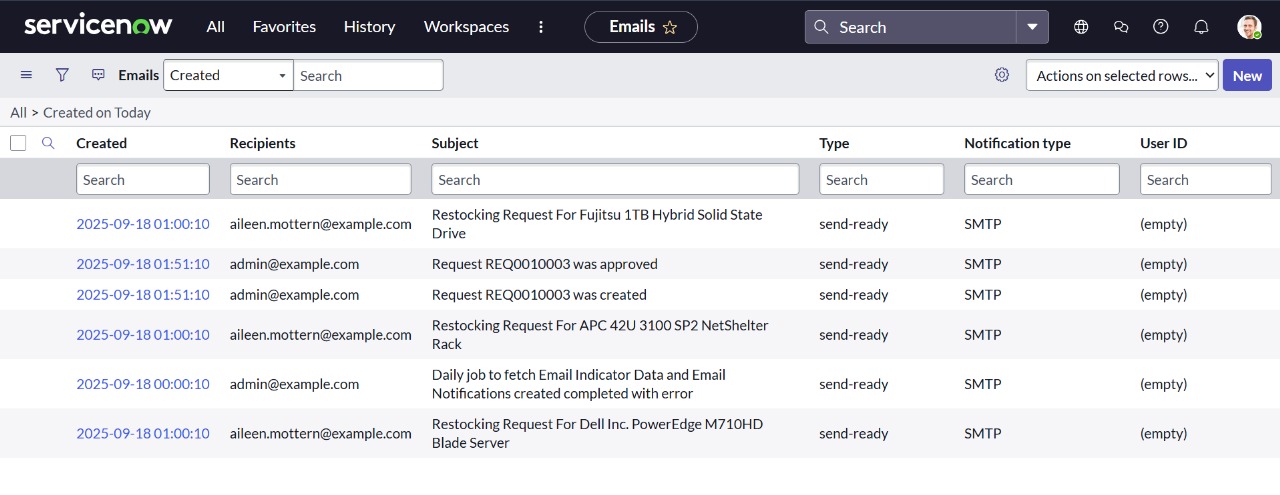
Then Request Number can generated after that Request item number can copied and test into flow .



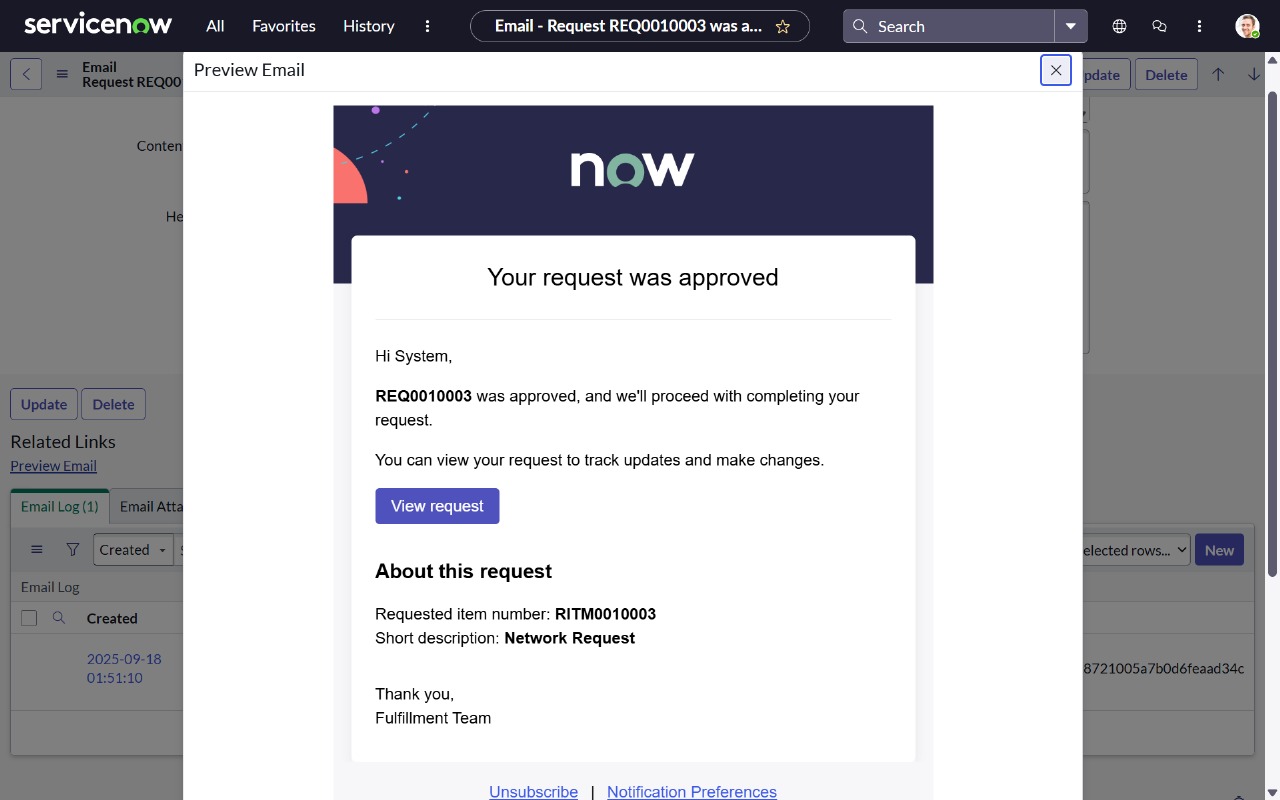


Go to System logs->Email

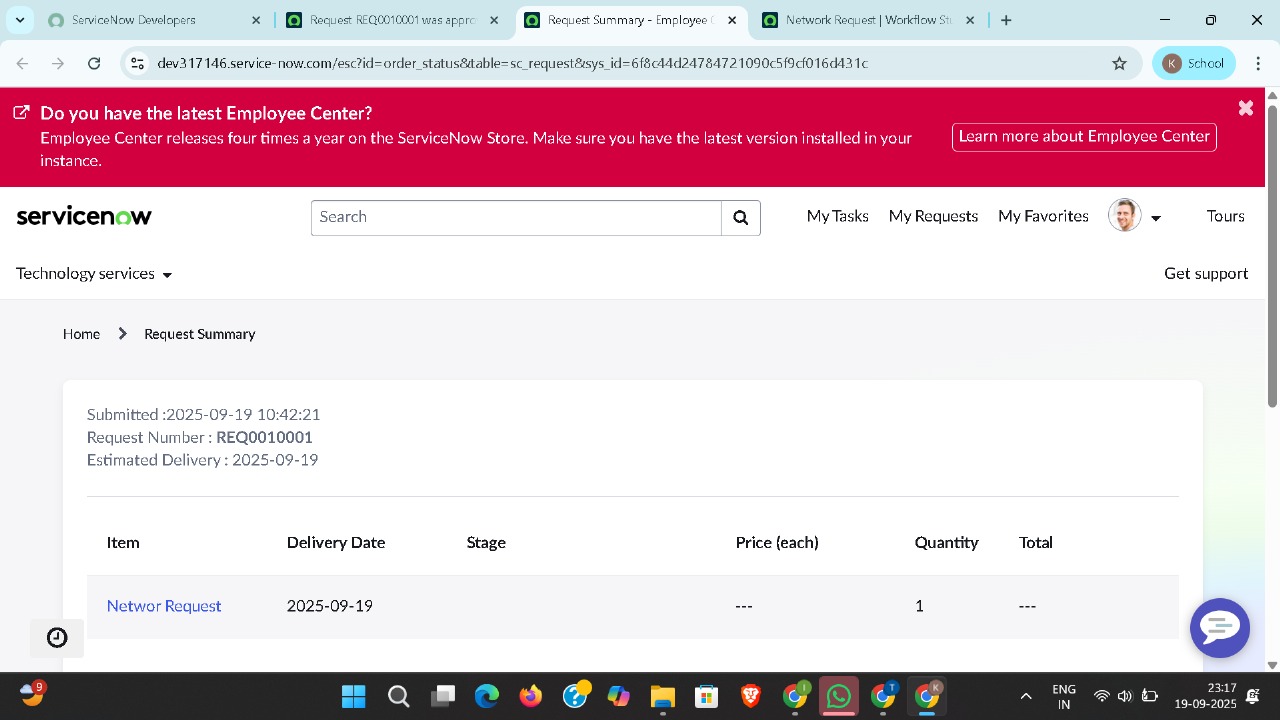
And we can see our request number is created and got acknowledged as request approved.



Now click on the Request approved. And choose preview Email. And we can see the request approval for our Order.



And if we click on view request we can see the summary of the Request.



## Advantages of the Project

* Automation of Repetitive TasksThe project eliminates manual handling of standard network requests by leveraging automated workflows, saving time for both requesters and administrators.
* Improved EfficiencyWorkflows, approvals, and triggers in ServiceNow ensure faster request processing compared to traditional email- or paper-based processes.
* Accuracy in Data CaptureDynamic forms and variables reduce the chances of incomplete or incorrect request submissions, ensuring accurate data collection.
* Enhanced TransparencyEnd-users can track the real-time status of their network requests, eliminating uncertainty in communication.
* ScalabilityThe solution can be easily extended to serve multiple departments or larger organizations with higher request volumes.
* Centralized TrackingAll requests and their statuses are stored in a centralized ServiceNow system, reducing miscommunication and loss of information.
* Integration FriendlyOffers the capability to integrate with external infrastructure orchestration tools, scripts, or monitoring systems to handle standard network tasks automatically.
* SLA Monitoring and AdherenceBuilt-in reporting ensures service-level agreements (SLAs) are monitored, helping to improve accountability and efficiency.
* Reduction in Human ErrorsManual handling errors such as missed requests or incorrect routing are minimized by automation..

## Practical Benefits to Stakeholders

* For End-Users
  + Easy-to-use self-service portal.
  + Real-time visibility on request status.
  + Quicker fulfillment of routine network requirements.
* For Network Teams
  + Reduced workload of repetitive tasks.
  + Clear prioritization and routing for requests based on established rules.
  + Easier collaboration with other IT teams using ServiceNow integrations.
* For Management
  + Insights into request trends, bottlenecks, and resolution times through analytics.
  + Supports decision-making with data-driven reports.
  + Cost reduction via streamlined workflows.

## Use Cases of Automated Network Request Management

* Requesting access to new Wi-Fi or VPN connections.
* Relocation of LAN, telephony, or hardware network points.
* Issuing standard network services for new employees or departments.
* Approving and tracking firewall policy change requests.
* Automating device registration for Laptops, Mobiles, or IoT devices.
* Handling surge in requests during organizational events (e.g., infrastructure migrations or expansions).

## Challenges and Limitations

* Initial Setup ComplexityDesigning workflows, configuring ServiceNow, and integrating tools require technical expertise.
* Dependency on ServiceNow PlatformOrganizations must have licensing and infrastructure to support ServiceNow deployment.
* Customization OverheadToo many customizations may complicate future upgrades of the ServiceNow platform.
* Integration ChallengesConnecting with third-party network automation tools may require REST APIs or custom scripts.
* Change ManagementEmployees may need training to adapt from manual request handling to automated workflows.

## Future Enhancements

* AI-Powered Request CategorizationImplementing ServiceNow Virtual Agent or AI Ops for automatically categorizing and resolving simple network requests.
* Chatbot IntegrationAllowing employees to raise network requests via chat applications like Microsoft Teams or Slack.
* Predictive AnalyticsUsing historical data to forecast network issues or surges in request volumes.
* Mobile AccessibilityEnhancing the user experience by enabling mobile-friendly catalog submission and approvals.
* Self-Healing NetworksIntegration with advanced orchestration tools to automatically resolve common network faults without user intervention.
* Multi-Cloud IntegrationExtending request workflows into hybrid cloud environments for organizations managing distributed infrastructure.

## Conclusion

The project, *Automated Network Request Management in ServiceNow*, successfully demonstrates the capability of leveraging automation to transform traditional IT request management systems. By offering a self-service, transparent, and efficient mechanism for network requests, this solution minimizes delays, improves communication, reduces error rates, and enhances end-user satisfaction.

The integration of ServiceNow’s catalog, workflows, and flow designer ensures scalability and adaptability for organizational needs. With real-time tracking, notifications, analytics, and reporting, the project not only streamlines operations but also improves decision-making for IT administrators and management alike.

As businesses continue to digitize their operations, automation of service requests will become central to IT service management. This project stands as a foundation for future advancements such as AI-driven resolutions, predictive analytics, and fully autonomous network management systems, making it both relevant and forward-looking.

By addressing both immediate efficiency gains and providing a roadmap for future scalability, this project holds academic relevance, practical industry benefits, and scope for continuous improvement.