# 📘 Project Report: Automated Network Request Management in ServiceNow

**Project Title:** Automated Network Request Management in ServiceNow **Submitted by:** Mamidi Amrutha Varshini **GitHub Repository:**[Automated-Network-Request-Management-in-Servicenow](https://github.com/paru19851972/Automated-Network-Request-Management-in-Servicenow.git)

**Screenshots Folder:** [View Screenshots](https://github.com/paru19851972/Automated-Network-Request-Management-in-Servicenow/tree/main/Screenshots)

## 

## Abstract

## This project showcases the automation of network request handling within ServiceNow by leveraging Service Catalogs, Variable Sets, Flow Designer, Approvals, and Notifications. The solution enables end-users to submit network-related requests through an intuitive catalog item, where all required details are captured via well-structured variable sets.

## Once submitted, the request is automatically processed using Flow Designer workflows, ensuring seamless routing, approval management, and task assignment without manual intervention. Automated approval workflows reduce delays, while real-time notifications keep stakeholders informed at each stage of the process.

## By replacing traditional manual request handling, this project improves operational efficiency, accuracy, and response time, while also enhancing the overall user experience within the ServiceNow platform.

## 

## 

## 

## Introduction

ServiceNow is a cloud-based ITSM platform that streamlines IT processes. Network requests like VPN access, firewall updates, or port changes are common and often delayed when handled manually. This project automates the entire lifecycle of such requests, ensuring faster resolution and better tracking.

This project addresses the issue by:

1. Building a Service Catalog item for network requests.
2. Automating request handling through Flow Designer.
3. Adding supervisor approvals for control.
4. Notifying stakeholders via automated emails.

Overall, this solution minimizes manual effort and enhances service delivery in IT operations.

## Objectives

1. **Automation** – Replace manual request processing with fully automated workflows.
2. **Standardization** – Ensure every request follows a structured and auditable path.
3. **Efficiency** – Reduce delays caused by manual approvals and notifications.
4. **User Experience** – Provide a simple and professional request submission form.
5. **Transparency** – Keep all stakeholders informed of request status in real time.

## 

## 

## Literature Review

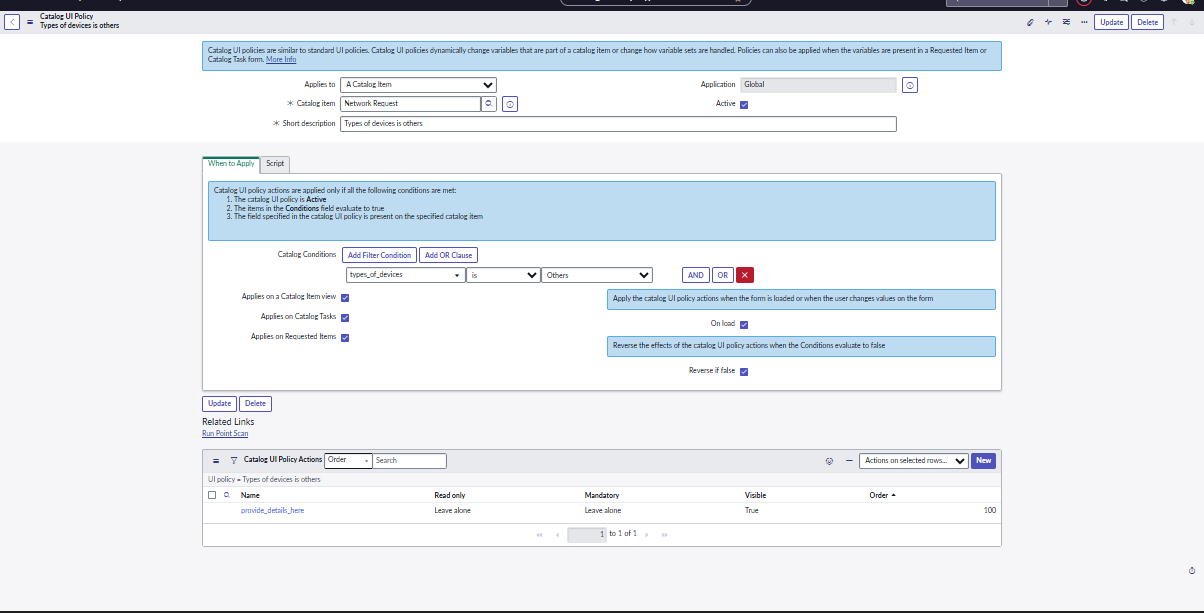
1. **Service Catalog** – A digital catalog containing services available for request. Each item can have fields (variables) to capture user input.
2. **Variable Set** – A reusable group of variables that can be applied to multiple catalog items.
3. **Flow Designer** – A no-code automation tool in ServiceNow for creating process flows.
4. **Approvals** – Used to enforce authorization before fulfilling requests.
5. **Notifications** – Automated emails or messages sent at key stages of the workflow.

This project combines all of these ServiceNow concepts into a single automated solution.

## Methodology

The project follows a **step-by-step approach**:

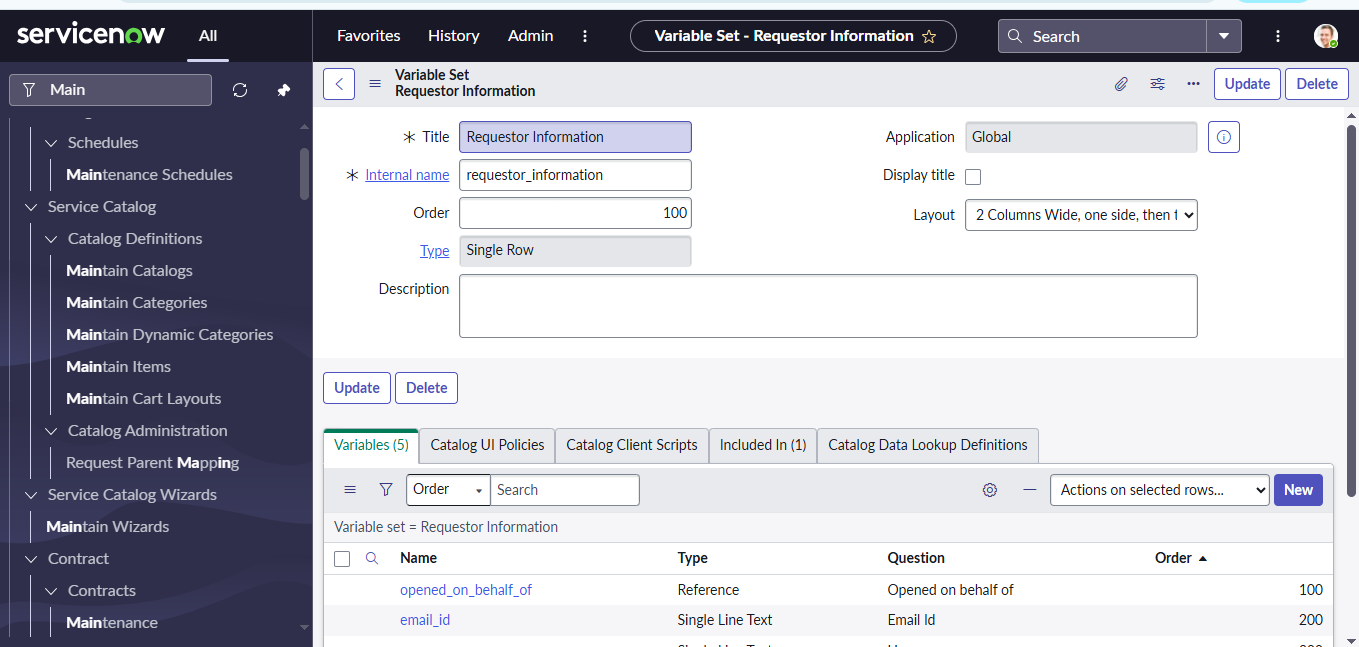
### Step 1: Service Catalog Item Creation

A catalog item titled **Network Request** was created to allow users to submit network-related requests. 

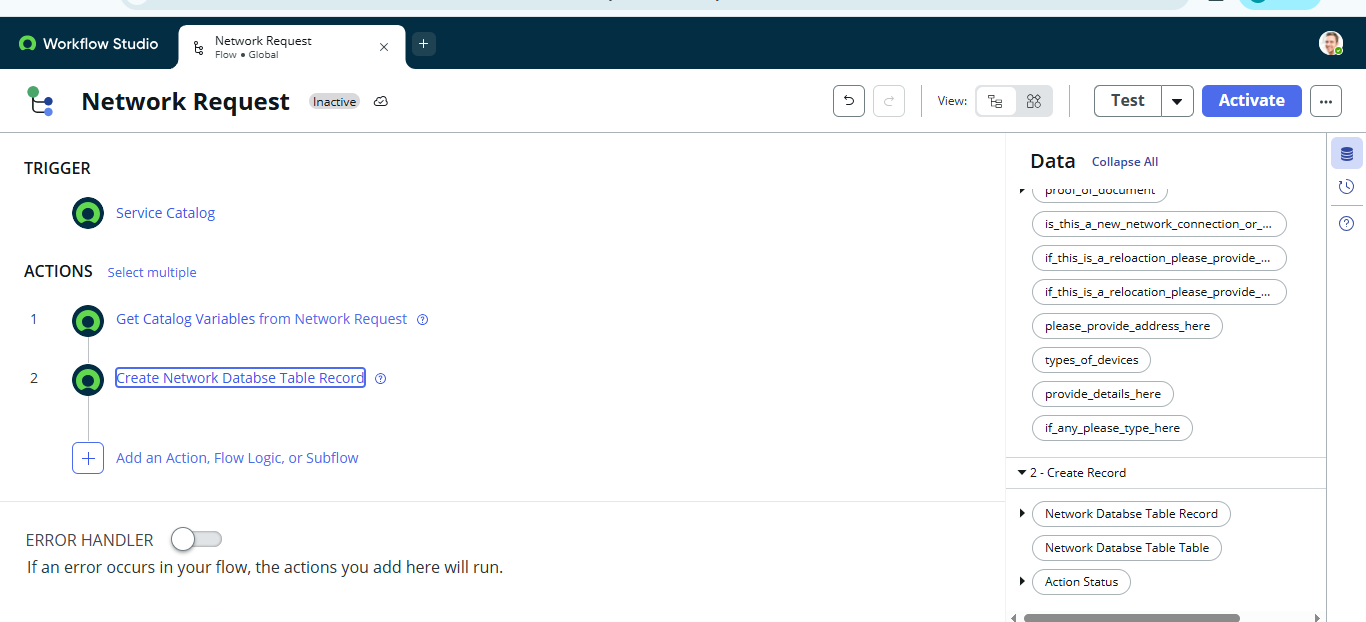
### Step 2: Defining Variables and Variable Sets

A **Variable Set** was configured to capture inputs like:

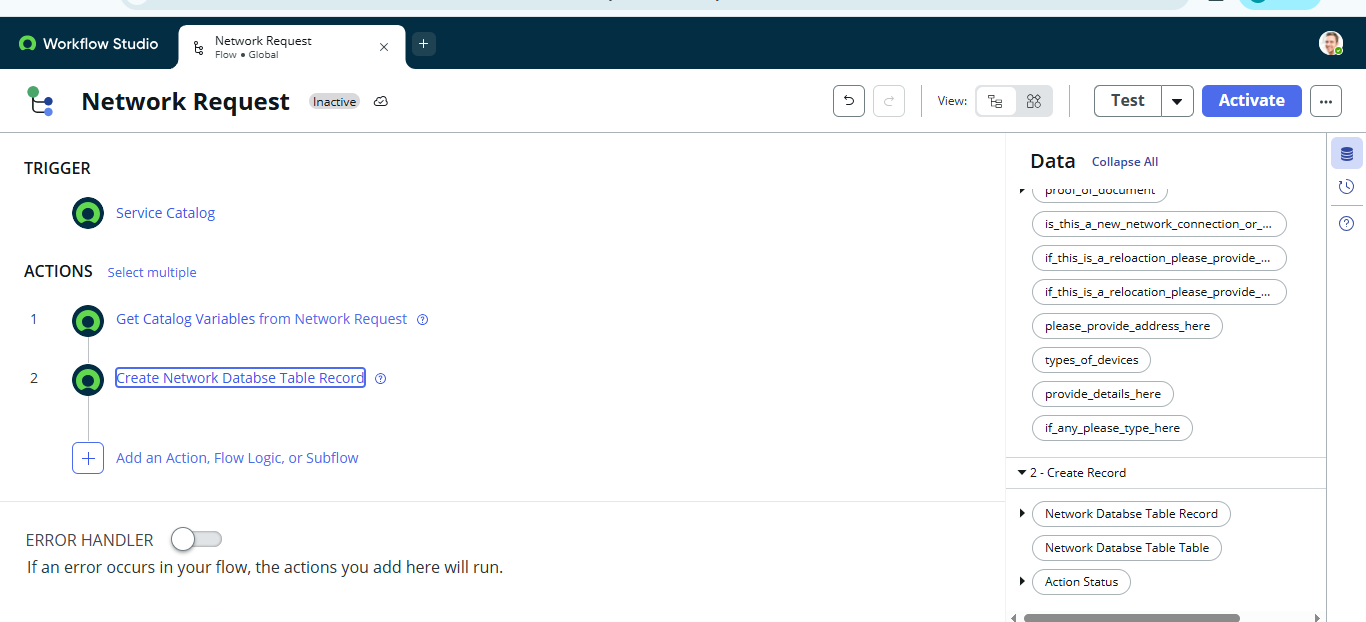
1. Request Type (VPN, Firewall, Port Access, etc.)
2. Requester Details (Name, Department, Email)
3. Justification



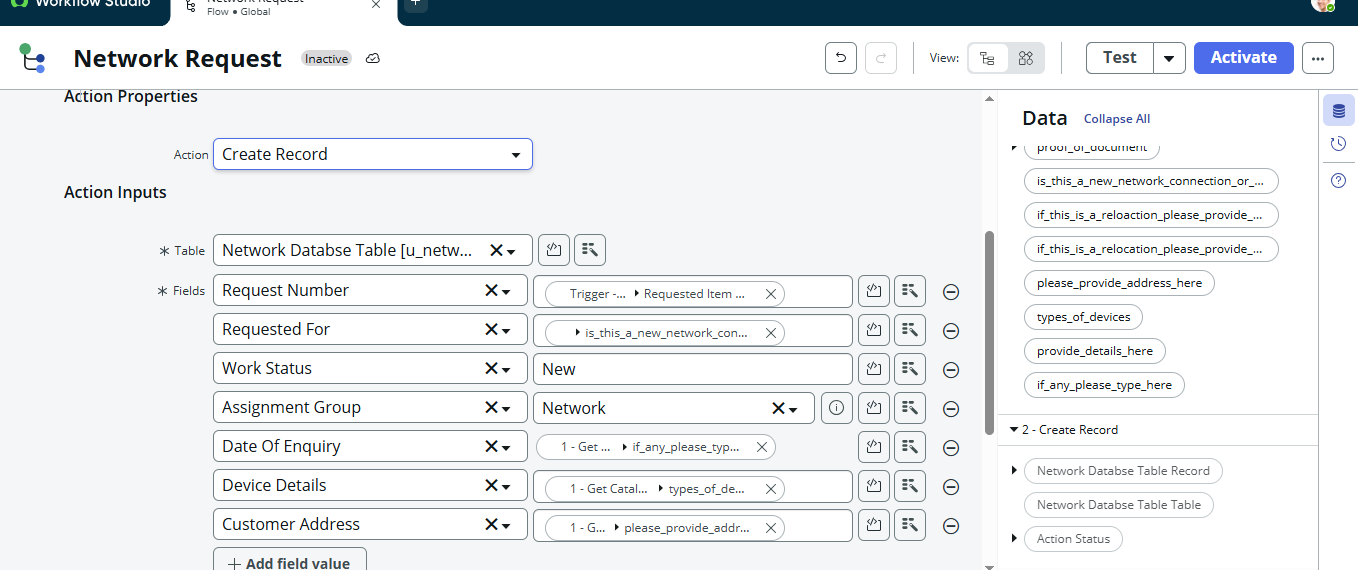
### Step 3: Flow Designer – Initial Setup

A new flow was created in Flow Designer triggered when the catalog item is submitted. 

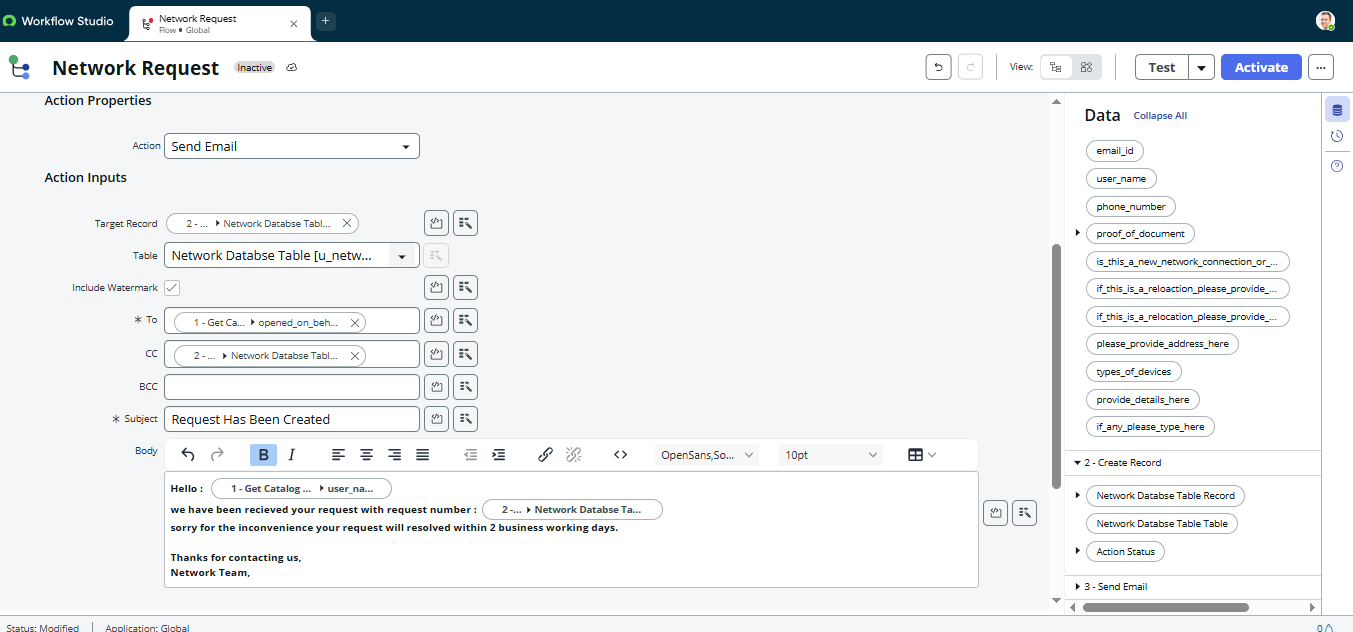
### Step 4: Get Catalog Variables Action

This step fetches the submitted variables for further processing. 

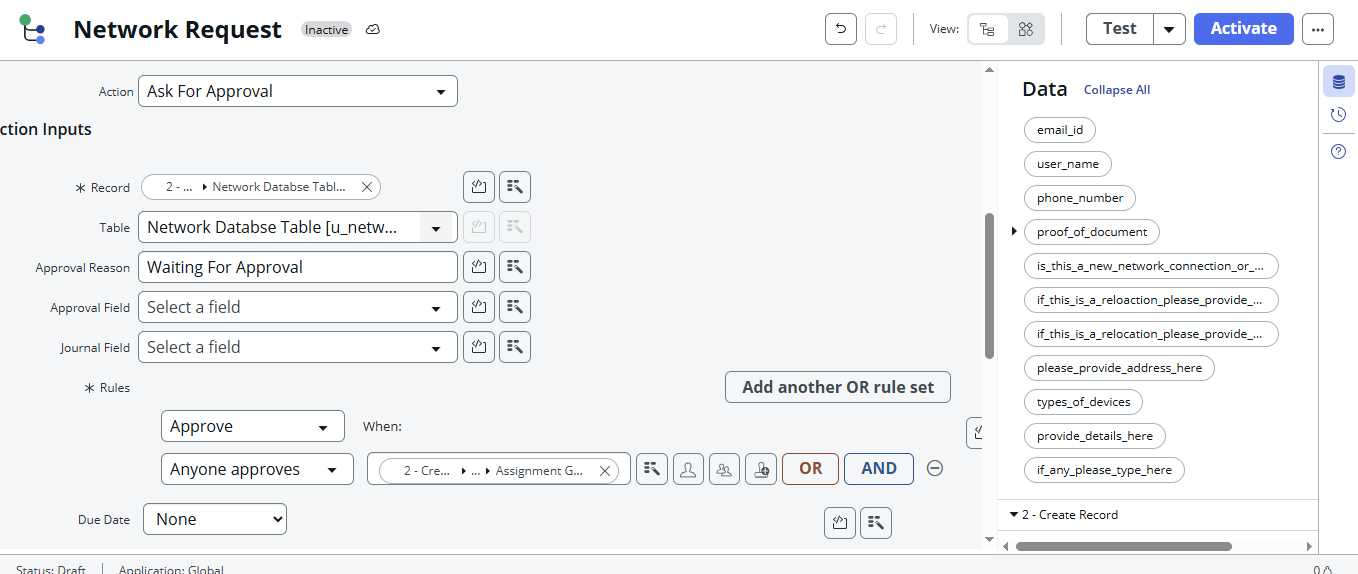
### Step 5: Create Record Action

Automatically creates a record in the database for tracking. 

### Step 6: Send Email Action

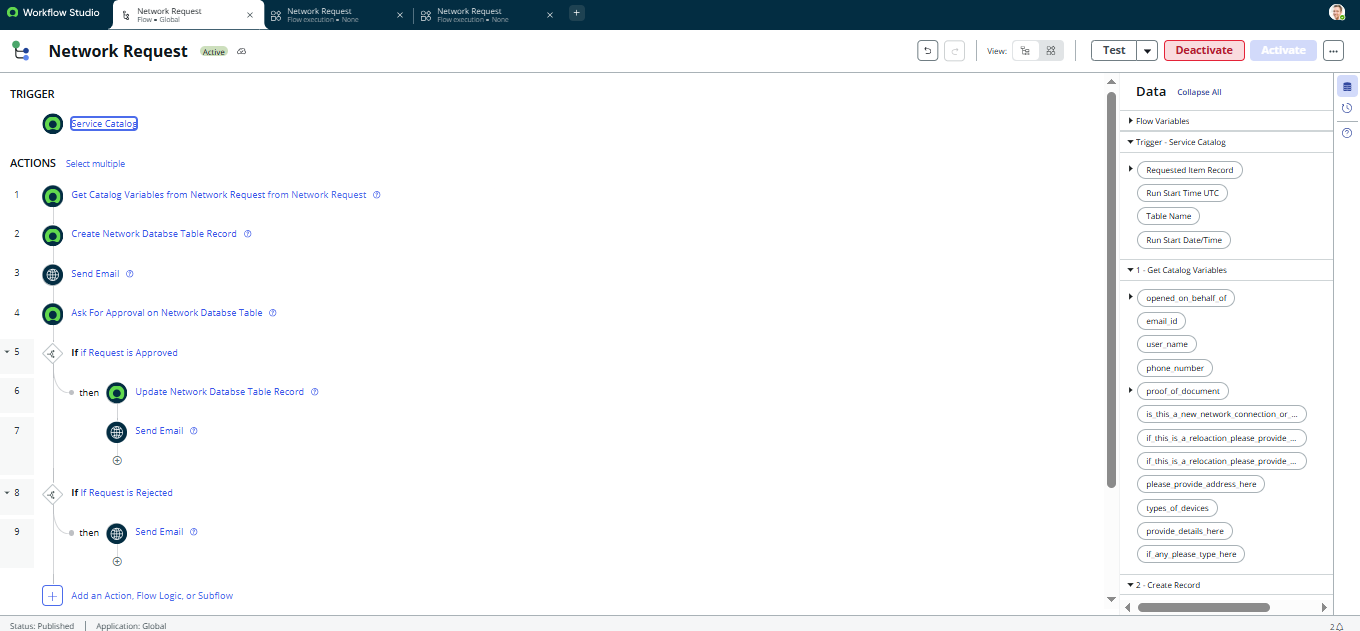
An automated email notification is sent to the requester with request details. 

### Step 7: Ask for Approval Action

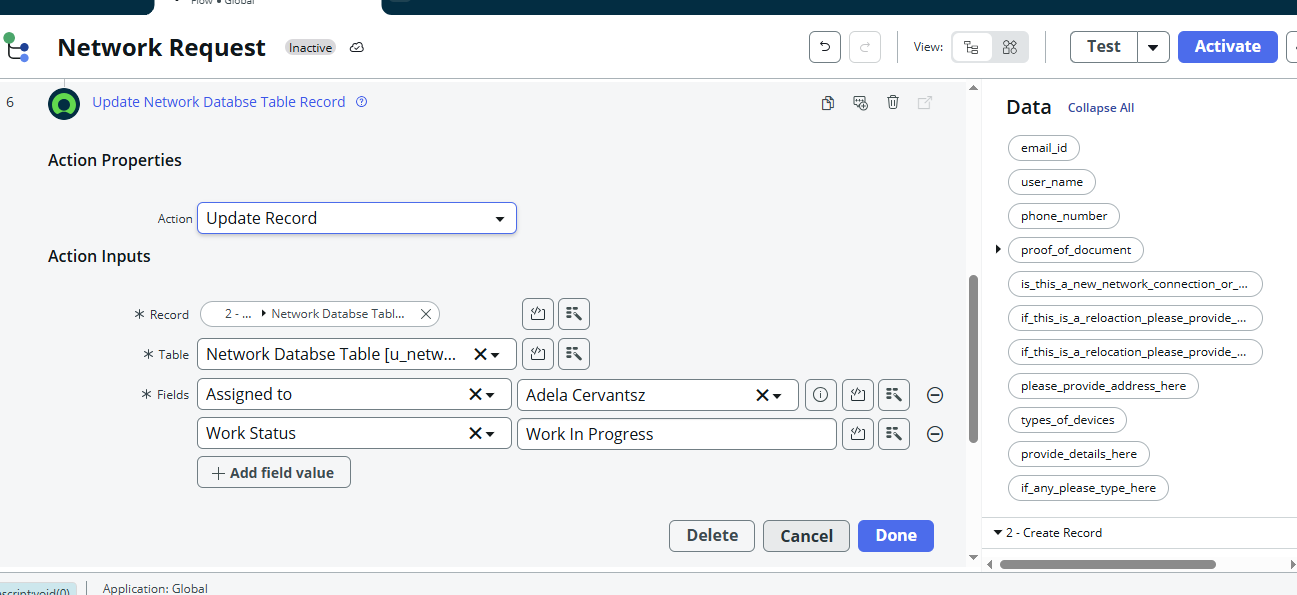
The request is routed to the **Approver (Manager/Network Admin)** for authorization. 

### Step 8: Conditional Flow Logic

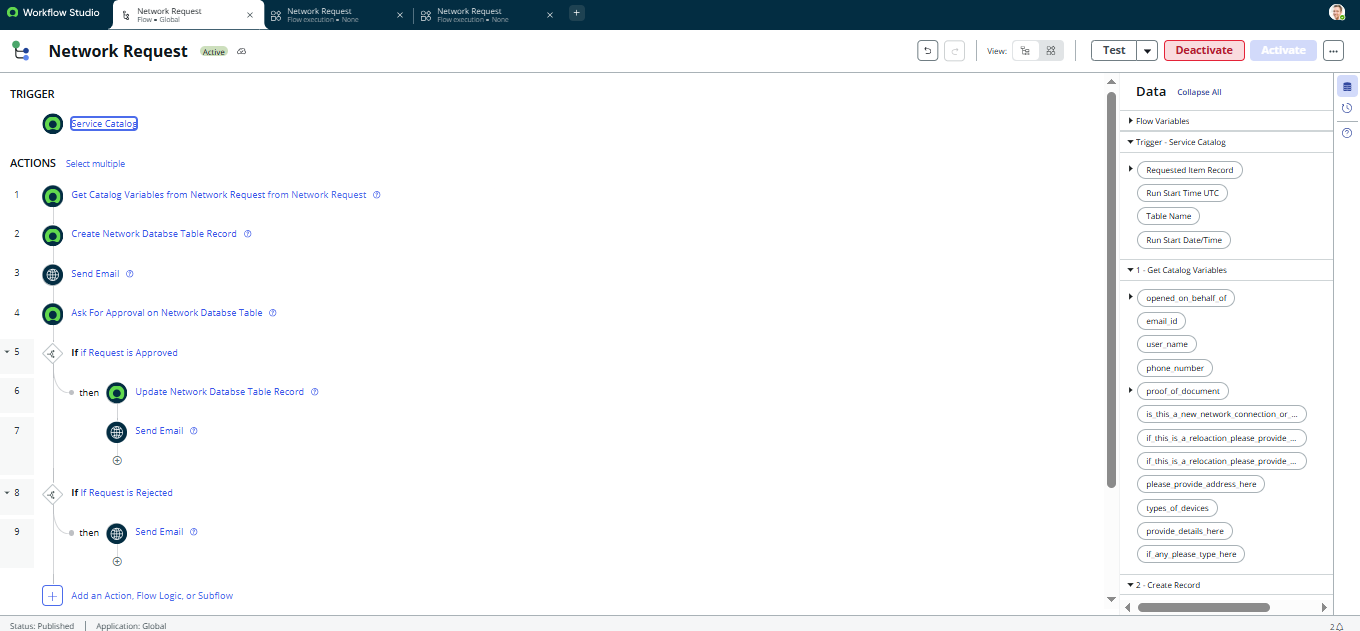
Based on approval:

1. **If Approved → Proceed with request fulfillment**.
2. **If Rejected → Notify requester with reason**.

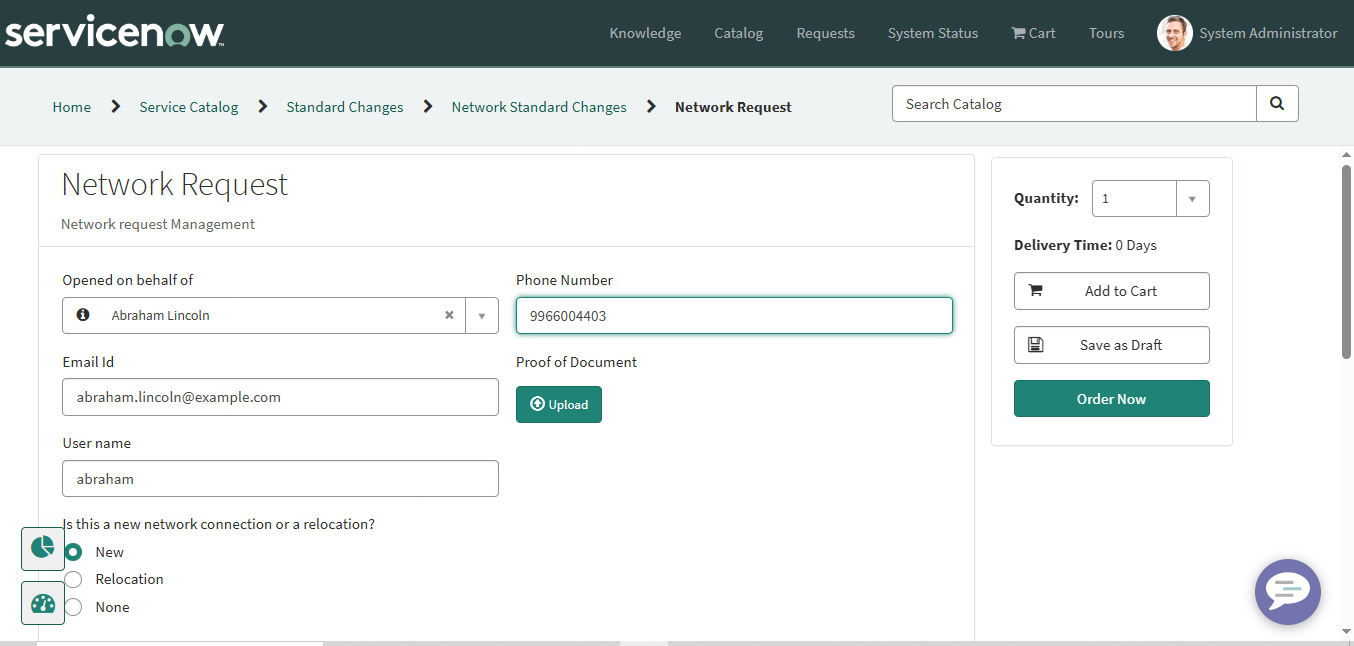
### Step 9: Update Record Action

The record is updated with the latest status (Approved/Rejected). 

### Step 10: Final Flow Design

The complete workflow is now automated end-to-end. 

## Results

1. Users can now raise network requests directly from the catalog.
2. Each request is automatically logged, tracked, and routed.
3. Approvals and notifications are **fully automated**.
4. Manual effort and errors are significantly reduced.

## Conclusion

The project successfully demonstrates the **automation of network requests in ServiceNow**. By leveraging **Catalogs, Variables, Flow Designer, Approvals, and Notifications**, the system ensures faster resolution, transparency, and standardization.

This implementation can be scaled across other IT service requests like hardware provisioning, software installations, and employee onboarding.

## Future Enhancements

1. **Integration with CMDB** – Automatically update configuration items linked to requests.
2. **SLA Tracking** – Attach Service Level Agreements for timely fulfillment.
3. **Chatbot Integration** – Allow employees to raise requests via a virtual agent.
4. **Reporting Dashboard** – Provide real-time analytics on request trends.

## References

1. ServiceNow Documentation – [Servicenow Document](https://www.servicenow.com/docs/)
2. GitHub Project Repository – [Automated Network Request Management in Servicenow](https://github.com/paru19851972/Automated-Network-Request-Management-in-Servicenow.git)
3. Project Screenshots –[Screenshots Folder](https://github.com/paru19851972/Automated-Network-Request-Management-in-Servicenow/tree/main/Screenshots)

# 

## 