Technical Blueprint :

Automated Network Request Management – ServiceNow

1. Overview:

This technical blueprint outlines the **system architecture, automation workflows, data mapping, approval logic, and portal integration** used to implement the Automated Network Request Management solution in ServiceNow.

The solution leverages **Service Catalog, Flow Designer, custom tables, and role-based approvals** to deliver a scalable and compliant automation framework.

2. Flow Designer Workflows:

2.1 Flow Overview:

The automation is implemented using **ServiceNow Flow Designer**, triggered upon catalog item submission.

Trigger:

- Catalog Item → Network Request Submission

Key Actions Used:

- Get Catalog Variables

- Create / Update Record
- Ask for Approval
- If (Conditional Logic)
- Send Email Notification

2.2 Flow Workflow Diagram:

START

|

Catalog Item Submitted

|

Get Catalog Variables

|

Create Record (u_network_database)

|

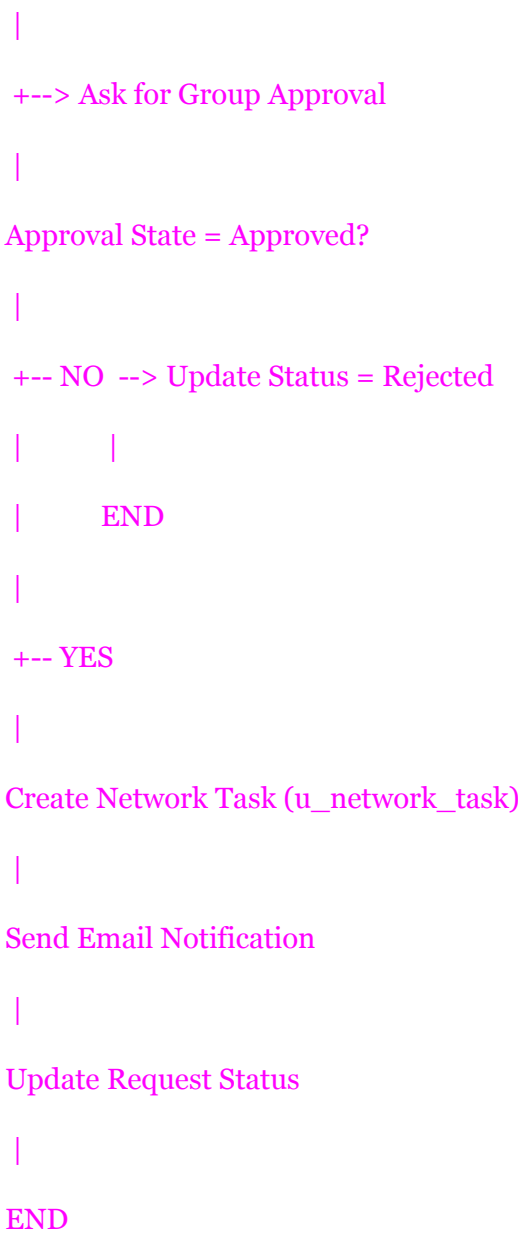
IF Request Sensitivity / Urgency

|

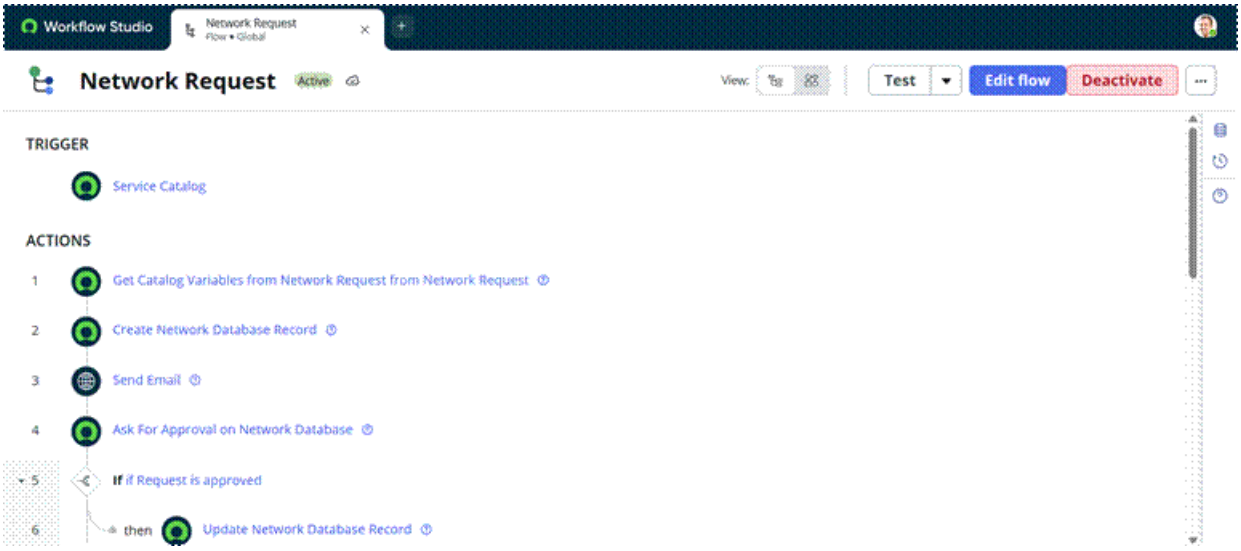
+--> Ask for Manager Approval

|

+--> Ask for Network Security Approval



2.3 Flow Explanation:



📝 Explanation: The flow dynamically retrieves catalog variables, evaluates conditions, routes approvals, and updates records without manual intervention.

3. Variable-to-Field Mapping Logic:

3.1 Mapping Approach:

Catalog variables are captured during submission and mapped to structured fields in a custom table using **Flow Designer**.

Action Used:

- *Get Catalog Variables*
- *Create Record / Update Record*

3.2 Mapping Diagram:

Catalog Item Variables

|

v

Get Catalog Variables

|

v

Field Mapping Logic

|

v

Custom Table (u_network_database)

3.3 Sample Variable Mapping Table:

Catalog Variable	Target Table	Target Field
Request Type	u_network_database	u_request_type
Justification	u_network_database	u_justification
Portal Details	u_network_database	u_portal_details
Urgency	u_network_database	u_urgency
Requested For	u_network_database	u_requested_for

4. Custom Table Schema:

4.1 u_network_database (Request Master Table):

Purpose:

Stores all network request data for tracking, approvals, and reporting.

Field Name	Type	Description
u_request_number	String	Unique request identifier
u_request_type	Choice	Type of network request
u_justification	String	Business justification
u_portal_details	String	Application/portal info
u_urgency	Choice	Request urgency
u_status	Choice	Request state
u_requested_for	Reference (User)	Requester
u_approval_state	Choice	Approval status

4.1 u_network_task (Fulfilment Task Table):

Purpose:

Tracks execution tasks created after approval.

Field Name	Type	Description
u_task_number	String	Task identifier
u_parent_request	Reference	Linked request
u_assigned_group	Reference	Network team
u_task_status	Choice	Task state
u_work_notes	String	Execution notes

5. Approval Condition Logic (Flow Designer):

5.1 Approval Decision Logic:

Approvals are dynamically assigned based on request attributes.

Approvals are dynamically assigned based on request attributes

|

IF Request Type = Standard

|

→ Manager Approval

|

IF Request Type = Security Sensitive

|

→ Network Security Approval

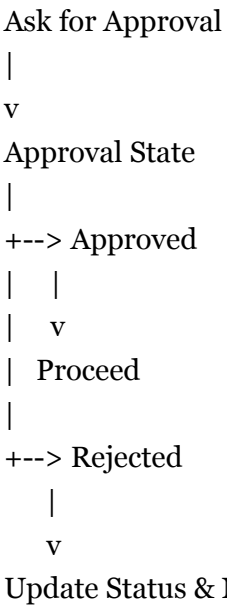
|

IF Department-Specific Request

|

→ Group Approval

5.2 Approval State Validation



This ensures **no request progresses without mandatory approval**.

6. Portal Integration & Widget References:

6.1 Service Portal Usage:

- Standard **ServiceNow Service Portal** used (/sp)
- Network Request catalog item exposed to end users
- No custom widget development required

6.2 Portal Flow:

User Login

|

v

Service Portal (/sp)

|

v

Search: Network Request

|

v

Fill Catalog Form

|

v

Submit Request

|

v

Email Notification Sent

7. Security & Access Control:

- Default ACLs applied to custom tables
- Role-based access enforced for:
    - Read
    - Write
    - Approval actions
- Sensitive fields are protected from unauthorised access

8. Technical Benefits:

✔ Modular and scalable automation

✔ Structured data storage

✔ Dynamic approval routing

✔ Minimal manual intervention

✔ Audit-ready architecture

9. Conclusion:

This technical blueprint demonstrates a robust, enterprise-grade implementation of Automated Network Request Management using ServiceNow.

By combining Flow Designer automation, structured data models, and role-based approvals, the solution delivers efficiency, compliance, and scalability aligned with ITSM best practices.