FINAL PROJECT REPORT

1. INTRODUCTION

1.1 Project Overview

The **Automated Network Request Management System** is a ServiceNow-based solution designed to modernize and streamline the lifecycle of network service requests. The system replaces traditional manual processes with an automated, centralized workflow that improves efficiency, accuracy, and transparency.

By utilizing **Service Catalog**, **Flow Designer**, and **custom data handling**, the solution enables seamless request submission, approval, fulfillment, and tracking. It also enhances governance by maintaining audit trails, enforcing approval rules, and supporting SLA monitoring.

1.2 Purpose

The primary objectives of this project are to:

- Automate network request submission, approval, and fulfillment
- Standardize workflows across departments
- Improve visibility, transparency, and tracking of requests
- Reduce manual intervention and operational errors
- Provide a scalable foundation for future IT service automation

2. IDEATION PHASE

2.1 Problem Statement

Before automation, network service requests were managed manually, leading to:

- Delays in request processing
- Inconsistent and incomplete data collection

- Lack of standardized approval mechanisms
- Poor visibility for requesters and approvers

2.2 Empathy Map Canvas

Stakeholder	Think & Feel	See & Hear	Say & Do	Pain Points
End Users	Frustrated by delays	System inconsistencies	Submit requests	No status visibility
IT Administrators	Overloaded with tickets	Complex configurations	Manage workflows	Manual effort and errors
Network Team	Needs clear data	Incomplete requests	Executes tasks	Data inconsistency
Approvers	Need compliance	Multiple approvals	Approve/reject	Lack of structure

2.3 Brainstorming

- Automation using **ServiceNow Flow Designer**
- Dynamic approvals based on role and request type
- Auto-populated fields to reduce manual entry
- Improved Service Portal user experience
- Role-based access and data security

3. REQUIREMENT ANALYSIS

3.1 Customer Journey Map

1. User logs in through Service Portal
2. Selects **Network Request** catalog item

3. Completes dynamic form with auto-filled data
4. Flow Designer triggers workflow
5. Approval is routed to the appropriate authority
6. Task is created in the Network Database
7. Notifications are sent
8. Request is completed and closed

3.2 Solution Requirements

- Service Catalog with dynamic fields
- Automated approval workflows
- Email notifications at key stages
- Custom table (**u_network_database**) for data storage
- Role-based access control and security

3.3 Data Flow Diagram

Service Portal → Flow Designer → Get Catalog Variables →
Create Record → Approval → Notification → Closure

3.4 Technology Stack

- **Platform:** ServiceNow PDI
- **Tools:** Flow Designer, UI Policies, Catalog UI
- **Database:** Custom Table (u_network_database)
- **Scripting:** Minimal JavaScript for client-side logic

4. PROJECT DESIGN

4.1 Problem–Solution Fit

Manual processes were replaced with automated workflows to ensure:

- Faster approvals
- Accurate data handling
- Improved transparency

4.2 Proposed Solution

- Custom Network Database for structured data
- Flow Designer for automation and approvals
- Dynamic forms for better user interaction
- UI policies for validation and visibility

4.3 Solution Architecture

Components:

- **Frontend:** Service Portal
- **Backend:** Flow Designer, Custom Tables
- **Automation:** Approvals, notifications, record updates
- **Security:** Role-based access control

5. PROJECT PLANNING & SCHEDULING

Phase	Activities	Duration
Phase 1 – Requirement Analysis	Stakeholder mapping, scope definition	1 week
Phase 2 – Backend Development	Table creation, flows, logic	2 weeks
Phase 3 – UI/UX Customization	Portal UI, forms, variables	1 week
Phase 4 – Testing & Security	QA testing, data validation, ACLs	1 week
Phase 5 – Deployment & Documentation	Deployment and user documentation	1 week

6. FUNCTIONAL & PERFORMANCE TESTING

6.1 Performance Testing

- Simulated multiple concurrent requests
- Verified approval routing accuracy
- Confirmed email notifications
- Validated error-free system logs

7. RESULTS

7.1 Output Screenshots

- Dynamic request form
- Approval flow execution
- Email notifications
- Records stored in the Network Database

8. ADVANTAGES & DISADVANTAGES

Advantages

- Faster request processing
- Reduced human errors
- Automated approvals and notifications
- Enhanced visibility and scalability

Disadvantages

- Initial configuration requires ServiceNow expertise
- Dependency on platform availability
- Limited customization without scripting

9. CONCLUSION

The Automated Network Request Management System successfully streamlines network service operations by automating workflows, improving data integrity,

and enhancing user experience. The solution provides a scalable, secure, and efficient foundation for enterprise IT service management.

10. FUTURE SCOPE

- Integration with tools like **Cisco DNA Center** or **Ansible**
- Advanced dashboards and analytics
- Expansion to additional IT services such as access provisioning
- Predictive analytics for workload optimization