Automated Network Request Management in ServiceNow

# 1. Project Overview

The Automated Network Request Management system is developed to transform the way organizations handle network-related service requests. In many enterprises, employees frequently require network services such as new connections, relocation of existing setups, changes in access levels, or device-specific configurations. Traditionally, these requests are managed through emails, phone calls, or manual forms, which often lead to delays, errors, and lack of transparency. The proposed system overcomes these inefficiencies by offering a centralized platform where users can raise requests and track their status, while IT administrators can manage, approve, and resolve requests seamlessly.

By leveraging the ServiceNow platform, the solution integrates Service Catalogs, Variables, UI Policies, Workflows, and Custom Tables to build an automated, user-friendly, and scalable network request management system. The automation not only reduces the workload of IT staff but also ensures timely delivery of services, thereby enhancing user satisfaction and overall organizational productivity.

Key Features:

* • Centralized catalog item for all network-related requests.
* • Dynamic form controls using variables and conditional UI policies.
* • Automated approval processes using Flow Designer and Workflows.
* • Real-time notifications and tracking for end-users.
* • Custom database integration for request details and audit history.
* • Scalability to support different types of network services and requests.

# 2. Objectives

Business Goals:

* • Efficiency: Minimize manual intervention by automating request creation and routing.
* • Accuracy: Ensure correctness in request details through validation and mandatory fields.
* • Transparency: Allow end-users to track the status of their requests in real-time.
* • Cost Reduction: Lower the operational costs by reducing time and manual resources.
* • Scalability: Design the system to adapt to changing organizational needs and growing demand.

Specific Outcomes:

* • Online Service Catalog: Users can submit requests through an intuitive self-service portal.
* • Customizable Forms: Capture detailed inputs using ServiceNow variables and containers.
* • Automated Routing: Assign requests based on type, location, or user group.
* • Database Integration: Store and manage request data in a Network Request Table.
* • Enhanced Communication: Send automated emails/notifications at every stage of the process.
* • Performance Analytics: Generate reports and dashboards for better decision-making.

# 3. ServiceNow Key Features and Concepts Utilized

Service Catalog:

The Service Catalog serves as the entry point for users to submit network-related requests. It includes catalog items such as 'Network Request,' under categories like 'Network and Connectivity.' Each catalog item is designed with multiple variables to capture user details, service type, and device information.

Catalog Variables and UI Policies:

Variables such as text fields, select boxes, checkboxes, and containers are used to design the request form. UI Policies ensure that certain fields are displayed only when specific conditions are met. For example, a relocation-specific section appears only when 'Relocation' is selected as the request type.

Workflows and Flow Designer:

Automated workflows handle the approval process, request routing, and fulfillment. For instance, a request for a new connection may require approval from the manager before being assigned to the network team. Flow Designer simplifies automation by allowing low-code configurations for sending notifications, updating records, and integrating with other modules.

Custom Tables:

A custom 'Network Request Table' is created to store request details, including Request Number, User Information, Location, Device Type, Service Type, Status, Assigned Group, and Resolution Notes. This table ensures that all data is structured and available for reporting and audits.

Client Scripts:

Client scripts provide additional validation and runtime form behavior. For example, a script may disable certain fields once the request is submitted, or enforce validation on device ID formats.

Notifications:

ServiceNow notifications are configured to update users and administrators at each stage. For example, a confirmation email is sent when a request is submitted, an approval email is sent to managers, and closure notifications are sent when the request is resolved.

# 4. Detailed Steps to Solution Design

Data Models:

* • Network Request Table: Stores all request details including type, device, and status.
* • User Profiles: Integrates with ServiceNow’s user table to capture requester details.
* • Request Variables: Define dynamic fields for different request types.
* • Feedback Table: Stores user satisfaction and ratings for closed requests.

User Interface Designs:

* • Service Catalog Item 'Network Request' with clear, intuitive layout.
* • Use of sections and containers for grouping related fields.
* • Categories organized by IT services for easy navigation.

Business Logic:

* • Approval workflows based on request type and user level.
* • Assignment rules to route requests automatically to the right group.
* • SLA definitions to ensure timely resolution of requests.
* • Escalation mechanisms for overdue requests.

# 5. Testing and Validation

Unit Testing:

* • Validate catalog variables, form visibility rules, and client scripts.
* • Ensure workflows execute correctly for different request types.

User Interface Testing:

* • Conduct end-user testing to confirm form usability and accuracy.
* • Validate field behaviors under different conditions.

Integration Testing:

* • Test integration between the Service Catalog, custom table, and workflows.
* • Verify notifications and approvals across different user roles.

Performance Testing:

* • Assess system performance with multiple concurrent requests.
* • Ensure that SLA commitments are met even under high load.

# 6. Key Scenarios Addressed by ServiceNow in the Implementation Project

Network Connection Request:

* • Employees can request new connections by specifying location, device, and bandwidth needs.
* • Automated approvals ensure only valid requests move forward.

Relocation Request:

* • Special fields appear when relocation is chosen, capturing current and new location details.
* • Automatic updates notify users about progress.

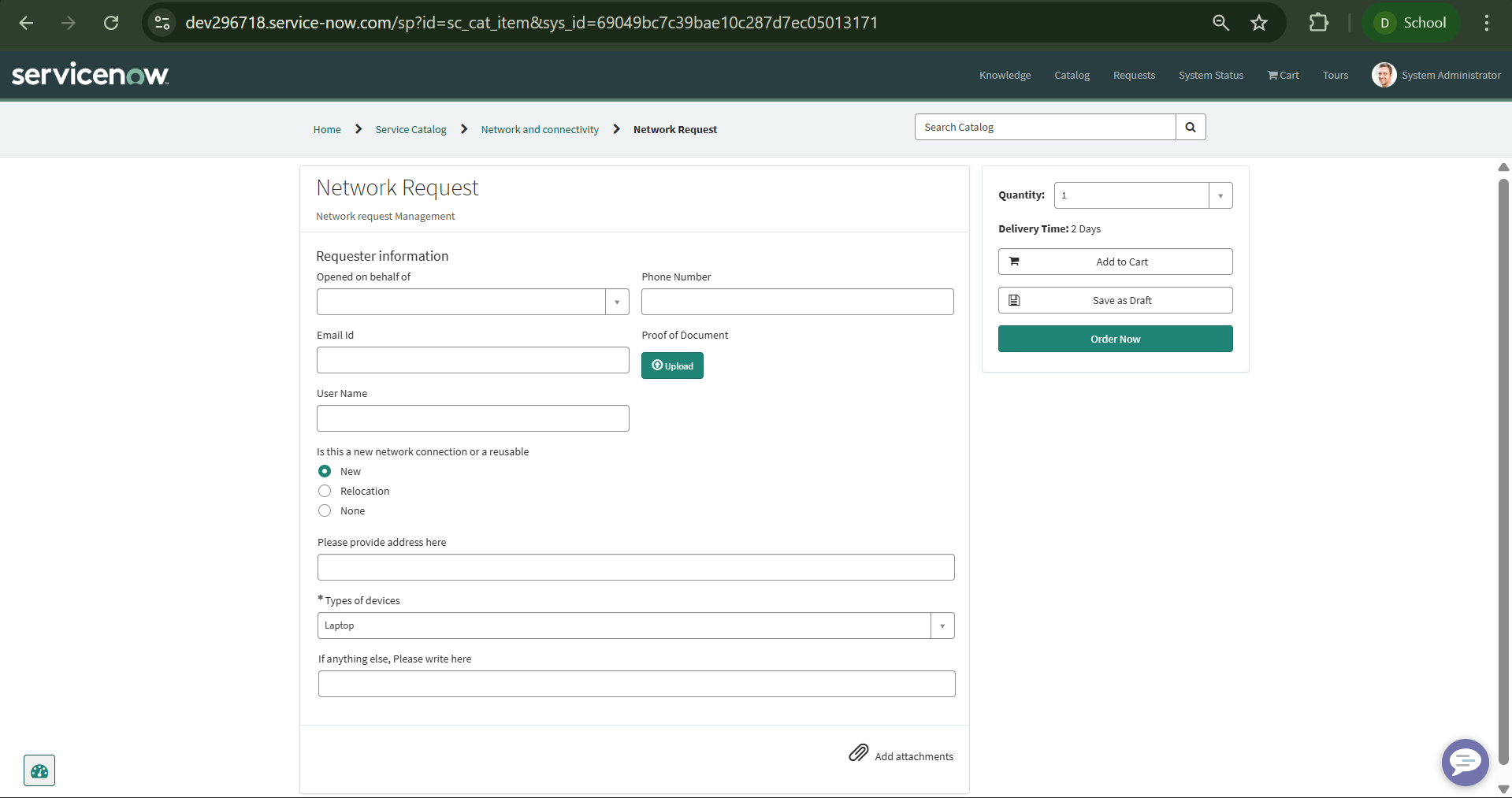
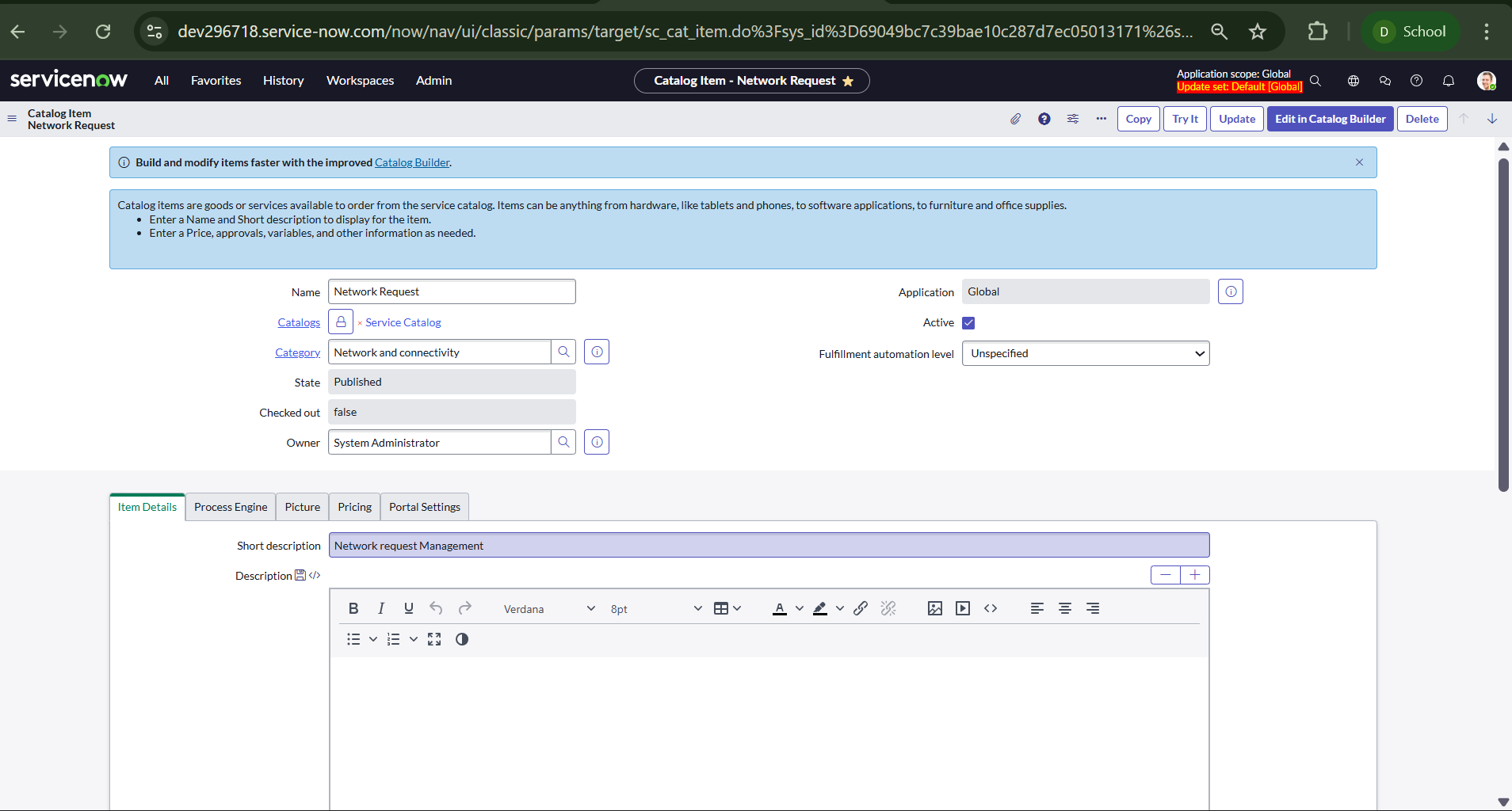
Device Configuration:

* • Users can request configuration changes for specific devices.
* • Workflow ensures assignment to the right team with required technical expertise.

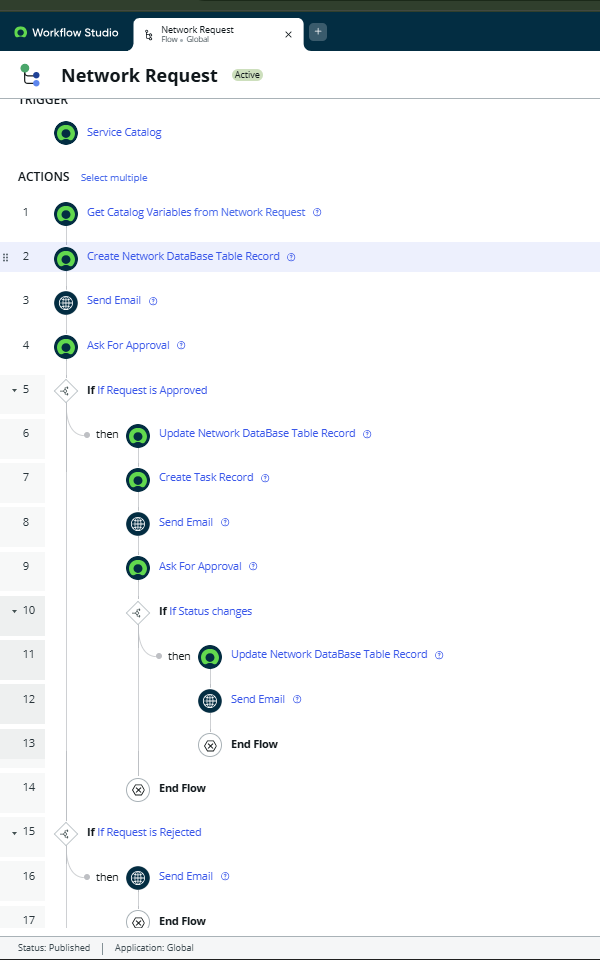
Request Tracking and Approvals:

* • Users can track request progress in real-time.
* • Approvers receive email and portal notifications for pending approvals.

**7. Sample Outputs**

# 



# 8. Conclusion

The Automated Network Request Management project demonstrates the power of ServiceNow in automating and managing IT services. By leveraging Service Catalog, Workflows, Variables, and Custom Tables, the project delivers an efficient, user-friendly, and transparent request management solution. The implementation ensures accuracy in capturing request details, reduces manual effort, and accelerates service delivery.

Key achievements include centralized catalog management, automated routing and approvals, and enhanced communication through notifications. Additionally, the system provides visibility into request status and performance metrics, enabling data-driven decision-making.

Future enhancements may include integration with CMDB to link network requests with configuration items, AI-based chatbots for request creation and resolution, and predictive analytics to forecast demand trends.