

# PROJECT REPORT

## Optimizing User, Group, And Role Management With Access Control And Workflows

### 1. INTRODUCTION

#### 1.1 Project Overview

This project focuses on optimizing user, group, and role management in ServiceNow by implementing structured access control mechanisms and automated workflows. The system ensures proper role-based access, task tracking, approval handling, and accountability throughout the project lifecycle.

#### 1.2 Purpose

- Implement role-based access control (RBAC).
- Automate approval workflows using Flow Designer.
- Ensure structured task management.
- Reduce confusion in task assignments.
- Improve accountability and transparency.

### 2. IDEATION PHASE

#### 2.1 Problem Statement

In a small project management team consisting of a Project Manager (Alice) and a Team Member (Bob), there was difficulty managing tasks efficiently due to lack of clear role definitions, structured workflows, and proper access control. This resulted in confusion regarding approvals and progress tracking.

#### 2.2 Empathy Map Canvas

- Users require clear task visibility and structured approvals.
- Project Managers need control over assignments and approvals.
- Administrators require secure access control and workflow monitoring.

#### 2.3 Brainstorming

- Create users, roles, and groups.
- Implement Access Control Lists (ACLs).
- Design approval workflow using Flow Designer.
- Automate task state updates.
- Enable notification system.

## 3. REQUIREMENT ANALYSIS

### 3.1 Customer Journey Map

- 1 User logs into ServiceNow.
- 2 Task is created in Task table.
- 3 Flow Designer triggers approval process.
- 4 Approver reviews and approves/rejects.
- 5 Task status updated automatically.
- 6 Notification sent to stakeholders.

### 3.2 Solution Requirement

#### *Functional Requirements:*

- User and role creation.
- Group assignment.
- Approval workflow automation.
- Automatic status updates.

#### *Non-Functional Requirements:*

- Secure access control.
- Reliable workflow execution.
- User-friendly interface.
- Fast system performance.

### 3.3 Data Flow Diagram (Explanation)

User → Task Table → Flow Designer → Approval Table → Update Record → Notification → User

### 3.4 Technology Stack

- ServiceNow Platform
- User & Role Management
- Access Control Lists (ACL)
- Flow Designer
- Approval Engine
- Task Table
- Notification System

## 4. PROJECT DESIGN

### 4.1 Problem Solution Fit

The designed system eliminates role confusion and approval delays by implementing structured workflows and strict role-based access control.

## 4.2 Proposed Solution

The solution manages users, groups, and roles efficiently, triggers approvals using Flow Designer, updates task states automatically, and maintains complete tracking within ServiceNow.

## 4.3 Solution Architecture

User Interface → Task Table → Flow Designer → Approval System → Update Record → Notification System

## 5. PROJECT PLANNING & SCHEDULING

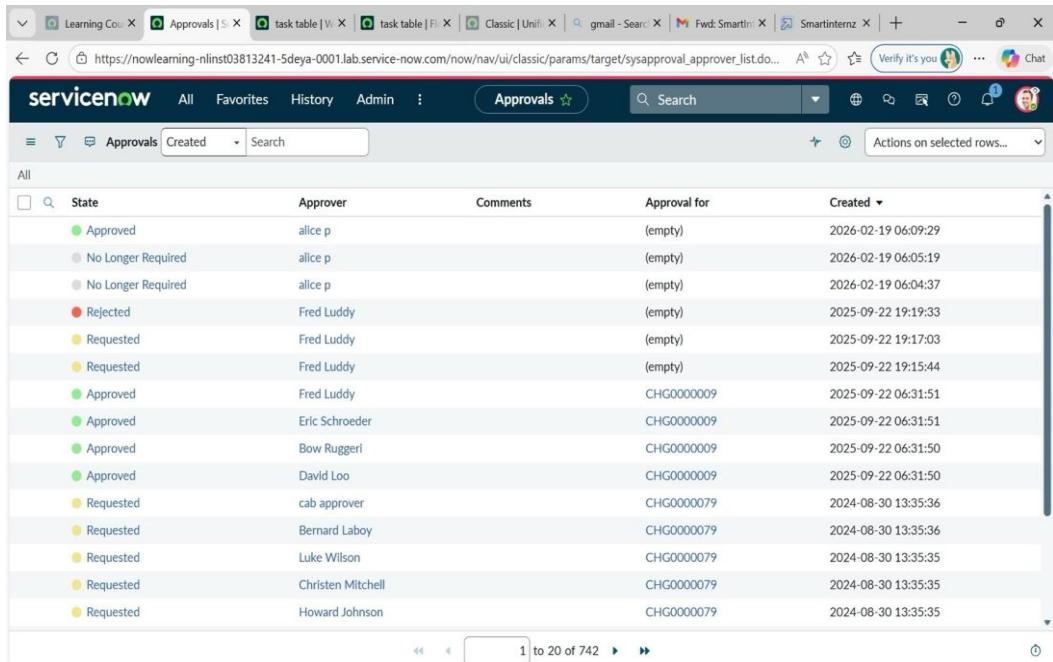
- Phase 1: Requirement Analysis
- Phase 2: User & Role Configuration
- Phase 3: Flow Design
- Phase 4: Approval Setup
- Phase 5: Testing
- Phase 6: Documentation
- Estimated Duration: 2–3 Weeks

## 6. FUNCTIONAL AND PERFORMANCE TESTING

Tested task creation, approval workflow, automatic state updates, rejection handling, and access restrictions. The system performed successfully under normal conditions.

## 7. RESULTS

### 7.1 Output



A screenshot of a ServiceNow web browser window. The address bar shows the URL: https://nowlearning-nlinst03813241-5deya-0001.lab.service-now.com/now/nav/ui/classic/params/target/sysapproval\_approver\_list.do... The main content is a table titled "Approvals" with the following columns: State, Approver, Comments, Approval for, and Created. The table lists various approval records, including rows for "Approved", "No Longer Required", "Rejected", and "Requested" states, each associated with a different approver like alice p, Fred Luddy, Eric Schroeder, etc., and their respective creation dates.

State	Approver	Comments	Approval for	Created
Approved	alice p	(empty)		2026-02-19 06:09:29
No Longer Required	alice p	(empty)		2026-02-19 06:05:19
No Longer Required	alice p	(empty)		2026-02-19 06:04:37
Rejected	Fred Luddy	(empty)		2025-09-22 19:19:33
Requested	Fred Luddy	(empty)		2025-09-22 19:17:03
Requested	Fred Luddy	(empty)		2025-09-22 19:15:44
Approved	Fred Luddy	CHG0000009		2025-09-22 06:31:51
Approved	Eric Schroeder	CHG0000009		2025-09-22 06:31:51
Approved	Bow Ruggeri	CHG0000009		2025-09-22 06:31:50
Approved	David Loo	CHG0000009		2025-09-22 06:31:50
Requested	cab approver	CHG0000079		2024-08-30 13:35:36
Requested	Bernard Laboy	CHG0000079		2024-08-30 13:35:36
Requested	Luke Wilson	CHG0000079		2024-08-30 13:35:35
Requested	Christen Mitchell	CHG0000079		2024-08-30 13:35:35
Requested	Howard Johnson	CHG0000079		2024-08-30 13:35:35

## **workflow:**

The screenshot shows the Workflow Studio interface with multiple tabs open. The active tab is 'task table' under 'EXECUTION DETAILS'. The flow run is labeled 'Test Run - Completed'. The flow statistics show it was run by 'System Administrator' and completed at '2026-02-19 06:23:57' with a duration of '3ms'. The flow starts with a trigger 'task table 2 Created' and contains two actions: 'Update Record' and 'Ask For Approval', both of which were completed. There is also an 'ERROR HANDLER' section.

## **8. ADVANTAGES & DISADVANTAGES**

### ***Advantages:***

- Improved accountability.
- Structured approval process.
- Secure role-based access.
- Automated task tracking
- Clear workflow visibility.

### ***Disadvantages:***

- Dependent on ServiceNow platform.
- Initial configuration complexity.
- Requires user training.

## **9. CONCLUSION**

The project successfully demonstrates optimized user, group, and role management with access control and workflow automation in ServiceNow. The implementation improves efficiency, transparency, and structured task handling within the organization.

## **10. FUTURE SCOPE**

- Dashboard reporting and analytics.
- Advanced scripting validation.
- Integration with additional ServiceNow modules.
- Enhanced role hierarchy automation.

## **11. APPENDIX**

**GitHub & Project Demo Link:**

**<https://github.com/Thanujakonegari110/servicenow-user-role-management>**