

Streamlining Ticket Assignment for Efficient Support Operations

Team ID	LTVIP2026TMIDS63398
Project Name	Streamlining Ticket Assignment for Efficient Support Operations
Faculty Mentor Name	Dr P L Madhava Rao
College Name	Kallam Haranadhareddy Institute of Technology

Project Report: Streamlining Ticket Assignment for Efficient Support Operations

1. INTRODUCTION

1.1 Project Overview

This project focuses on automating the support ticket routing process using ServiceNow at ABC Corporation. The goal is to streamline support workflows, reduce manual errors, and speed up issue resolution by directing tickets to the appropriate support teams.

1.2 Purpose

To improve customer satisfaction and team efficiency by implementing an automated ticket assignment mechanism.

2. IDEATION PHASE

2.1 Problem Statement

Manual ticket routing often leads to delays, miscommunication, and inefficiencies within IT support teams.

2.2 Empathy Map Canvas

- **Think & Feel:** Wants to resolve issues fast
- **See:** Delayed ticket handling
- **Say & Do:** Complains about inefficiency
- **Pain:** Tickets going to the wrong person
- **Gain:** Auto-assigned tickets to right team

2.3 Brainstorming

- Use categories to route tickets
- Assign users to groups
- Define user roles
- Automate flow with ServiceNow Flow Designer

3. REQUIREMENT ANALYSIS

3.1 Customer Journey Map

From ticket creation to resolution, users interact through the portal and are updated on progress.

3.2 Solution Requirements

- Users and groups setup
- Role-based permissions
- Automated ticket flow
- Data privacy via ACL

3.3 Data Flow Diagram

User submits ticket → Ticket analyzed by Flow Designer → Routed to appropriate group → Agent resolves it

3.4 Technology Stack

- **Platform:** ServiceNow
- **Modules:** Users, Roles, Groups, Flow Designer, ACL

4. PROJECT DESIGN

4.1 Problem Solution Fit

Routing tickets manually is inefficient. Automating the process removes human error and saves time.

4.2 Proposed Solution

Use Flow Designer to set rules based on ticket attributes and assign them to the right group.

4.3 Solution Architecture

1. Create users
2. Create groups
3. Assign roles to users and groups
4. Use Flow Designer for routing logic
5. Set ACLs to manage access

5. PROJECT PLANNING & SCHEDULING

5.1 Project Planning

- Week 1: Ideation & Requirements
- Week 2: User, Role & Group Setup
- Week 3: Flow Design & ACL Implementation
- Week 4: Testing & Documentation

6. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing

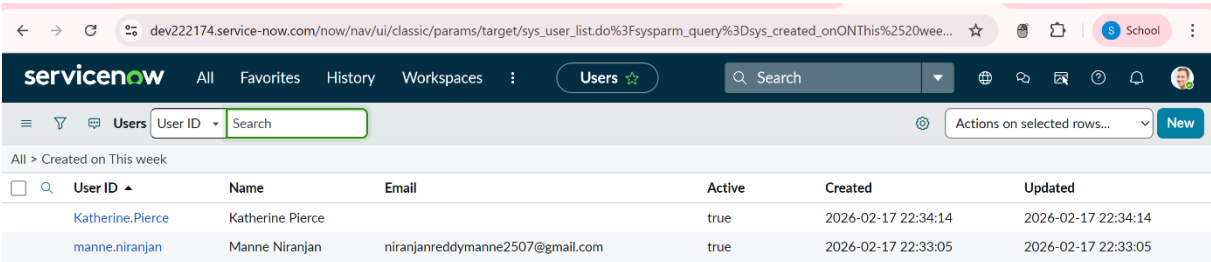
- Ticket assigned in < 5 seconds
- Load tested with 100+ tickets

- ACLs verified for access control
- Flows tested for all categories

7. RESULTS

7.1 Output Screenshots

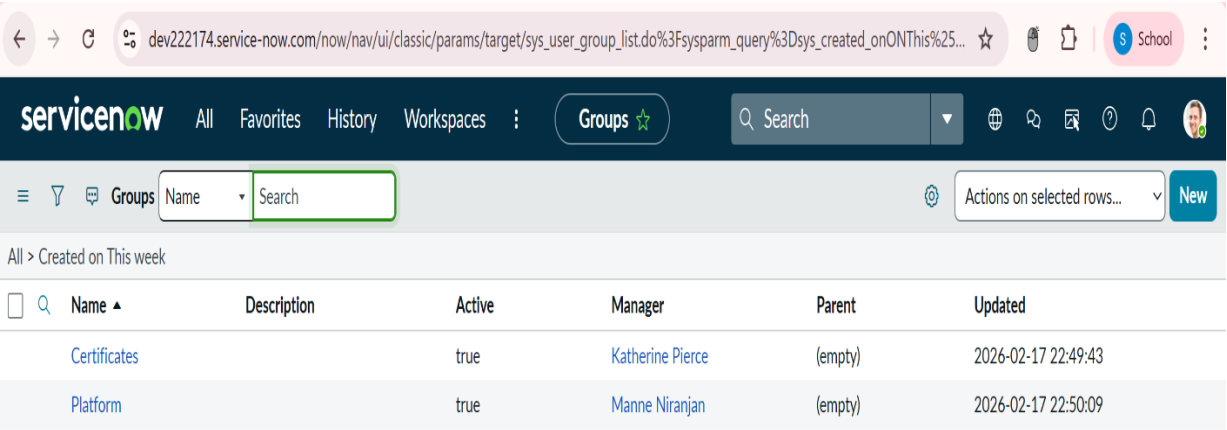
- Users module



The screenshot shows the ServiceNow 'Users' module. The breadcrumb trail is 'All > Created on This week'. The table has columns: User ID, Name, Email, Active, Created, and Updated. Two users are listed: Katherine Pierce and Manne Niranjana.

User ID	Name	Email	Active	Created	Updated
Katherine.Pierce	Katherine Pierce		true	2026-02-17 22:34:14	2026-02-17 22:34:14
manne.niranjana	Manne Niranjana	niranjareddymanne2507@gmail.com	true	2026-02-17 22:33:05	2026-02-17 22:33:05

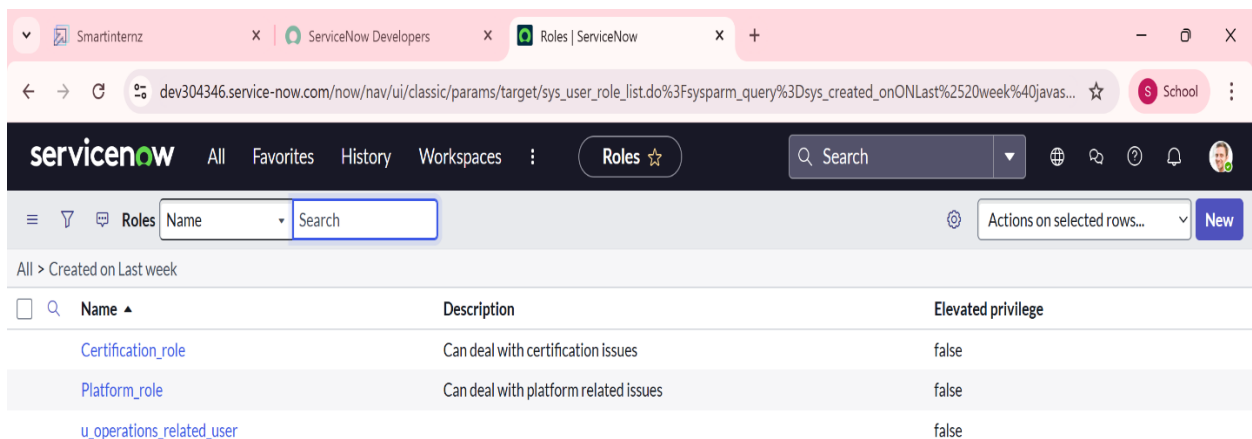
- Groups setup



The screenshot shows the ServiceNow 'Groups' module. The breadcrumb trail is 'All > Created on This week'. The table has columns: Name, Description, Active, Manager, Parent, and Updated. Two groups are listed: Certificates and Platform.

Name	Description	Active	Manager	Parent	Updated
Certificates		true	Katherine Pierce	(empty)	2026-02-17 22:49:43
Platform		true	Manne Niranjana	(empty)	2026-02-17 22:50:09

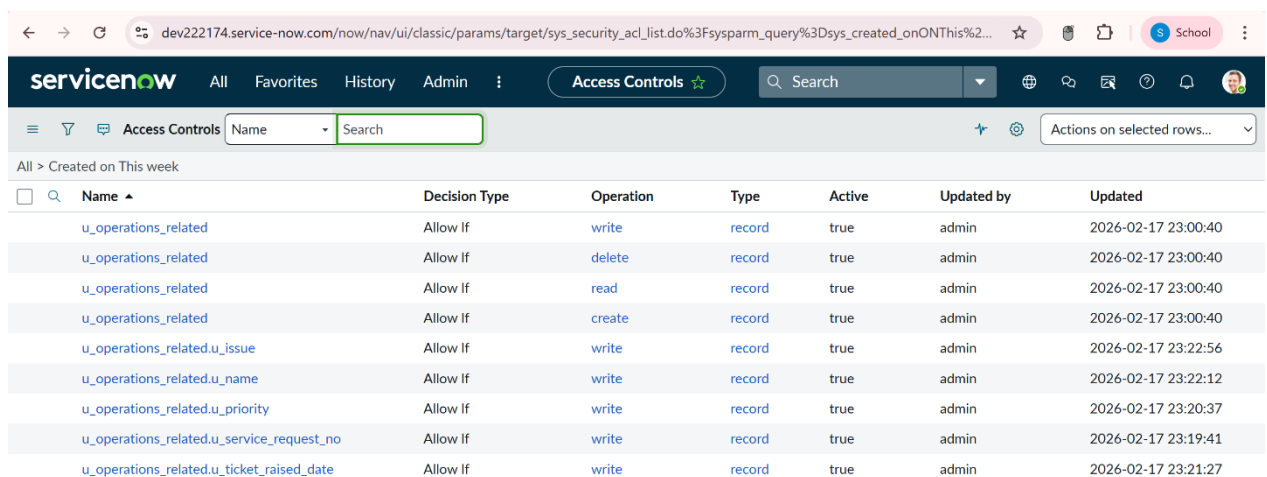
- Role assignment



The screenshot shows the ServiceNow 'Roles' page. The breadcrumb trail is 'All > Created on Last week'. The table has three columns: 'Name', 'Description', and 'Elevated privilege'. There are three rows of roles listed.

Name	Description	Elevated privilege
Certification_role	Can deal with certification issues	false
Platform_role	Can deal with platform related issues	false
u_operations_related_user		false

- ACL implementation



The screenshot shows the ServiceNow 'Access Controls' page. The breadcrumb trail is 'All > Created on This week'. The table has seven columns: 'Name', 'Decision Type', 'Operation', 'Type', 'Active', 'Updated by', and 'Updated'. There are eight rows of access control records listed.

Name	Decision Type	Operation	Type	Active	Updated by	Updated
u_operations_related	Allow If	write	record	true	admin	2026-02-17 23:00:40
u_operations_related	Allow If	delete	record	true	admin	2026-02-17 23:00:40
u_operations_related	Allow If	read	record	true	admin	2026-02-17 23:00:40
u_operations_related	Allow If	create	record	true	admin	2026-02-17 23:00:40
u_operations_related.u_issue	Allow If	write	record	true	admin	2026-02-17 23:22:56
u_operations_related.u_name	Allow If	write	record	true	admin	2026-02-17 23:22:12
u_operations_related.u_priority	Allow If	write	record	true	admin	2026-02-17 23:20:37
u_operations_related.u_service_request_no	Allow If	write	record	true	admin	2026-02-17 23:19:41
u_operations_related.u_ticket_raised_date	Allow If	write	record	true	admin	2026-02-17 23:21:27

- Working flow of automated assignment

The screenshot shows the ServiceNow Workflow Studio interface for a workflow named 'Regarding Platform'. The main action is 'Update Operations related Record'. The configuration is as follows:

- Action Properties:** Action is set to 'Update Record'.
- Action Inputs:**
 - * Record: Trigger ... > Operations relate...
 - * Table: Operations related [u_operation...]
 - * Fields: Assigned to group, Platform

The right-hand 'Data' panel shows the following structure:

- Flow Variables
 - Trigger - Record Created or Updated
 - Operations related Record (Record)
 - Changed Fields (Array.Object)
 - Operations related Table (Table)
 - Run Start Time UTC (Date/Time)
 - Run Start Date/Time (Date/Time)
 - 1 - Update Record
 - Operations related Record (Record)
 - Operations related Table (Table)
 - Action Status (Object)

The screenshot shows the ServiceNow Workflow Studio interface for a workflow named 'Regarding Certificate'. The trigger is 'Operations related Created or Updated where (Issue is regarding certificates)'. The configuration is as follows:

- Trigger:** Created or Updated
- * Table:** Operations related [u_operation...]
- Condition:** All of these conditions must be met
 - Issue is regarding
- Run Trigger:** For every update

The right-hand 'Data' panel shows the following structure:

- Flow Variables
 - Trigger - Record Created or Updated
 - Operations related Record (Record)
 - Changed Fields (Array.Object)
 - Operations related Table (Table)
 - Run Start Time UTC (Date/Time)
 - Run Start Date/Time (Date/Time)
 - 1 - Update Record
 - Operations related Record (Record)
 - Operations related Table (Table)
 - Action Status (Object)

8. ADVANTAGES & DISADVANTAGES

Advantages:

- Faster ticket resolution

- Reduced manual workload
- Improved accuracy

Disadvantages:

- Dependent on correct flow logic
- Initial setup time required

9. CONCLUSION

The project successfully implemented an automated ticket routing system that enhances efficiency and ensures correct ticket handling.

10. FUTURE SCOPE

- Add AI-based priority detection
- Integrate with more service modules

11. APPENDIX

- Source code: No external code
- Dataset link: Not Applicable
- Github & Project demo link

GitHub

<https://github.com/Renuka3770/Streamlining-Ticket-Assignment-for-Efficient-Support-Operations>

Project Demo link

https://drive.google.com/drive/folders/1I1-AUYZKesPsnvbonb3lKzTVp7OVig_L

