Streamlining Ticket Assignment for Efficient Support Operations

Project Documentation format

1. Introduction

Project Title: *Streamlining Ticket Assignment for Efficient Support Operations*

Team ID: LTVIP2026TMIDS74504

Team Size: 5

- **Team Leader**: G Sravani – Project Coordination, Users & Groups Configuration

- **Team Member**: G Chandana – Roles & Custom Table Creation

- **Team Member**: J Meghana – Assigning Roles to Users & Groups

- **Team Member**: M Sanjana – Table-Level Role Assignment & ACL Configuration

- **Team Member**: S Swathi – Flow Designer Automation

2.Project Overview

2.1. Purpose

The purpose of this project is to automate and optimize the ticket assignment process in ServiceNow to ensure faster resolution, improved accuracy, and better support efficiency.

2.2. Features

- Custom table: Operations Related

- Automated ticket routing using Flow Designer

- Dynamic assignment to support groups (Platform / Certificates)

- Role-based access control (ACLs)

- Status tracking and updates
- SLA tracking
- Feedback and reporting
- Performance monitoring dashboard

3. Architecture

3.1. Application Architecture
This project is built using the ServiceNow Platform (Global Application Scope).

3.2. Components Used
- Custom Table (u_operations_related)
- Dictionary Fields
- Access Controls (ACL)
- Roles
- User Groups
- Flow Designer (Automation)
- Notifications
- SLA Tracking
- Reports & Dashboards

3.3. Database Design

Custom table fields include:
- Name
- Issue (Choice field)
- Priority
- Assigned to User
- Assigned to Group
- Service Request Number
- Ticket Raised Date
- Status
- Comments
- Work Notes

4. Setup Instructions

4.1. Prerequisites
- • ServiceNow Developer Instance
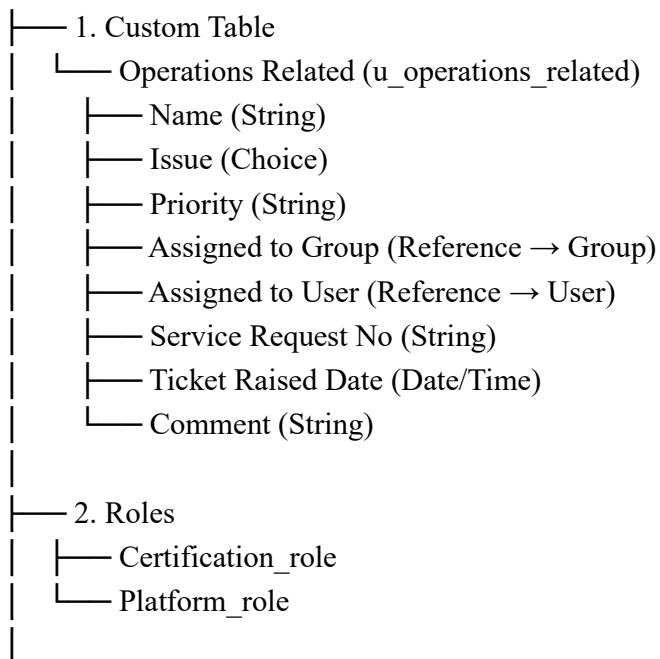- • Admin Access
- • Basic knowledge of Flow Designer and ACL
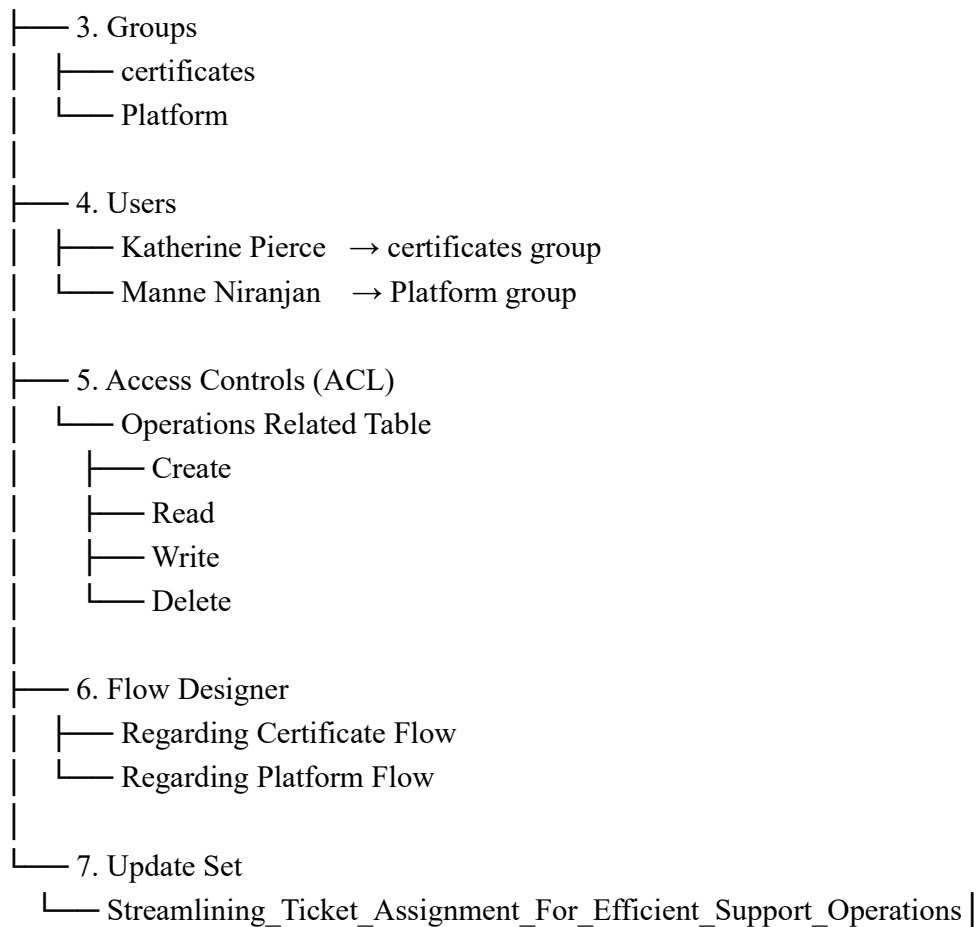
4.2. Installation / Configuration Steps
1. Created custom table: **Operations Related**
2. Added required fields and choice values

3. Created user groups (Platform, Certificates)

4. Created roles and assigned to users

5. Configured ACLs for read, write, create, delete

6. Built flows for automatic ticket assignment

7. Tested automation scenarios

5. Folder Structure

5.1. Client (Flask Frontend)

```
Streamlining_Ticket_Assignment_For_Efficient_Support_Operations
├── 1. Custom Table
│   └── Operations Related (u_operations_related)
│       ├── Name (String)
│       ├── Issue (Choice)
│       ├── Priority (String)
│       ├── Assigned to Group (Reference → Group)
│       ├── Assigned to User (Reference → User)
│       ├── Service Request No (String)
│       ├── Ticket Raised Date (Date/Time)
│       └── Comment (String)
│
├── 2. Roles
│   ├── Certification_role
│   └── Platform_role
│
```

```
├──── 3. Groups
│     ├──── certificates
│     └──── Platform
│
├──── 4. Users
│     ├──── Katherine Pierce   → certificates group
│     └──── Manne Niranjan   → Platform group
│
├──── 5. Access Controls (ACL)
│     └──── Operations Related Table
│         ├──── Create
│         ├──── Read
│         ├──── Write
│         └──── Delete
│
├──── 6. Flow Designer
│     ├──── Regarding Certificate Flow
│     └──── Regarding Platform Flow
│
└──── 7. Update Set
          └──── Streamlining_Ticket_Assignment_For_Efficient_Support_Operations
```

• The project is built as a custom ServiceNow application.
• The core component is the custom table **u_operations_related**.
• Automation is handled through Flow Designer.
• Security is managed using Roles and ACLs.
• Performance monitoring is implemented using Reports and Dashboards.

6. Running the Application

1. User logs into ServiceNow portal
2. Creates a ticket in **Operations Related table**
3. System automatically:
    • Triggers flow
    • Assigns ticket to appropriate group
    • Sends notification
4. Support agent updates status

5. Ticket gets resolved and closed

7. API Documentation

ServiceNow Table API can be used:

Example:

- • GET /api/now/table/u_operations_related
- • POST /api/now/table/u_operations_related Used for integration or external system access.

8. Authentication

Authentication is handled by ServiceNow platform:

• Role-based access control (RBAC)

- • ACL rules configured for:
- • Create
- • Read
- • Write
- • Delete
- • User roles:
- • Admin
- • Operations Related User

9. User Interface

The user interface includes:

- • Service Portal for ticket creation
- • Backend form view for agents
- • Status updates
- • Work notes section
- • SLA tracking panel
- • Reports & dashboards

10. Testing

Test Scenarios

1. Create ticket with Issue = "Regarding Certificate" → Assigned to Certificates group
2. Create ticket with Issue = "Platform Login Issue" → Assigned to Platform group
3. Verify ACL restrictions
4. Verify notification triggers
5. Verify status updates
6. Verify SLA tracking

All test cases passed successfully.

11. Screenshots or Demo
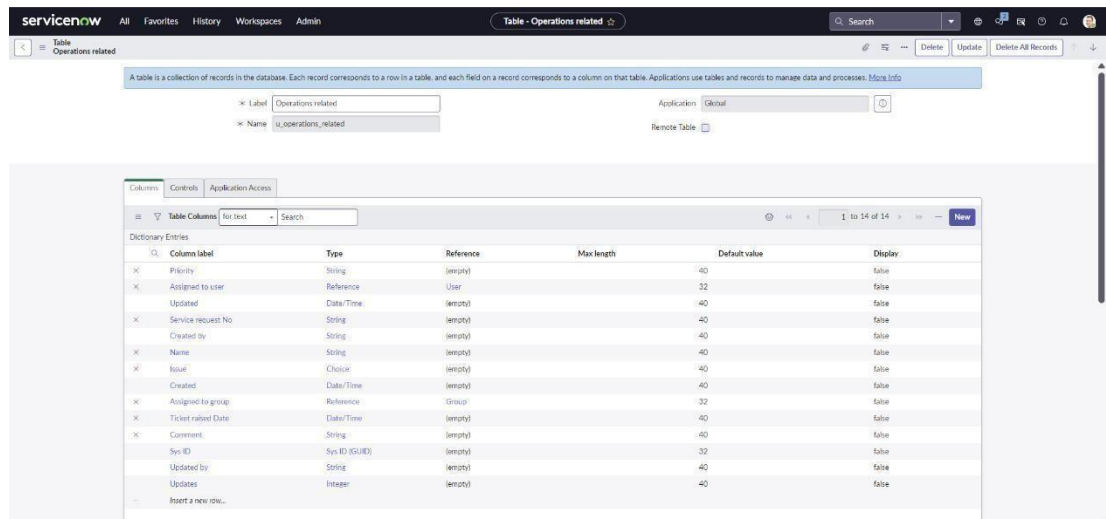
Figure 11.1: Operations Related – Custom Table Creation

Figure11.2: Operations Related – New Record Form

Figure 11.3: Assigned Group automatically triggered

Figure 11.4: Custom Table Columns Configuration (Operations Related Table)
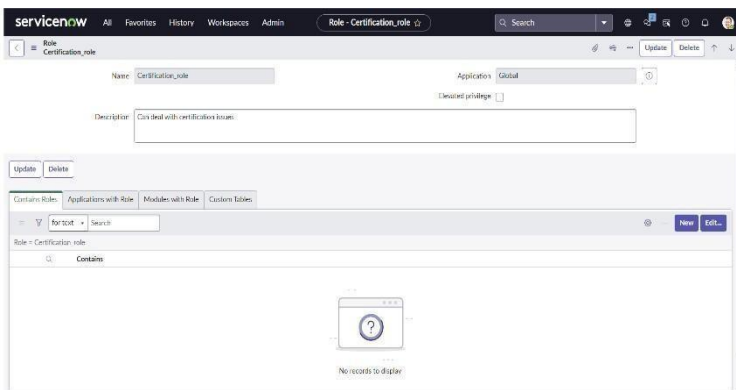
Figure 11.5: Custom Role – Certification_role Configuration
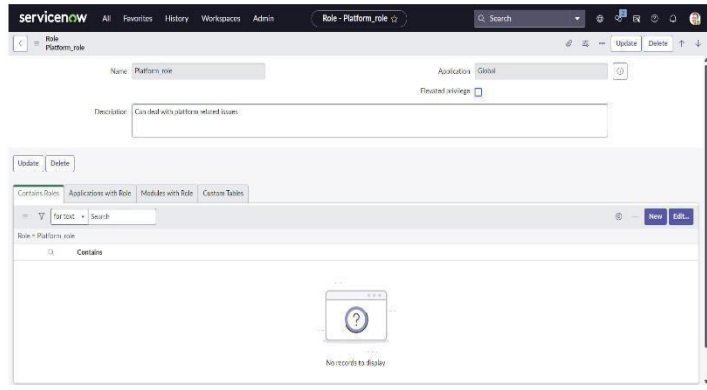
Figure 11.6: Custom Role – Platform_role Configuration
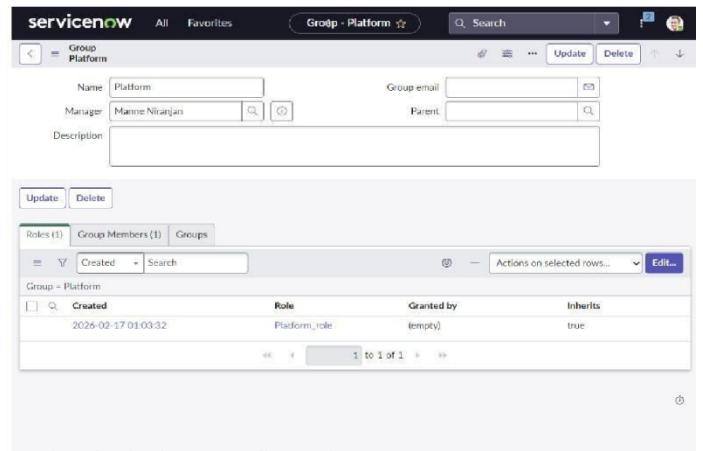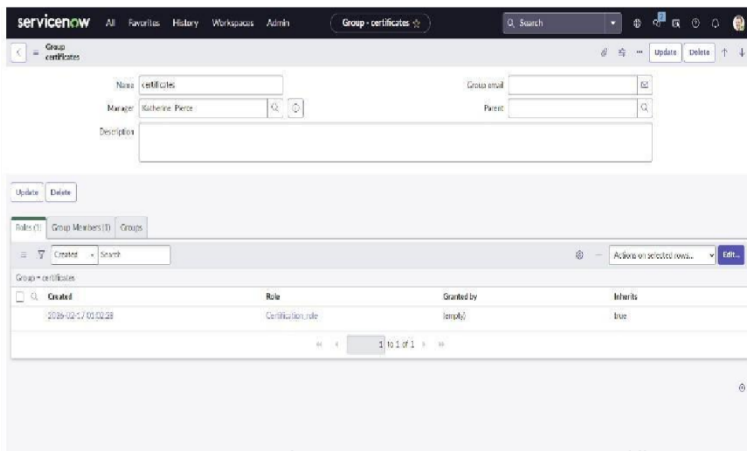
Figure11.7: Support Group –Certificates Configuration & Platform Configuration

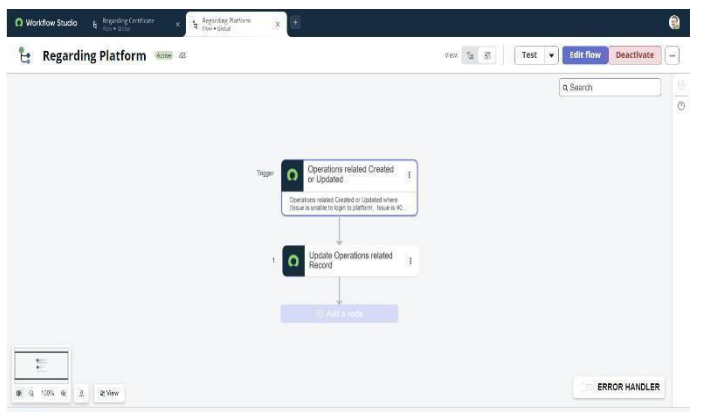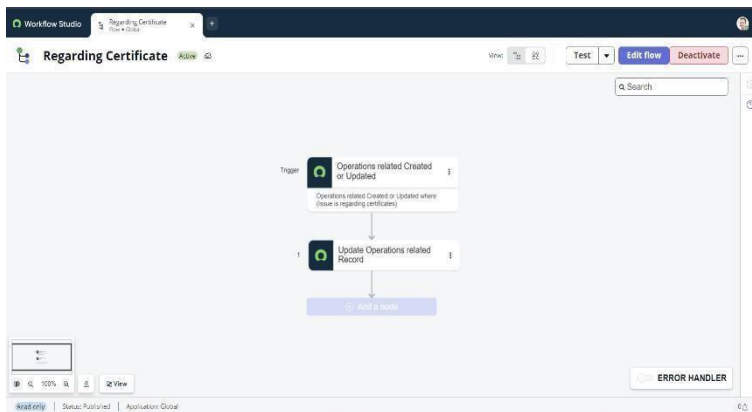*Figure 11.8: Flow Designer – Regarding Certificate Automation Flow & Platform Automation Flow*

Figure 11.9: Update Set – Streamlining Ticket Assignment Configuration

Demo Link:

12. Known Issues

- Manual ticket reassignment required in rare edge cases
- SLA depends on correct configuration
- Requires proper role assignment for access

13. Future Enhancements

- AI-based ticket categorization

- Predictive assignment

- Advanced SLA analytics

- Email integration

- Escalation workflow automation

- Performance tracking dashboard with KPIs