### 

### Streamlining Ticket Assignment for Efficient Support Operations

**Team ID**: 160608

**Category**: ServiceNow

**GitHub Link**: https://github.com/Geethanjali37/Streamlining-Ticket-Assignment-for-Efficient-Support-Operations

## 1. Introduction

In many large enterprises, support teams face inefficiencies due to **manual ticket allocation**, leading to misrouted cases, increased resolution times, and underutilization of resources. This project leverages **ServiceNow Flow Designer** and **Access Control Lists (ACLs)** to implement an **automated ticket routing mechanism**. The system ensures tickets are forwarded to the correct group depending on the issue type, ultimately **reducing delays, enforcing security, and improving customer satisfaction**.

## 2. Objective

The primary aim is to **design and deploy an automated ticket management system** in ServiceNow that:

* Eliminates manual routing efforts.
* Ensures condition-based assignment of tickets.
* Implements **role-based access** to maintain security.
* Enhances overall productivity of IT support operations.

## 3. Skills & Tools Used

* **User & Group Management**
* **Role Creation and Assignment**
* **Custom Table Configuration**
* **Access Control Lists (ACLs)**
* **Flow Designer for Automation**
* **Testing & Validation**

## 4. Modules Implemented

### 4.1 User & Group Management

* Created users to represent support agents (e.g., *Katherine Pierce, Manne Nirajanan*).
* Formed dedicated groups: **Certificate Group** and **Platform Group**.
* Mapped users to groups as per their expertise.

**Outcome:** Structured team setup for role-based ticket handling.

### 4.2 Roles Creation & Assignment

* Defined custom roles: **Certification\_role** and **Platform\_role**.
* Assigned roles to users, groups, and the custom ticket table.
* Roles defined **scope of access** (read/write/modify).

**Outcome:** Controlled and secure access to ServiceNow resources.

### 4.3 Table & Column Configuration

* Created a custom table: **Operations\_Ticket**.
* Added fields like: **issue type, assigned group, description**.
* Configured issue choices: *Login Error, 404 Error, Certificate Issue, Expired User*.

**Outcome:** Centralized data storage for all ticket operations.

### 4.4 Access Control Lists (ACLs)

* Restricted **read/write/modify/delete** access based on roles.
* Applied both **table-level** and **field-level ACLs**.
* Ensured unauthorized users could not access or edit sensitive data.

**Outcome:** Enforced **data security** and **compliance**.

### 4.5 Flow Designer – Ticket Automation

**Flow 1: Certificate Issues**

* **Trigger:** New ticket with issue = "Certificate Issue".
* **Action:** Auto-assign to **Certificate Group**.

**Flow 2: Platform Issues**

* **Trigger:** New ticket with issue = "Login Error / 404 Error / User Expired".
* **Action:** Auto-assign to **Platform Group**.

**Outcome:** Automated routing reduced manual effort and errors.

## 5. Testing & Validation

* Created test tickets for each issue type.
* Verified accurate assignment to relevant groups.
* Checked ACL rules using different user roles.

**Results:**

* Tickets routed correctly every time.
* Unauthorized users were blocked from modifications.
* Each group received only **relevant** tickets.

## 6. Key Learnings

### Technical Learnings

* Practical experience with **ServiceNow Flow Designer**.
* Building **custom tables, groups, and roles**.
* Implementation of **ACLs for security**.
* Workflow automation for real-time ticket management.

## 7. Conclusion

This project successfully automated the ticket assignment process in ServiceNow. By utilizing Flow Designer and ACLs, the solution:

* Improved **efficiency** and **resolution speed**.
* Ensured **secure, role-based data access**.
* Enhanced **resource utilization**.

The system is **scalable, secure, and enterprise-ready**, offering a robust framework for IT support operations in modern organizations.