



Streamlining Ticket Assignment for Efficient Support Operations

Project Overview:

At ABC Corporation, the increasing volume of support requests has highlighted the need for a more efficient and automated ticket management process. Currently, manual assignment of tickets often leads to delays, misrouting, and uneven workload distribution across support teams.

This project introduces an **automated ticket assignment system** designed to streamline support operations. Leveraging predefined rules, intelligent routing mechanisms, and workflow automation, the system ensures that each ticket is directed to the most suitable team or individual.

By implementing this solution, ABC Corporation aims to:

- **Reduce resolution delays** by eliminating manual routing errors.
- **Enhance customer satisfaction** through faster and more accurate issue handling.
- **Optimize resource utilization** by balancing workloads across support teams.
- **Improve operational transparency** with clear assignment logic and reporting.

Ultimately, this initiative supports the organization's goal of delivering **high-quality, responsive, and efficient IT support services** while empowering teams to focus on problem resolution rather than administrative overhead.

Table of Contents

1. Introduction
2. Project Objectives
3. Key Features
4. ServiceNow Developer Setup
5. Project Implementation in ServiceNow
6. Screenshots of Output
7. Conclusion

1. Introduction

In today's fast-paced business environment, timely and accurate handling of IT support requests is essential for maintaining productivity and customer satisfaction. At **ABC Corporation**, the growing number of support tickets has made manual assignment inefficient, often resulting in delays, misrouted issues, and increased workload on support staff.

To overcome these challenges, this project focuses on implementing an **automated ticket assignment system**. By leveraging workflow automation and intelligent routing, the system will ensure that tickets are assigned to the right teams or individuals without manual intervention. This not only improves response and resolution times but also enhances overall efficiency, resource utilization, and service quality within the support department.

2. Project Objective

The primary objective of this project is to **implement an automated ticket assignment system** at **ABC Corporation** that enhances the efficiency of IT support operations. The solution is designed to:

- **Automate ticket routing** to ensure accurate and timely assignment to the appropriate support teams or individuals.
- **Reduce delays in issue resolution** by minimizing manual intervention and routing errors.
- **Improve customer satisfaction** through faster response times and consistent service delivery.
- **Optimize resource utilization** by balancing workloads across support teams.
- **Enhance operational transparency** with clear assignment logic, reporting, and monitoring capabilities.

By achieving these objectives, the project aims to transform the support process into a more **streamlined, reliable, and customer-centric system**.

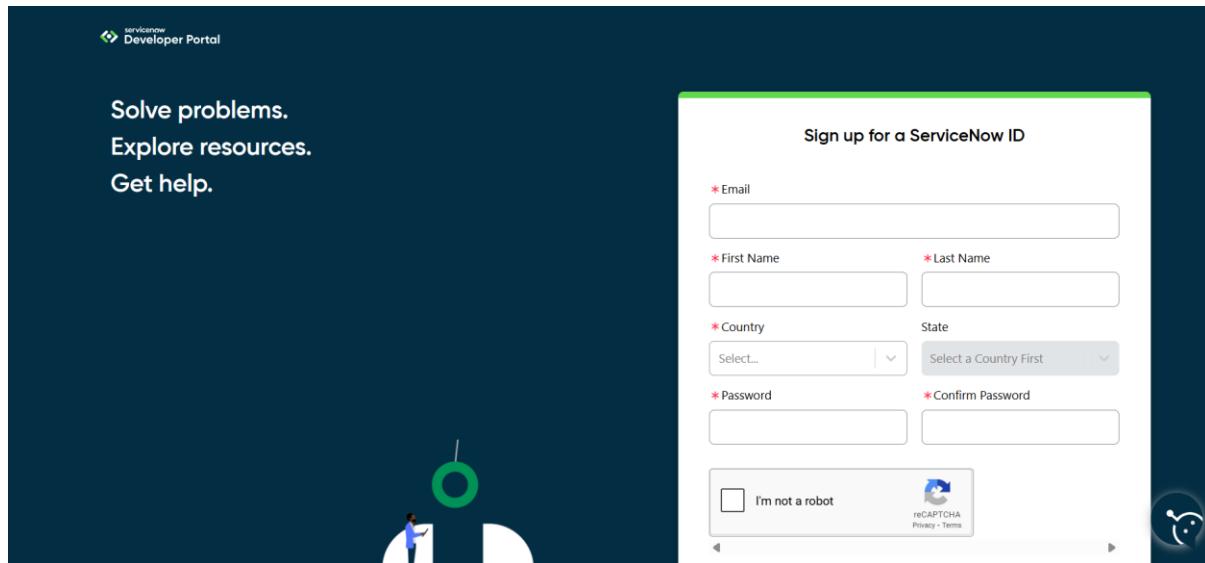
3. Key Features

- **Automated Routing** – Tickets assigned to the right team/person.
- **Dynamic Rules** – Configurable logic based on category, priority, etc.
- **Load Balancing** – Distributes workload evenly across teams.
- **Escalation Support** – Auto-escalates tickets nearing SLA breach.
- **Notifications** – Real-time alerts for quicker response.
- **Analytics** – Reports on ticket flow and team performance.

4. ServiceNow Developer Setup:

Create a Developer Account

1. Go to ServiceNow Developer Portal(<https://developer.servicenow.com/dev.do>).
- Sign up for a free developer account and fill the following details.



After signing up you will get an verification mail to you provided email id. After the verification your ServiceNow Developer Portal Home Page will appear

Now click on start building it will take you to the section where you can **request a Personal Developer Instance (PDI)** or start using **App Engine Studio** and other tools.

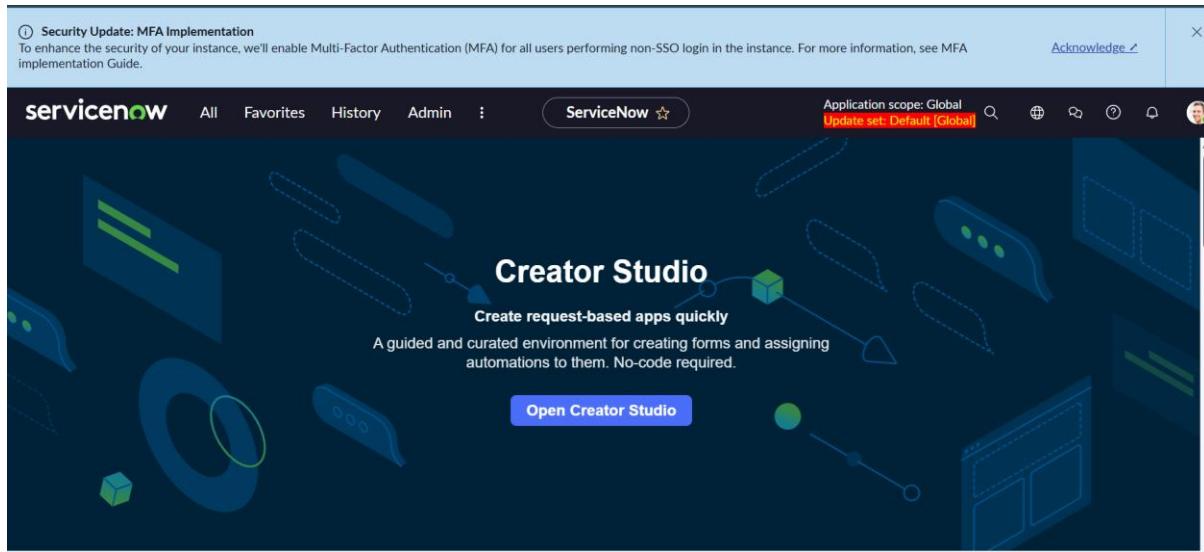
Profile Icon (Top Right Corner) → Manage your account, request instances, and check your developer profile.

A screenshot of the ServiceNow Developer Portal home page. The top navigation bar includes the ServiceNow logo, a search icon, a globe icon, a user icon, and a menu icon. The page title is "Developer". The main content area features a "Hello, GOWTHAMI" greeting and a "Welcome to ServiceNow!" message. It encourages users to "Start using ServiceNow's powerful Now Platform to build applications that make work better for your organization." A "Start Building" button is located at the bottom left. To the right, there's an illustration of three people (two men and one woman) working on laptops, with various icons like a chart, a gear, and a checkmark floating around them.

5. Project Implementation in ServiceNow:

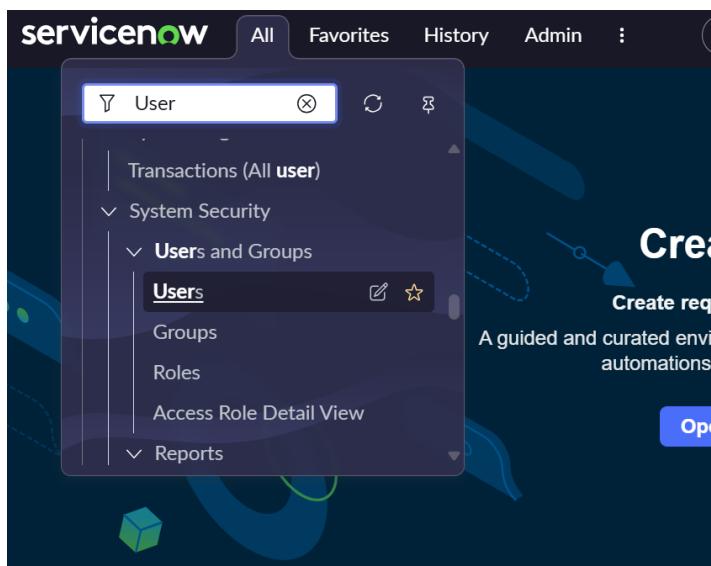
After the instance building is completed the page will be directed to your creator studio.

Creator Studio in ServiceNow provides a **guided, no-code environment** to build applications quickly. It is especially useful for creating **request-based applications** by defining forms, setting up tables, and automating workflows.



a. Creating Users:

1. In the left-hand navigation panel, click on **All** and search for **Users**.
2. Under **System Security**, select **Users**.



3. Click on **New** to create a new user record.

4. Fill in the required details (such as *First Name*, *Last Name*, *User ID*, *Email*, *Password*, *Roles*).

5. Click **Submit** to save the user.

Create the user by filling the following details:

The screenshot shows the ServiceNow User creation interface for a user named "Katherine Pierce". The form fields include:

User ID	Katherine Pierce	Email	
First name	Katherine	Identity type	Human
Last name	Pierce	Language	-- None --
Title		Calendar integration	Outlook
Department		Time zone	System (America/Los_Angeles)
Password needs reset	<input type="checkbox"/>	Date format	System (yyyy-MM-dd)
Locked out	<input type="checkbox"/>	Business phone	
Active	<input checked="" type="checkbox"/>	Mobile phone	
Internal Integration User	<input type="checkbox"/>	Photo	Click to add...

Buttons at the bottom: Update, Set Password, Delete.

Related Links: View linked accounts.

Create another user:

- Repeat the same steps to add a second user with different details.
- Click **Submit** again to save the second user.

The screenshot shows the ServiceNow User creation interface for a user named "Manne Nirajan". The form fields include:

User ID	manne.niranjan	Email	niranjanreddymanne2507@gmail.com
First name	Manne	Identity type	Human
Last name	Niranjan	Language	-- None --
Title		Calendar integration	Outlook
Department		Time zone	System (America/Los_Angeles)
Password needs reset	<input type="checkbox"/>	Date format	System (yyyy-MM-dd)
Locked out	<input type="checkbox"/>	Business phone	
Active	<input checked="" type="checkbox"/>	Mobile phone	
Internal Integration User	<input type="checkbox"/>	Photo	Click to add...

Buttons at the bottom: Update, Set Password, Delete.

Related Links: View linked accounts.

b. Create Groups:

1. In the left-hand navigation panel, click on **All** and search for **Groups**.
2. Under **System Security**, select **Groups**.
3. Click on **New** to create a new group record.
4. Fill in the required details such as:
 - **Name** – Group name.
 - **Description** – Short description of the group
 - **Manager** – Assign a manager if required.
5. Click **Submit** to save the group

Create the group by filling the following details:

The screenshot shows a software interface for creating a group named 'certificates'. The 'Name' field is filled with 'certificates'. The 'Manager' field contains 'Katherine Pierce'. The 'Description' field is empty. The top right features standard toolbar buttons for 'Update', 'Delete', and navigation. Below the form are 'Update' and 'Delete' buttons, and a link to '[SN Utils] Versions (0)'.

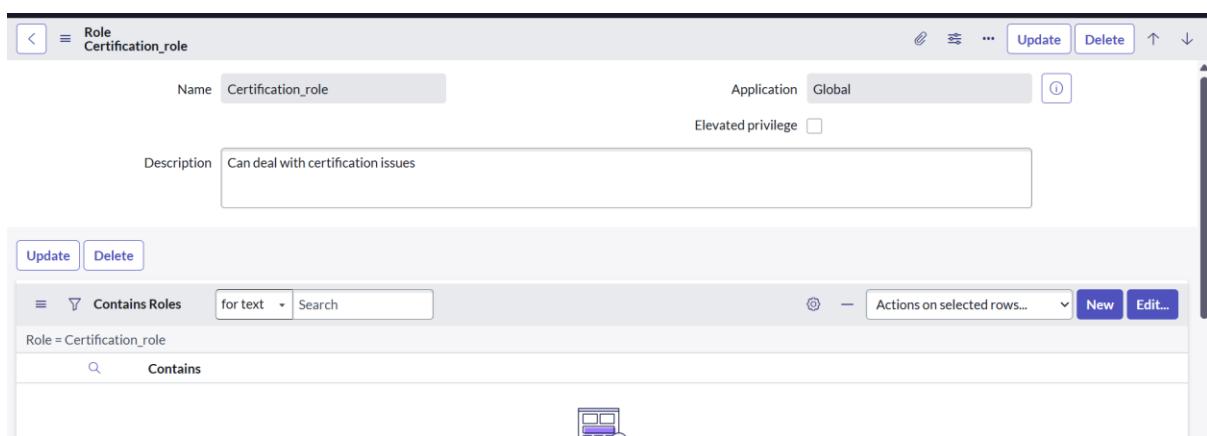
Create another group:

- Repeat the same steps to create a second group with different details.
- Click **Submit** again to save the second group.

The screenshot shows a software interface for creating a group named 'Platform'. The 'Name' field is filled with 'Platform'. The 'Manager' field contains 'Manne Niranjan'. The 'Description' field is empty. The top right features standard toolbar buttons for 'Update', 'Delete', and navigation. Below the form are 'Update' and 'Delete' buttons.

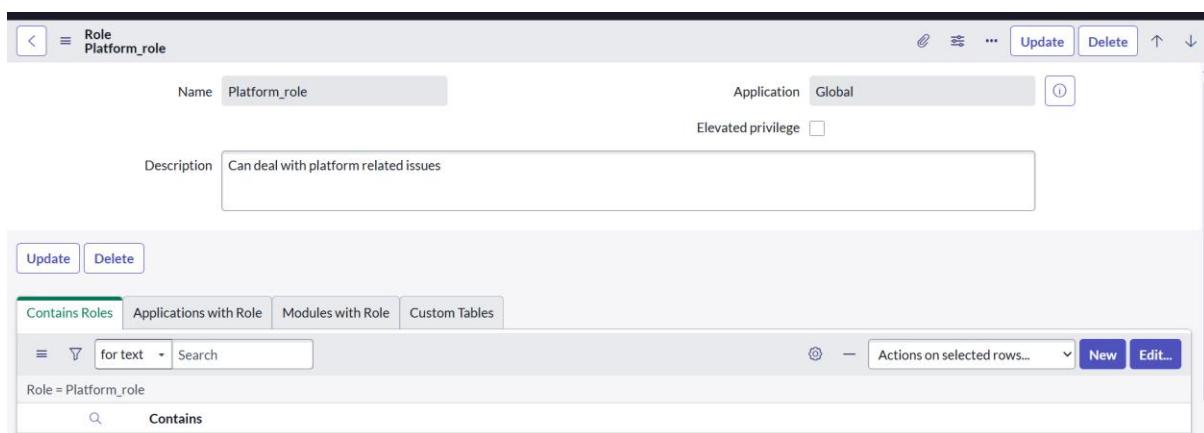
c. Create Roles:

1. In the left navigation panel, select **All** → search for **Roles**.
2. Under **System Security**, click **Roles**.
3. Choose **New** to add a role.
4. Enter the required information, for example:
 - **Name** – A unique role identifier (e.g., *Certificate_role*).
 - **Description** – Brief details of what this role is meant for (e.g., Can deal with certification issues).
5. Click **Submit** to save.



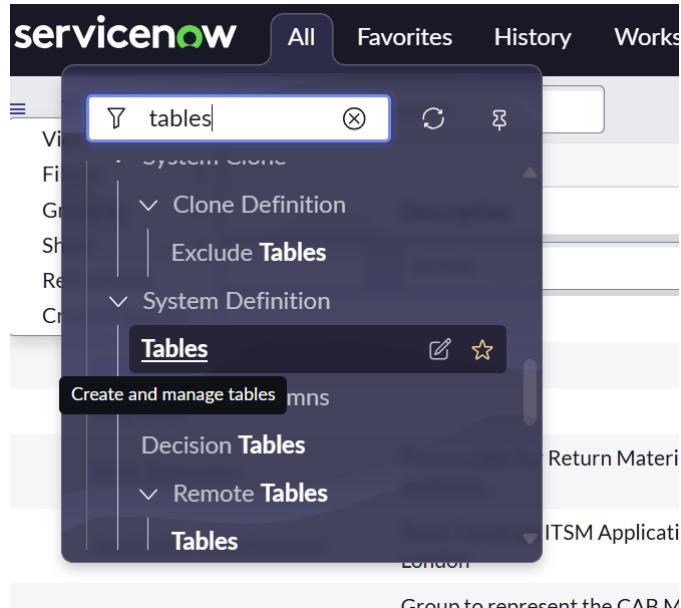
Add another role:

- Follow the same steps to define a second role.
- Role name as *Platform_role* and with suitable description.



d. Create Table:

1. In the left-hand navigation panel, click on **All** → search for **Tables**.
2. Under **System Definition**, select **Tables**.



3. Click on **New** to create a new table.
4. Fill in the required details:
 - **Label:** *Operations related*
 - Check the boxes **Create module** and **Create mobile module**.
 - **New menu name:** *Operations related*

A screenshot of the 'Table Operations related' configuration page. The page shows a 'New' form with the following fields:

- * Label: Operations related
- * Name: u_operations_related
- Extends table: (dropdown menu)
- Application: Global (checkbox)
- Remote Table: (checkbox)
- Create module: (checkbox checked)
- Create mobile module: (checkbox checked)
- Add module to menu: (dropdown menu: -- Create new --)
- New menu name: (text input field)

5. Define the **table columns** as per the project requirements (e.g., *Issue*, *Description*, *Assigned To*, *Status*).
6. Click **Submit** to save the table.

Column label	Type	Reference	Max length	Default value	Display
Assigned to user	Reference	User	32	false	false
Comment	String	(empty)	40	false	false
Updated	Date/Time	(empty)	40	false	false
Service request No	String	(empty)	40	false	false
Assigned to group	Reference	Group	32	false	false
Name	String	(empty)	40	false	false
Created by	String	(empty)	40	false	false
Priority	String	(empty)	40	false	false
Created	Date/Time	(empty)	40	false	false
Sys ID	Sys ID (GUID)	(empty)	32	false	false
Issue	String	(empty)	40	false	false
Ticket raised Date	Date/Time	(empty)	40	false	false
Updated by	String	(empty)	40	false	false
Updates	Integer	(empty)	40	false	false
+ Insert a new row...					

This custom table will act as the data storage for support tickets in our project. It allows us to capture details about issues raised by users and makes it possible to route them automatically to the right team. Without this table, we wouldn't have a centralized place to manage project-specific records.

Adding Choices for the **Issue** Field

1. Navigate to the created table and open **Form Design**.
2. Select the **Issue** field.
3. Add the following choices:
 - *Unable to login to platform*
 - *404 error*
 - *Regarding certificates*
 - *Regarding user expired*

Choices	Description
404 Error	404 error
Regarding certificates	regarding certificates
Regrading User expired	regreding user expired
Unable to login to platform	unable to login to platform

4. Save the form design.

The **Issue** field with predefined choices ensures **standardization** when users log problems. This avoids ambiguity (e.g., someone typing “login issue” vs. “can’t login”) and makes it easier to set up **automation rules** for ticket assignment later.

e. Assign roles & users to groups:

i. Assign roles & users to certificate group:

1. In the left-hand navigation panel, click on **All** → search for **Groups**.
2. Under **System Security**, select **Groups**.
3. Open the **Certificates Group** that was created earlier.
4. In the **Group Members** tab:
 - Click **Edit**.
 - Select *Katherine Pierce* as a member.
 - Save the changes.

The screenshot shows the 'Group Members List' dialog box. At the top, there are filter options: 'Add Filter', 'Run filter', and a help icon. Below these are dropdown menus for 'choose field', 'oper', and 'value'. The main area is divided into two sections: 'Collection' on the left and 'Group Members List' on the right. The 'Collection' section contains a search bar and a list of user names. The 'Group Members List' section contains a list of selected members. Between the two sections are two buttons: a blue right-pointing arrow and a blue left-pointing arrow. At the bottom are 'Cancel' and 'Save' buttons.

Add Filter Run filter ⓘ

-- choose field -- -- oper -- -- value --

Collection

Group Members List

certificates

Katherine Pierce

>

<

Cancel Save

The screenshot shows the 'Groups' list view. The top navigation bar includes tabs for 'Roles (1)', 'Group Members (1)', and 'Groups'. The 'Group Members (1)' tab is active. The search bar shows 'User' and the search term 'Katherine Pierce'. The results table shows one row for 'Katherine Pierce' under the 'User' column. The bottom of the screen shows standard browser controls like back, forward, and search.

Roles (1) Group Members (1) Groups

= User Search

Group = certificates

User Katherine Pierce

5. In the **Roles** tab:

- Click **Edit**.
- Select *Certification_role*.
- Save the changes.

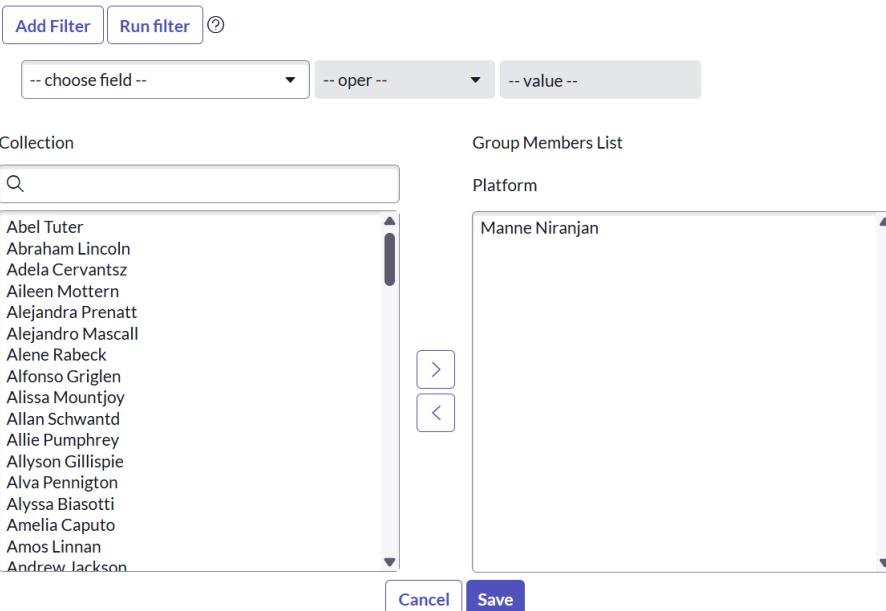
The screenshot shows the 'Roles' tab interface. At the top, there are buttons for 'Add Filter' and 'Run filter'. Below that is a search bar with dropdown menus for 'choose field', 'operator', and 'value'. The main area is divided into two sections: 'Collection' on the left and 'Roles List' on the right. The 'Collection' section contains a list of roles: access_analyzer_admin, action_category_creator, action_designer, activity_admin, activity_creator, actsub_admin, actsub_user, admin, agent_admin, agent_security_admin, agent_workspace_user, ais_admin, ais_high_security_admin, aisa_admin, analytics_admin, analytics_categories_admin, and analytics_filter_admin. The 'Roles List' section contains a single role: 'Certification_role'. Between these sections are two buttons: a blue outlined button with a right-pointing arrow and a blue outlined button with a left-pointing arrow. At the bottom are 'Cancel' and 'Save' buttons.

Roles (1)	Group Members (1)	Groups
<input type="checkbox"/> Created		
2025-08-31 08:23:08	Certification_role	(empty) true

Assigning users and roles to the **Certificates Group** ensures that only authorized members can handle **certificate-related support tickets**. This allows proper **access control**, ensures **responsibility assignment**, and enables the ticket assignment automation to work correctly.

ii. Assign roles & users to platform group:

1. In the left-hand navigation panel, click on **All** → search for **Groups**.
2. Under **System Security**, select **Groups**.
3. From the list of groups, open the **platform group**.
4. In the **Group Members** tab:
 - Click **Edit**.
 - Select *Manne Niranjan* from the available users.
 - Save the changes.



The screenshot shows a table titled 'Groups' with three tabs: 'Roles (1)', 'Group Members (1)', and 'Groups'. The 'Group Members (1)' tab is selected. The table has columns for 'User' and 'Search'. It shows one row for 'Manne Niranjan'. At the top of the table are buttons for 'New' and 'Edit...'. The table also includes standard navigation buttons like arrows and a page number indicator '1 to 1 of 1'.

5. In the **Roles** tab:
 - Click **Edit**.
 - Select *platform group* from the available roles.
 - Save the changes.

Add Filter Run filter ?

-- choose field -- -- oper -- -- value --

Collection Roles List

Platform Platform_role

access_analyzer_admin action_category_creator action_designer activity_admin activity_creator actsub_admin actsub_user admin agent_admin agent_security_admin agent_workspace_user ais_admin ais_high_security_admin aisa_admin analytics_admin analytics_categories_admin analytics_filter_admin

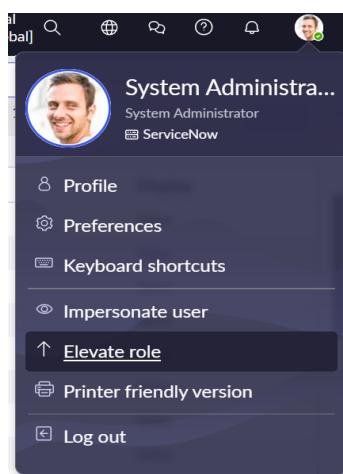
> <

Cancel Save

Roles (1)	Group Members (1)	Groups
Created	Search	
Group = Platform		
2025-08-31 08:26:37	Platform_role	(empty) true

f. Assign role to table:

1. In the left-hand navigation panel, click on **All** → search for **Tables**.
2. From the list, select the **Operations related** table.
3. Go to the **Application Access** tab.
4. Click on your profile (top-right corner).



5. Select **Elevate Role** → choose **security_admin** → click **Update**.

The screenshot shows a 'Dictionary Entries' view for the 'Operations related' table. A modal window titled 'Elevate role' is open, prompting the user to add privileges by selecting a role from a list. The 'security_admin' role is selected and described as granting modification access to High Security Settings. The 'Update' button is visible at the bottom right of the modal.

6. Under **u_operations_related [Read]** operation:

- In **Requires Role**, double-click to insert a new row.
- Add **platform_role** and **certificate_role**.
- Click **Update**.

7. Under **u_operations_related [Write]** operation:

- In **Requires Role**, double-click to insert a new row.
- Add **platform_role** and **certificate_role**.
- Click **Update**

The screenshot displays the 'Access Controls' table and the 'Requires role' list. The 'Access Controls' table has columns for Name, Decision Type, Operation, Type, Active, Updated by, and Updated date. It contains four entries for the 'u_operations_related' operation, each with 'Allow If' as the decision type and 'record' as the type. The 'Requires role' list shows three roles: 'u_operations_related_user', 'Platform_role', and 'Certification_role'.

Assigning roles to the **Operations related** table ensures that only authorized users (those with **Platform Role** or **Certificate Role**) can **read** and **write** records. This provides **data security, controlled access**, and ensures tickets are handled only by the correct teams.

g. Creating Access Control Rules (ACL):

1. Open ServiceNow and log in to your Personal Developer Instance (PDI).
2. In the left-hand navigation panel, click on All → search for **ACL**.
3. Under **System Security**, select **Access Control (ACL)**.
4. Click on **New** to create a new ACL.

5. Fill in the required details for the ACL rule (such as *Table*, *Operation*, *Field*).

The screenshot shows the 'Access Control - New Record' page in ServiceNow. The 'Type' field is set to 'record'. The 'Operation' field is set to 'write'. The 'Decision Type' field is set to 'Allow If'. The 'Application' field is set to 'Global'. The 'Active' checkbox is checked. The 'Admin overrides' checkbox is checked. The 'Protection policy' dropdown is set to 'None'. The 'Name' field contains 'Operations related [u_operations_related]'. The 'Field' dropdown is set to 'Service request No'. The 'Description' field is empty. The 'Applies To' section shows 'No. of records matching the condition: 4' with options to 'Add Filter Condition' or 'Add OR Clause'. Below this are dropdowns for 'choose field', 'oper', and 'value'.

6. Scroll down to the **Requires Role** section.

- Double-click on **Insert a new row**.
- Add the **admin** role.

The screenshot shows the 'Requires role' section. The 'Role' field has 'admin' selected. There is a tooltip 'Showing 1 through 1 of 1' above the list. Below the list are radio buttons for 'Local or Existing' (Existing is selected) and 'Local'.

The screenshot shows the 'Requires role' section again. The 'Role' field has 'admin' selected. There is a tooltip 'Showing 1 through 1 of 1' above the list. Below the list are radio buttons for 'Local or Existing' (Local is selected) and 'Existing'.

7. Click **Submit** to save the ACL.

Similarly, create four ACLs for the following fields:

- Field 1: Issue
- Field 2: Priority
- Field 3: Ticket raised Date

Access Control
New record

① Warning: A role, security attribute, data condition, script or ACL control via reference fields is required to properly secure access with this ACL.

* Type: record * Operation: write Decision Type: Allow If Application: Global Active: Advanced:

Admin overrides:

Protection policy: -- None -- fields: Issue

* Name: Operations related [u_operations_related] Applies To: No. of records matching the condition: 4

Add Filter Condition Add OR Clause

-- choose field -- -- oper -- -- value --

Submit

Access Control
New record

① Warning: A role, security attribute, data condition, script or ACL control via reference fields is required to properly secure access with this ACL.

* Type: record * Operation: write Decision Type: Allow If Application: Global Active: Advanced:

Admin overrides:

Protection policy: -- None -- fields: Priority

* Name: Operations related [u_operations_related]

Description:
 Applies To: No. of records matching the condition: 4

Add Filter Condition Add OR Clause

-- choose field -- -- oper -- -- value --

Submit

Access Control
New record

① Warning: A role, security attribute, data condition, script or ACL control via reference fields is required to properly secure access with this ACL.

* Type: record * Operation: write Decision Type: Allow If Application: Global Active: Advanced:

Admin overrides:

Protection policy: -- None -- fields: Ticket raised Date

* Name: Operations related [u_operations_related]

Description:
 Applies To: No. of records matching the condition: 4

Add Filter Condition Add OR Clause

-- choose field -- -- oper -- -- value --

Submit

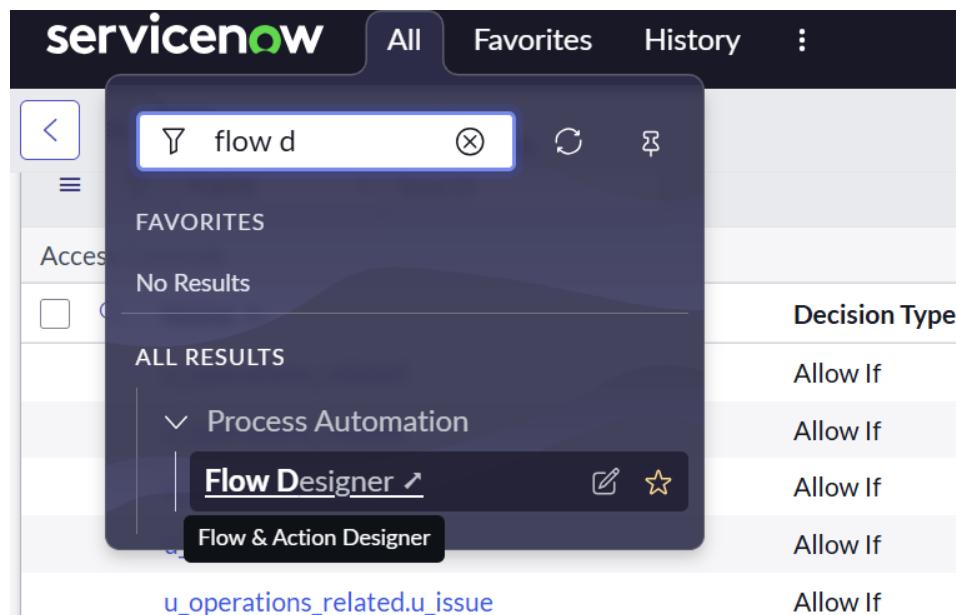
u_operations_related.u_name	Allow If	write	record	true	admin	2025-08-31 08:38:59
u_operations_related.u_priority	Allow If	write	record	true	admin	2025-08-31 08:37:12
u_operations_related.u_service_request_no	Allow If	write	record	true	admin	2025-08-31 08:35:05
u_operations_related.u_ticket_raised_date	Allow If	write	record	true	admin	2025-08-31 08:38:04

ACLs (Access Control Rules) provide **granular security** at the **table and field level**. By creating these ACLs, we ensure that only users with the **admin role** (or other required roles) can access or modify sensitive fields. This prevents unauthorized actions and protects critical data in the **Operations related** table.

h. Flow:

i. Create a Flow to Assign operations ticket to group:

1. **Open ServiceNow** and log in to your instance.
2. In the left navigation panel, click on **All** → search for **Flow Designer**.
3. Select **Flow Designer** under *Process Automation*.



4. Once Flow Designer opens, click on **New** → select **Flow**.

Configure Flow Properties:

- **Flow Name:** *Regarding Certificate*
- **Application:** *Global*
- **Run As User:** *System User*
- Click **Submit**.

Let's get the details for your flow

Flow name * ⓘ
Regarding certificate

Application * ⓘ
Global

Description ⓘ
Describe your flow.

▼ Hide additional properties

Protection ⓘ
-- None --

Run as ⓘ
System user

Cancel **Build flow**

Add a Trigger:

1. Click **Add a trigger**.
2. Search and select **Create or update a record**.
3. In the configuration:
 - o **Table: Operations related**
 - o **Condition:**
 - **Field:** Issue
 - **Operator:** is
 - **Value:** Regarding Certificates
4. Click **Done**.

 **Regarding Certificate** Active Edit View Test Debug Deactivate

TRIGGER

 Operations related Created or Updated where (Issue is Regarding certificates)

Trigger **Created or Updated**

* Table **Operations related [u_operation...]**

Condition All of these conditions must be met

Issue is Regarding **OR** **AND**

or

New Criteria

Run Trigger **For every update**

Advanced Options

Add an Action:

1. Click **Add an action**.
2. Search for **Update Record** and select it.
3. In the **Record field**, drag the required fields from the **Data Panel** on the left.
4. Configure the update:
 - o **Table:** *Auto-assigned from trigger*
 - o **Field:** Assigned to Group
 - o **Value:** Certificates
5. Click **Done**.

ACTIONS Select multiple

The screenshot shows the 'Update Operations related Record' configuration screen. It includes sections for 'Action Properties' (set to 'Update Record'), 'Action Inputs' (with three fields: 'Record' (Operations related...), 'Table' (Operations related [u_operation...]), and 'Fields' (Assigned to group, certificates)), and a 'Done' button at the bottom right.

Final Steps:

- Click **Save** to save the Flow.
- Click **Activate** to enable it.

The screenshot shows the 'Regarding Certificate' flow configuration in Workflow Studio. It includes a 'TRIGGER' section (Operations related Created or Updated where (Issue is Regarding certificates)), an 'ACTIONS' section (1 Update Operations related Record), and an 'ERROR HANDLER' section. On the right side, there is a 'Data' panel showing various flow variables and triggers.

This Flow ensures that whenever a new **Operations related** ticket is created with the issue "*Regarding Certificates*", it will be **automatically routed** to the **Certificates group**. This eliminates manual assignment, speeds up issue resolution, and improves efficiency in ticket handling.

ii. Create a Flow to Assign operations ticket to Platform group:

1. **Open ServiceNow** and log in to your instance.
2. In the left navigation panel, click on **All** → search for **Flow Designer**.
3. Select **Flow Designer** under *Process Automation*.
4. Once Flow Designer opens, click on **New** → select **Flow**.

Configure Flow Properties:

- **Flow Name:** *Regarding Platform*
- **Application:** *Global*
- **Run As User:** *System User*
- Click **Submit**.

Let's get the details for your flow

Flow name * ⓘ
Regarding platform

Application * ⓘ
Global

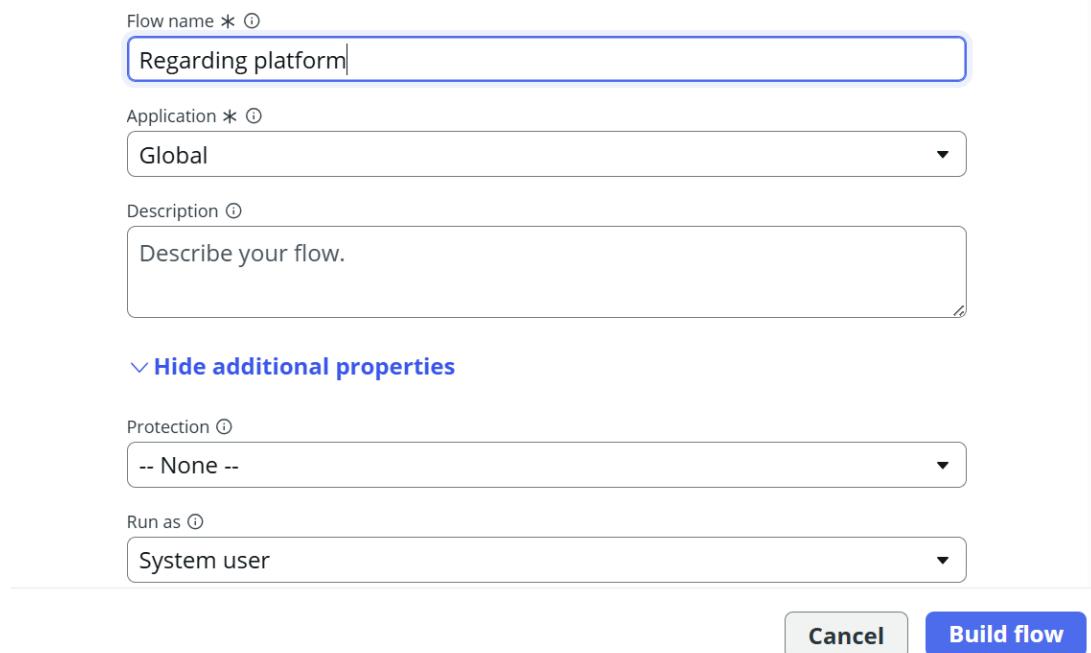
Description ⓘ
Describe your flow.

▼ Hide additional properties

Protection ⓘ
-- None --

Run as ⓘ
System user

Cancel **Build flow**



Add a Trigger:

1. Click **Add a trigger**.
2. Search and select **Create or update a record**.

3. Configure the trigger:

- o **Table:** *Operations related*
- o **Conditions:**
 - **Field:** Issue → is → *Unable to login to platform*
 - **Click New Criteria** → Issue → is → *404 Error*
 - **Click New Criteria** → Issue → is → *Regarding User expired*

4. Click **Done**.

TRIGGER

The screenshot shows the 'Trigger' configuration screen. At the top, it says 'Operations related Created or Updated where (Issue is Unable to login to platform; Issue is 404 Error; Issue is Regarding User expired)'. Below this, the 'Trigger' dropdown is set to 'Created or Updated' and the 'Table' dropdown is set to 'Operations related [u_operation...]'.

The configuration consists of three parallel conditions, each starting with 'Issue' and 'is'. The first condition is 'Issue is Unable to login to platform'. The second condition is 'Issue is 404 Error'. The third condition is 'Issue is Regarding User expired'. Each condition has an 'OR' button to its right, and all three conditions have an 'AND' button at the far right.

Below these conditions is a 'New Criteria' button. At the bottom left is a 'Run Trigger' dropdown set to 'Once', and at the bottom right is an 'Advanced Options' dropdown.

Add an Action:

1. Click **Add an action**.
2. Search for **Update Record** and select it.
3. In the **Record field**, drag the fields from the **Data Panel** on the left.
4. Configure the update:
 - o **Table:** *Auto-assigned from trigger*
 - o **Field:** Assigned to Group
 - o **Value:** Platform
5. Click **Done**.

ACTIONS Select multiple

The screenshot shows the configuration for an 'Update Record' action. The 'Action' dropdown is set to 'Update Record'. The 'Record' field is set to 'Trigger ... ▶ Operations relate...' with a delete icon. The 'Table' field is set to 'Operations related [u_operation...]' with a delete icon. The 'Fields' section contains 'Assigned to group' (with a dropdown and delete icon) and 'Platform' (with a dropdown and delete icon). A '+ Add field value' button is also present.

Final Steps:

- Click **Save** to save the Flow.
- Click **Activate** to enable it.

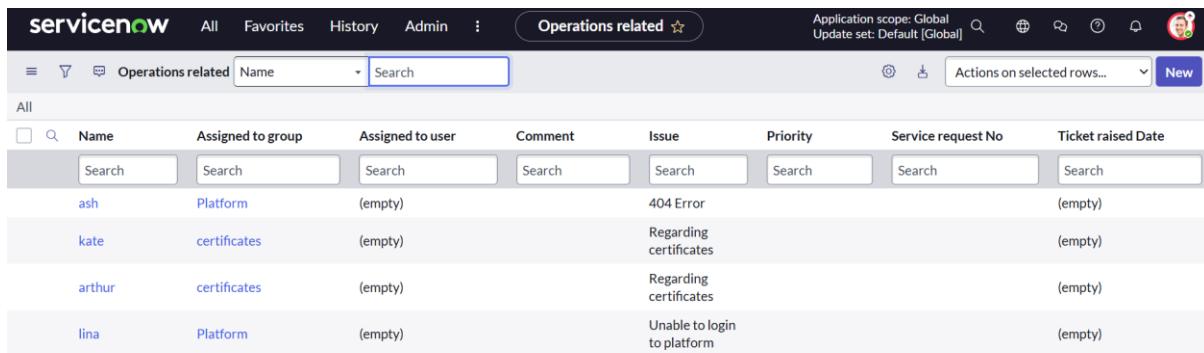
The screenshot shows the 'Regarding Platform' flow in the Workflow Studio. The flow consists of a trigger ('Operations related Created or Updated') and one action ('Update Operations related Record'). The action is currently selected. On the right side, there is a sidebar with sections for 'Data' (Flow Variables, Trigger - Record Created or Updated, Record, Array.Object, Table, Date/Time), '1 - Update Record' (Operations related Record, Operations related Table, Action Status), and status indicators (Status: Published, Application: Global).

This Flow ensures that all tickets in the **Operations related** table with issues like “*Unable to login to platform*”, “*404 Error*”, or “*Regarding User expired*” are **automatically assigned** to the **Platform group**. This reduces manual intervention, speeds up resolution times, and ensures tickets reach the correct support team without delays.

6. Screenshots of Output:

After implementing the Flows in ServiceNow, the ticket assignment works as follows:

- When a ticket is created in the **Operations related** table with the issue “**Regarding Certificates**”, it is **automatically assigned** to the **Certificates group**.
- When a ticket is created with issues like “**Unable to login to platform**”, “**404 Error**”, or “**Regarding User expired**”, it is **automatically assigned** to the **Platform group**



The screenshot shows the ServiceNow interface for the 'Operations related' table. The top navigation bar includes 'servicenow', 'All', 'Favorites', 'History', 'Admin', and a search bar. The table has columns: Name, Assigned to group, Assigned to user, Comment, Issue, Priority, Service request No, and Ticket raised Date. There are four rows of data:

Name	Assigned to group	Assigned to user	Comment	Issue	Priority	Service request No	Ticket raised Date
ash	Platform	(empty)		404 Error			(empty)
kate	certificates	(empty)		Regarding certificates			(empty)
arthur	certificates	(empty)		Regarding certificates			(empty)
lina	Platform	(empty)		Unable to login to platform			(empty)

7. Conclusion:

The implementation of automated ticket assignment in ServiceNow has streamlined the support operations at **ABC Corporation**. By leveraging **Flow Designer**, tickets are now intelligently routed to the correct support groups based on the issue type. This eliminates manual intervention, reduces delays, and ensures efficient handling of incidents.

With the setup of **users, groups, roles, tables, ACLs, and automated flows**, the support process is now:

- **Faster** – Tickets reach the right team instantly.
- **Accurate** – Reduced chances of misrouting.
- **Efficient** – Optimized resource utilization across support groups.
- **Customer-focused** – Improved resolution times lead to higher customer satisfaction.

In summary, this project demonstrates how ServiceNow can be used to **enhance IT service management (ITSM)** by automating repetitive tasks and empowering support teams to focus on resolving issues rather than managing ticket assignments.