

# Phase 5: Project Development Phase

ServiceNow Project: Prevent User Deletion if Assigned to an Incident

Team ID	LTVIP2026TMIDS84378
Team Leader	Valluri Reethi
Team Member	Pothula Lalitha Sri Pani
Team Member	Ramena Mohana Satya Vamsi Kowsik
Team Member	Rudraraju Jogi Subrahmanyam Raju
Date	14 Feb 2026
Document Version	1.0

## 1. Development Overview

This phase documents the implementation steps performed in ServiceNow to develop the solution. The development work includes creating test users, assigning roles, creating and assigning incidents, implementing the Business Rule, and validating the final output with screenshots.

## 2. Environment Setup

Parameter	Value
Platform	ServiceNow
Instance Type	Personal Developer Instance (PDI)
Application Scope	Global
Module Used	Incident Management
Tables Used	sys_user, incident

### 3. Development Steps (with Evidence)

#### Step 1: Create User 1

A new user (User 1) is created in the sys\_user table. This user is later assigned to an incident to test the delete-blocking condition.

The screenshot shows the ServiceNow user profile page for 'Rudraraju Jogi Subrahmanyam Raju'. The form includes fields for User ID (raju), First name (Rudraraju), Last name (Jogi Subrahmanyam Raju), Title, Department, Email (rudrarajugisubrahmanyamraju@gi), Language (-- None --), Calendar integration (Outlook), Time zone (System (America/Los\_Angeles)), Date format (System (yyyy-MM-dd)), Business phone, Mobile phone, Photo (Click to add...), Password needs reset, Locked out, Active (checked), Identity type (Human), and Internal Integration User. Below the form are buttons for Update, Set Password, and Delete. Related links include View linked accounts, View Subscriptions, and Reset a password. The bottom navigation bar shows Entitled Custom Tables, Roles (43), Groups (1), Delegates, Subscriptions, and User Client Certificates.

#### Step 2: Assign Role to User 1

A role is assigned to User 1. Although roles are not required for the Business Rule logic, it is part of standard ServiceNow user setup.

The screenshot shows the ServiceNow Roles table for user 'Rudraraju Jogi Subrahmanyam Raju'. The table has columns for Role, State, Inherited, and Inheritance Count. The roles listed are:

Role	State	Inherited	Inheritance Count
sn_request_read	Active	true	1
sn_itam_recomm.recommendations_read	Active	true	1
sn_sow.it_agent_dashboard_user	Active	true	1
sn_incident_read	Active	true	1
sn_gd_guidance.guidance_user	Active	true	7
viz_creator	Active	true	1
data_manager_user	Active	true	1
interaction_agent	Active	true	6
cmdb_ms_user	Active	true	1
sn_request_approver_read	Active	true	1
certification	Active	true	1
sn_sow.sow_list	Active	true	5
sn_change_read	Active	true	1

#### Step 3: Create an Incident and Assign to User 1

An incident record is created and assigned to User 1 using the Assigned to field. The incident is set to In Progress to represent an active scenario.

servicenow

AllFavoritesHistoryAdmin

Incident - INC0010003

Search

Incident

INC0010003

DiscussFollowUpdateResolveDelete

Number

INC0010003

\* Caller

Rudraraju Jogi Subrahmanyam Raji

Category

Inquiry / Help

Subcategory

-- None --

Service

Service offering

Configuration item

\* Short description

Test incident for user deletion validation

Description

Channel

-- None --

State

In Progress

Impact

3 - Low

Urgency

3 - Low

Priority

5 - Planning

Assignment group

Assigned to

Rudraraju Jogi Subrahmanyam Raji

Related Search Results

Notes

Related Records

Resolution Information

Watch list

Work notes list

Work notes

Work notes

## Step 4: Create Business Rule (When to run)

A Business Rule is created on the User table (sys\_user) with When = Before and Delete = true.

The screenshot shows the 'Business Rule - Prevent User Deletion' configuration page in ServiceNow. The 'When to run' tab is selected. The rule is named 'Prevent User Deletion' and is applied to the 'User [sys\_user]' table. The 'Application' is set to 'Global'. The 'Active' checkbox is checked, and the 'Advanced' checkbox is also checked. The 'When' dropdown is set to 'before' and the 'Order' is 100. The 'Filter Conditions' section is empty, with buttons for 'Add Filter Condition' and 'Add OR Clause'. The 'Actions' section shows 'Delete' is checked, while 'Insert', 'Update', and 'Query' are unchecked.

Business Rule - Prevent User Deletion

A business rule is a server-side script that runs when a record is displayed, inserted, deleted, or when a table is queried. Use business rules to automatically change values in form fields when the specified conditions are met. [More Info](#)

Name: Prevent User Deletion  
Table: User [sys\_user]  
Application: Global  
Active: ☒  
Advanced: ☒

When to run: before  
Order: 100

Filter Conditions: [Add Filter Condition](#) [Add OR Clause](#)

Insert: ☐  
Update: ☐  
Delete: ☒  
Query: ☐

## Step 5: Add Script Logic in Business Rule

The Business Rule script checks the incident table for any record where assigned\_to matches the current user sys\_id. If found, deletion is blocked.

The screenshot shows the 'Business Rule - Prevent User Deletion' configuration page in ServiceNow, with the 'Advanced' tab selected. The 'Script' section is active, showing a JavaScript function that checks for incidents assigned to the current user. The script is as follows:

```
1 (function executeRule(current, previous /*null when async*/) {  
2  
3   var incGr = new GlideRecord('incident');  
4   incGr.addQuery('assigned_to', current.sys_id);  
5   incGr.setLimit(1);  
6   incGr.query();  
7  
8   if (incGr.next()) {  
9     gs.addErrorMessage('This user cannot be deleted because they are assigned to one or  
10      more incidents.');
```

The script is enclosed in a function that takes 'current' and 'previous' parameters. It creates a 'GlideRecord' for the 'incident' table, adds a query for 'assigned\_to' matching 'current.sys\_id', sets a limit of 1, and queries the table. If a record is found, it adds an error message and sets the 'current' record's 'abortAction' to true. The function returns 'current' and 'previous'.

Update Delete

Related Links  
[Run Point Scan](#)

## Step 6: Test Deletion for Assigned User (User 1)

When attempting to delete User 1, ServiceNow blocks the deletion and displays an error message. This confirms the rule is working correctly.

servicenow

AllFavoritesHistoryWorkspaces

Users

Search

Actions on selected rows...

New

This user cannot be deleted because they are assigned to one or more incidents.

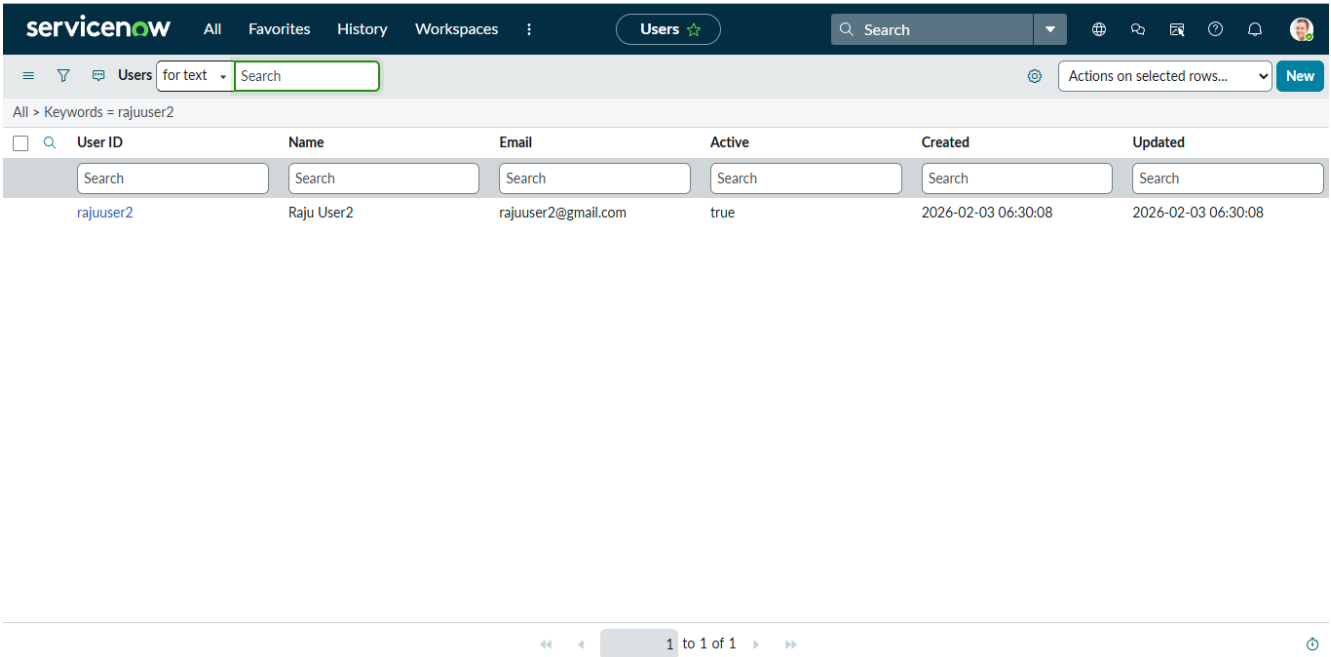
All > Keywords = raju

User ID	Name	Email	Active	Created	Updated
<div>Search</div> <div>raju2003</div>	<div>Search</div> <div>Rudraraju Jogi Subrahmanyam Raju</div>	<div>Search</div> <div>rudrarajujogisubrahmanyamraju@gmail.com</div>	<div>Search</div> <div>true</div>	<div>Search</div> <div>2026-02-03 05:02:02</div>	<div>Search</div> <div>2026-02-03 06:50:55</div>

1 to 1 of 1

## Step 7: Create User 2 and Test Deletion

A second user (User 2) is created without assigning any incident. Deleting User 2 succeeds normally, confirming that the rule does not block unrelated users.



## 4. Business Rule Script Used

The following script was implemented in the Business Rule (Advanced) section:

```
var incGr = new GlideRecord('incident');
incGr.addQuery('assigned_to', current.sys_id);
incGr.setLimit(1);
incGr.query();

if (incGr.next()) {
    gs.addErrorMessage('This user cannot be deleted because they are assigned to one or more incidents.');
```

```
    current.setAbortAction(true);
}
```

## 5. Testing Summary

Test Case	Input Condition	Expected Result	Actual Result	Status
TC-01	Delete User 1 (Assigned to Incident)	Deletion should be blocked	Deletion blocked with error message	PASS
TC-02	Delete User 2 (No incident assigned)	Deletion should be allowed	Deletion successful	PASS

## 6. Conclusion

The development phase successfully implemented the required Business Rule and validated the solution using test users and incidents. The final output confirms that the system blocks deletion of users who are assigned to incidents while allowing deletion for users with no incident assignments.