

## 1. INTRODUCTION

### 1.1 Project Overview

Prevent User Deletion in ServiceNow is a backend-based project developed to restrict unauthorized deletion of user records in the ServiceNow platform. The system uses business rules and role-based access control to ensure that critical user data cannot be deleted without proper authorization, thereby maintaining data integrity and system security.

### 1.2 Purpose

The main purpose of this project is to enhance data security by preventing accidental or unauthorized deletion of user records in ServiceNow and ensuring controlled administrative actions.

## 2. IDEATION PHASE

### 2.1 Problem Statement

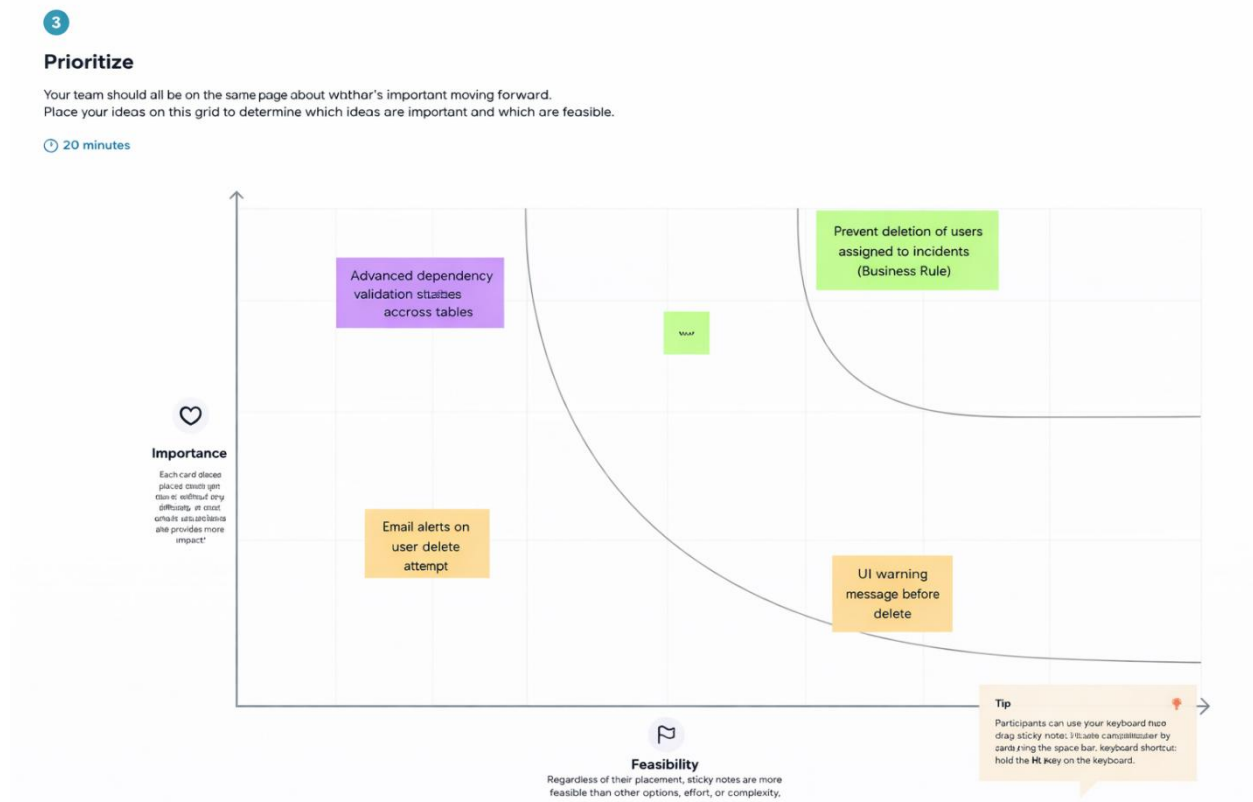
#### Customer Problem Statement Template

Create a problem statement to understand your customer's point of view. The Custommner Problem Statement template helps you focus on what matters to create experiences people will love.

	I am	I'm trying to	But	Because	Which makes me feel
PS-1	ServiceNow Administrator managing incidents	delete or manage user records safely in the system	some users are still assigned to active incidents	there is no default restriction to stop deletion of users linked to incidents	worried about data inconsistency and system errors
PS-2	IT support team member	maintain accurate incident-user relationships	users can be deleted even when incidents are assigned to them	the system does not enforce validation before user deletion	frustrated and concerned about data integrity
PS-2	IT support team member handling incident management	Maintain accurate incident-user relationships	users can be deleted even when incidents are assigned to them	the system does not enforce validation before user deletion	frustrated and concerned about data integrity

Organizations face issues where important user records are accidentally or intentionally deleted in ServiceNow, leading to data loss, security risks, and operational disruption.

## 2.2 Empathy Map Canvas



The target users are ServiceNow administrators and system managers who need secure user management. They think about system safety and data protection, feel concerned about accidental deletions, hear complaints about lost records, and seek a reliable control mechanism.

## Empathy Map Canvas

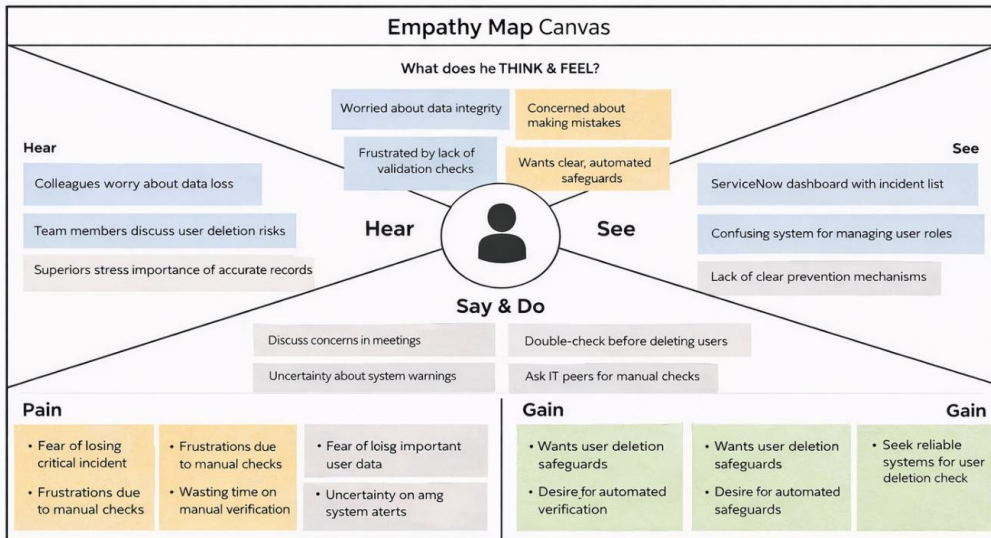
Date: 31 January 2025

Team ID: LTVIP2026TMDIS738881

Project Name:

Prevent User Deletion in ServiceNow

Maximum Marks: 4 Markks



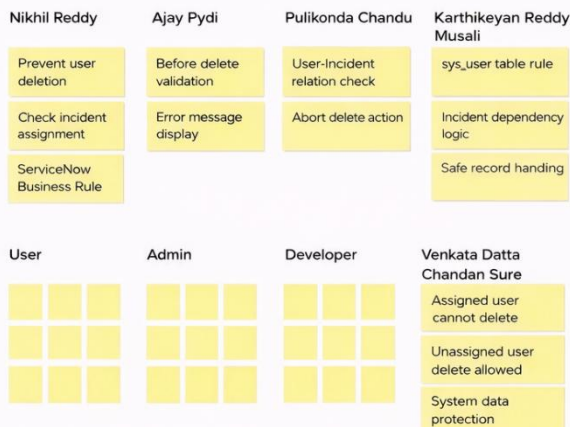
## 2.3 Brainstorming

2

### Brainstorm

Write down any ideas that come to mind that address your problem statement.

🕒 10 minutes



2

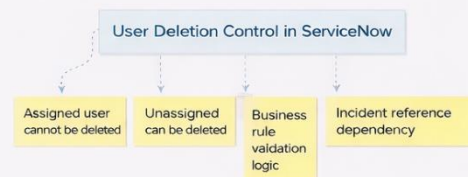
### Group Ideas

Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you can break it up into smaller sub-groups.

🕒 20 minutes

**TIP**

Use Business Rule on sys\_user table to validate assigned\_to field in incident table before deletion.



Multiple ideas were discussed such as role-based deletion control, audit logging, confirmation workflows, and business rule enforcement. The final idea selected was implementing a business rule to block unauthorized user deletion.

### 3. REQUIREMENT ANALYSIS

#### 3.1 Customer Journey Map

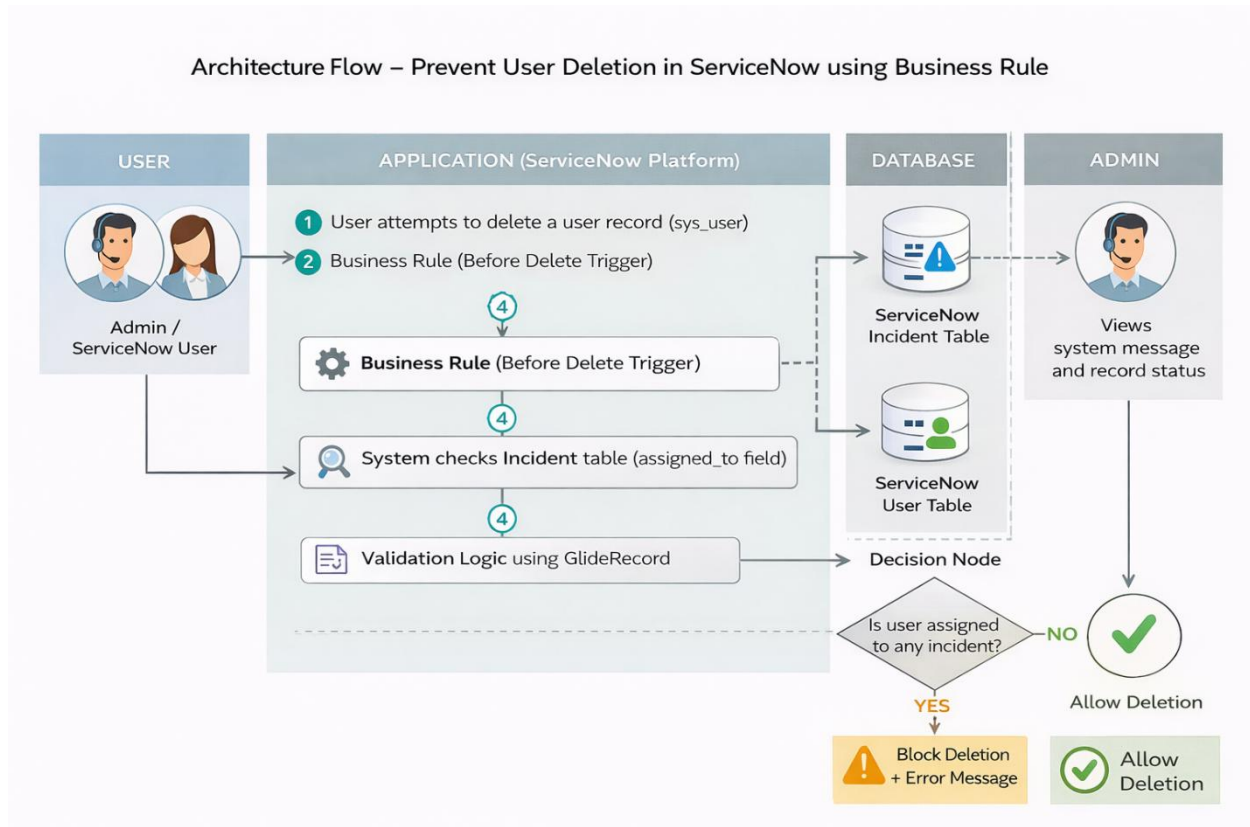
Admin logs into ServiceNow → accesses user records → attempts deletion → system validates role and permissions → unauthorized deletion is blocked → authorized actions are logged for monitoring.



### 3.2 Solution Requirement

Functional requirements include user validation, role checking, deletion restriction, and audit logging. Non-functional requirements include security, reliability, usability, and performance.

### 3.3 Data Flow Diagram



User/Admin sends delete request → ServiceNow backend checks business rule → role validation performed → if unauthorized, deletion blocked → if authorized, action logged and processed.

### 3.4 Technology Stack

Platform: ServiceNow

Backend Logic: Business Rules (Server-side scripting)

Language: JavaScript (ServiceNow scripting)

Tools: ServiceNow Studio, GitHub

Database: ServiceNow internal database



## 4. PROJECT DESIGN

Problem-Solution fit canvas 2.0		Purpose / Vision	
Define CS, Tinto CC	<b>1. CUSTOMER SEGMENT(S)</b> <span>CS</span> Who is your customer? • ServiceNow administrators (to "manage user records and incidents")	<b>6. CUSTOMER CONSTRAINTS</b> <span>CC</span> What constraints prevent your customer from taking action or limit their choice of solutions or, spotting problems ahead? • Authorized roles, ServiceNow platform boundaries, • Limited database access, etc...	<b>5. AVAILABLE SOLUTIONS</b> <span>AS</span> Which solutions are available the customers when they face the problem? (e.g., you have ready vendors, but even their face the problem, etc.) • Existing incident management, or services of external providers, prove to meet, so that you can use any functionality that solutions, of available devices, services, or digital assistants
	<b>2. JOBS-TO-BE-DONE / PROBLEMS</b> <span>J&amp;P</span> What jobs-to-be-done or problems do you address for your customer? • Detecting an issue assigned to active; assigned to active incidents • Validates the missing extensions in the system	<b>9. PROBLEM ROOT CAUSE</b> <span>RC</span> What's the root cause of your problem's existence? • Assigned users cannot be deleted to protect the integrity	<b>7. BEHAVIOUR</b> <span>RC</span> What does your customer do to address the problem and get the job done? I.e., solution or functionality to protect the valuable and costly data, are needed to check incidents manually in the system
Devise CS, Tinto CC, Tinto CC	<b>3. TRIGGERS</b> <span>TR</span> What triggers customer to act? (e.g., detecting an issue assigned to active; assigned to active incidents) • Missing alert for the existing situation in the system • Data anomaly: Failed the unassigned users	<b>10. YOUR SOLUTION</b> <span>SL</span> How does your solution solve customer problems and dissatisfaction? • Prevent deletion of users assigned to active incidents, by assigning assigned roles to users, checking contract to manage records. • Limit deletion to unassigned users	<b>8. CHANNELS &amp; BEHAVIOUR</b> <span>CH</span> 6.1 ONLINE What services can be used to interact with the customer? 6.1 OFFLINE • No other reliable offline channels
	<b>4. EMOTIONS: BEFORE / AFTER</b> <span>EM</span> How do customers feel before they have a problem (or job and attempt)? • Frustrated, worried, confused trying to detect users. • Satisfied after deletion attempt to delete or notified it denied.	<b>10. YOUR SOLUTION</b> <span>SL</span> Prevent deletion of users assigned to protect incidents • Better use of roles, or roles to protect incidents, by assigning assigned roles to users, checking contract to manage records. • Limit deletion attempt: if needed, agent to the required	<b>8. CHANNELS &amp; BEHAVIOUR</b> <span>CH</span> 6.1 ONLINE • ServiceNow admin portal UI 6.3 OFFLINE • No other reliable offline channels
Identify CS, Tinto CC, Tinto CC	<b>4. EMOTIONS: BEFORE / AFTER</b> <span>EM</span> Feeling frustrated, worried, confused trying to delete user and afterward? • Feeling relieved and confident when the problem is solved, or notified it denied, e.g., user is notified in the system		<b>8.2 OFFLINE</b> What is the behavior to get the problem and resolve it immediately in the system

Problem Solution fit canvas 2.0 is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 license  
 Created by Daria Veprikhina / Amaltama.com

## 4.1 Problem Solution Fit

The solution directly addresses the issue of unauthorized user deletion by enforcing strict validation and access control within the ServiceNow environment.

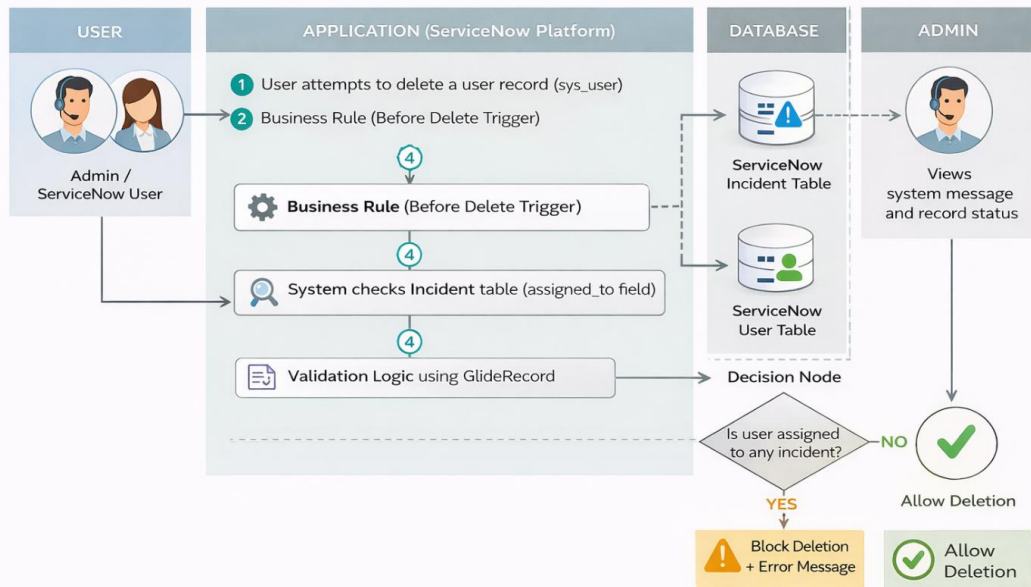
## 4.2 Proposed Solution

The proposed solution is to implement a ServiceNow business rule that prevents deletion of user records unless the user has proper administrative privileges, ensuring data safety and system control.

### 4.3 Solution Architecture

User/Admin Interface → ServiceNow Application Layer → Business Rule (Deletion Validation) → Role & Permission Check → Database (User Table) → Audit Logs.

### Architecture Flow – Prevent User Deletion in ServiceNow using Business Rule



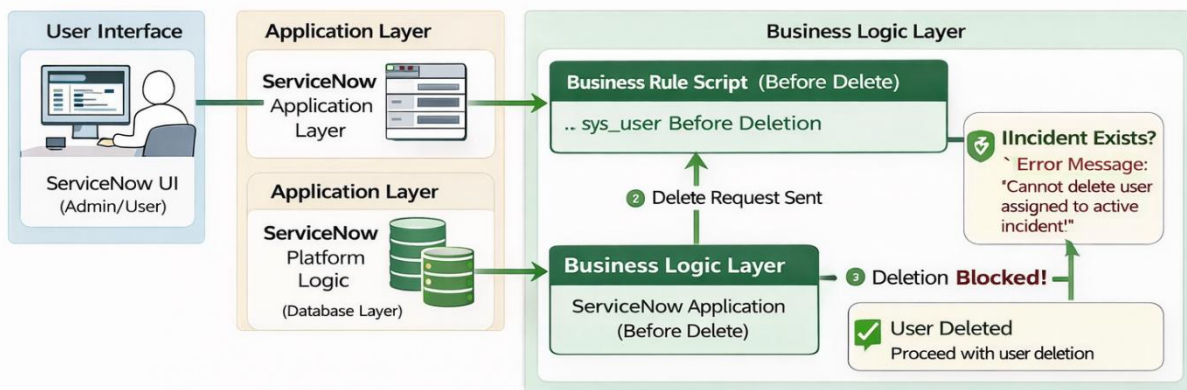
## 5. PROJECT PLANNING & SCHEDULING

### Solution Architecture – Prevent User Deletion in ServiceNow

Date	15 February 2025	Team ID	LTVIP2026TMIDS73881	Project Name	Prevent User Deletion in ServiceNow
Maximum Marks	4 Marks				



**Prevent deletion of users in ServiceNow if they are assigned to active incidents to maintain data integrity and avoid workflow disruption.**



- Admin selects a user record for deletion in ServiceNow UI.
- Delete request is sent to ServiceNow server.
- Business Rule triggers before deletion.
- System checks incident records where the user is assigned.
- If no incidents exist → User deletion permitted.

#### Data Flow

- Admin selects a user record for deletion in ServiceNow UI.
- Delete request is sent to ServiceNow server.
- System checks incident records where the user is assigned.
- If no incidents exist → User deletion permitted.

## 5.1 Project Planning

The project was divided into phases including ideation, requirement analysis, design, development, testing, and documentation. Tasks were allocated among team members and completed using sprintbased planning.

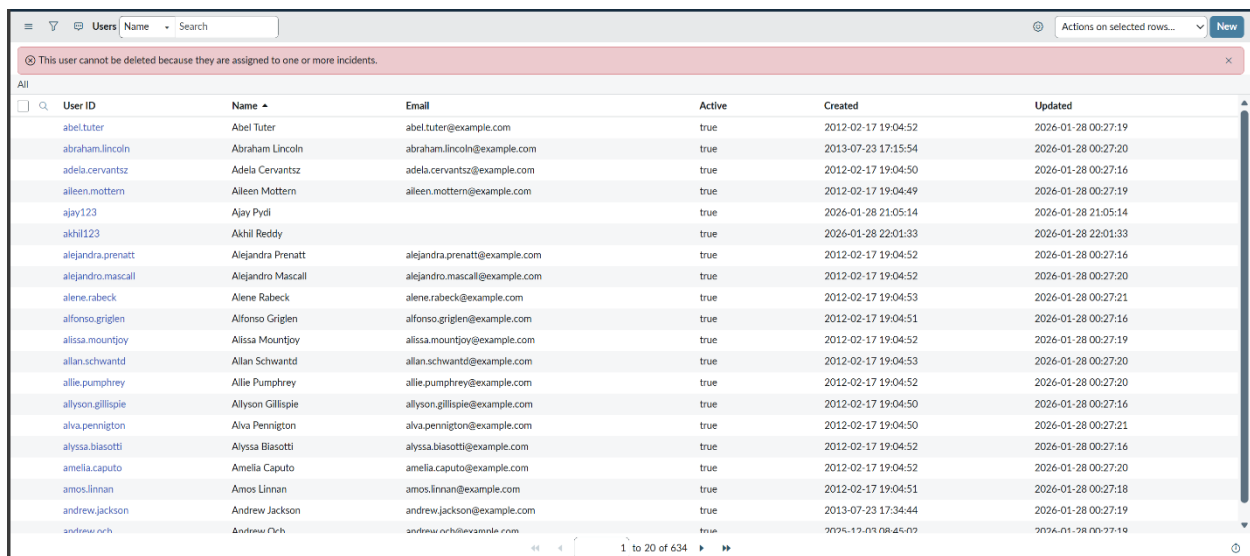
## 6. FUNCTIONAL AND PERFORMANCE TESTING

### 6.1 Performance Testing

The system was tested by attempting multiple user deletion scenarios with different roles. The business rule performed efficiently with instant validation and no noticeable performance delay.

## 7. RESULTS

### 7.1 Output Screenshots



User ID	Name	Email	Active	Created	Updated
abel.tuter	Abel Tuter	abel.tuter@example.com	true	2012-02-17 19:04:52	2026-01-28 00:27:19
abraham.lincoln	Abraham Lincoln	abraham.lincoln@example.com	true	2013-07-23 17:15:54	2026-01-28 00:27:20
adela.cervantsz	Adela Cervantsz	adela.cervantsz@example.com	true	2012-02-17 19:04:50	2026-01-28 00:27:16
alileen.mottern	Alleen Mottern	alileen.mottern@example.com	true	2012-02-17 19:04:49	2026-01-28 00:27:19
ajay123	Ajay Pydi		true	2026-01-28 21:05:14	2026-01-28 21:05:14
akhil123	Akhil Reddy		true	2026-01-28 22:01:33	2026-01-28 22:01:33
alejandra.prenatt	Alejandra Prenatt	alejandra.prenatt@example.com	true	2012-02-17 19:04:52	2026-01-28 00:27:16
alejandro.mascall	Alejandro Mascall	alejandro.mascall@example.com	true	2012-02-17 19:04:52	2026-01-28 00:27:20
alene.rabeck	Alene Rabeck	alene.rabeck@example.com	true	2012-02-17 19:04:53	2026-01-28 00:27:21
alfonso.griglen	Alfonso Griglen	alfonso.griglen@example.com	true	2012-02-17 19:04:51	2026-01-28 00:27:16
alissa.mountjoy	Alissa Mountjoy	alissa.mountjoy@example.com	true	2012-02-17 19:04:52	2026-01-28 00:27:19
allan.schwandt	Allan Schwandt	allan.schwandt@example.com	true	2012-02-17 19:04:53	2026-01-28 00:27:20
allie.pumphrey	Allie Pumphrey	allie.pumphrey@example.com	true	2012-02-17 19:04:52	2026-01-28 00:27:20
allyson.gillispie	Allyson Gillispie	allyson.gillispie@example.com	true	2012-02-17 19:04:50	2026-01-28 00:27:16
alva.pennigton	Alva Pennigton	alva.pennigton@example.com	true	2012-02-17 19:04:50	2026-01-28 00:27:21
alyssa.biasotti	Alyssa Biasotti	alyssa.biasotti@example.com	true	2012-02-17 19:04:52	2026-01-28 00:27:16
amelia.caputo	Amelia Caputo	amelia.caputo@example.com	true	2012-02-17 19:04:52	2026-01-28 00:27:20
amos.linnan	Amos Linnan	amos.linnan@example.com	true	2012-02-17 19:04:51	2026-01-28 00:27:18
andrew.jackson	Andrew Jackson	andrew.jackson@example.com	true	2013-07-23 17:34:44	2026-01-28 00:27:19
andrew.north	Andrew North	andrew.north@example.com	true	2013-07-23 17:34:44	2026-01-28 00:27:19

The system successfully prevented unauthorized user deletion and allowed deletion only for authorized roles. Screenshots demonstrate blocked and allowed deletion cases.

## 8. ADVANTAGES & DISADVANTAGES

Advantages:

Improves data security



Prevents accidental deletion

Enhances system reliability

Easy to implement using ServiceNow business rules

Disadvantages:

Dependent on correct role configuration

Limited to ServiceNow environment

## **9. CONCLUSION**

The Prevent User Deletion in ServiceNow project successfully provides a secure mechanism to protect critical user records by implementing role-based deletion restrictions and backend validation logic.

## **10. FUTURE SCOPE**

Integration with advanced audit dashboards

AI-based anomaly detection for suspicious actions

Multi-level approval workflow before deletion

Extension to other critical tables in ServiceNow

## **11. APPENDIX**

Source Code: ServiceNow Business Rule Scripts

Dataset Link: Not Applicable GitHub & Project Demo Link:

<https://github.com/Nikhilreddy810/Prevent-User-Deletion-ServiceNow>