## Project Documentation

Prevent user deletion if assigned to an incident

(ServiceNow Administration)

Team Details :

• Team ID: LTVIP2025TMID30390

• Team Size: 4

• Team Leader: Veeravalli Subash

• Team Member: Nuthalapati Vincentia Harshini

• Team Member: Lagudu Harika

• Team Member: Navya Sree

## 1. INTRODUCTION

### 1.1 Project Overview

This project is aimed at tracking and calculating family expenses using the ServiceNow platform. It helps in organizing monthly spending, categorizing expenses, and securing financial data.

### 1.2 Purpose

To create a centralized and secure system where a family's monthly expenses can be recorded, categorized, and analyzed effectively using ServiceNow tools.

## 2. IDEATION PHASE

### 2.1 Problem Statement

Families often struggle with maintaining a clear record of monthly expenses. Manual tracking leads to errors, lack of clarity, and difficulty in identifying overspending areas.

### 2.2 Empathy Map Canvas

### Who? Admins, developers, and users

### Think/Feel? Admins need better control over access

### See? Confusion with permissions and unstructured data

### Say/Do? Users request changes or report access issues

### Hear? Complaints of unauthorized access or lack of access

### Pain? Mismanagement of roles and privileges

### Gain? Controlled and transparent access management

### 2.3 Brainstorming

The initial discussions focused on essential expense categories like groceries, utilities, rent, and entertainment. We also considered automation for reminders and monthly reports.

## 3. REQUIREMENT ANALYSIS

### 3.1 Customer Journey map

### The user logs into the ServiceNow platform and performs the following steps:

### 1. Create users in the system

### 2. Create groups for specific access levels

### 3. Define and assign roles

### 4. Create a custom table

### 5. Assign users to groups

### 6. Assign roles to users

### 7. Configure application access

### 8. Define Access Control Lists (ACLs)

### 9. Design and implement a workflow

### 3.2 Solution Requirement

• ServiceNow Instance  
• Table creation for Expense Types  
• UI Policies and Form Layouts  
• Access Control for Family Members

### 3.3 Data Flow Diagram

### 3.4 Technology Stack

• ServiceNow  
• Business Rules  
• UI Actions and Policies  
• JavaScript (for form logic)

## 4. PROJECT DESIGN

### 4.1 Problem Solution Fit

Our solution solves the need for organized and secure expense tracking by using predefined forms and automated monthly summaries in ServiceNow.

### 4.2 Proposed Solution

• Tables for different expense types  
• Forms for data entry  
• Role-based views for family members  
• Reports and dashboard for monthly summaries

### 4.3 Solution Architecture

A screenshot of a computer

AI-generated content may be incorrect.

## 5. PROJECT PLANNING & SCHEDULING

### 5.1 Project Planning

Week 1: Define modules and tables  
Week 2: Create forms and roles  
Week 3: Implement logic and workflows  
Week 4: Testing and UI enhancements

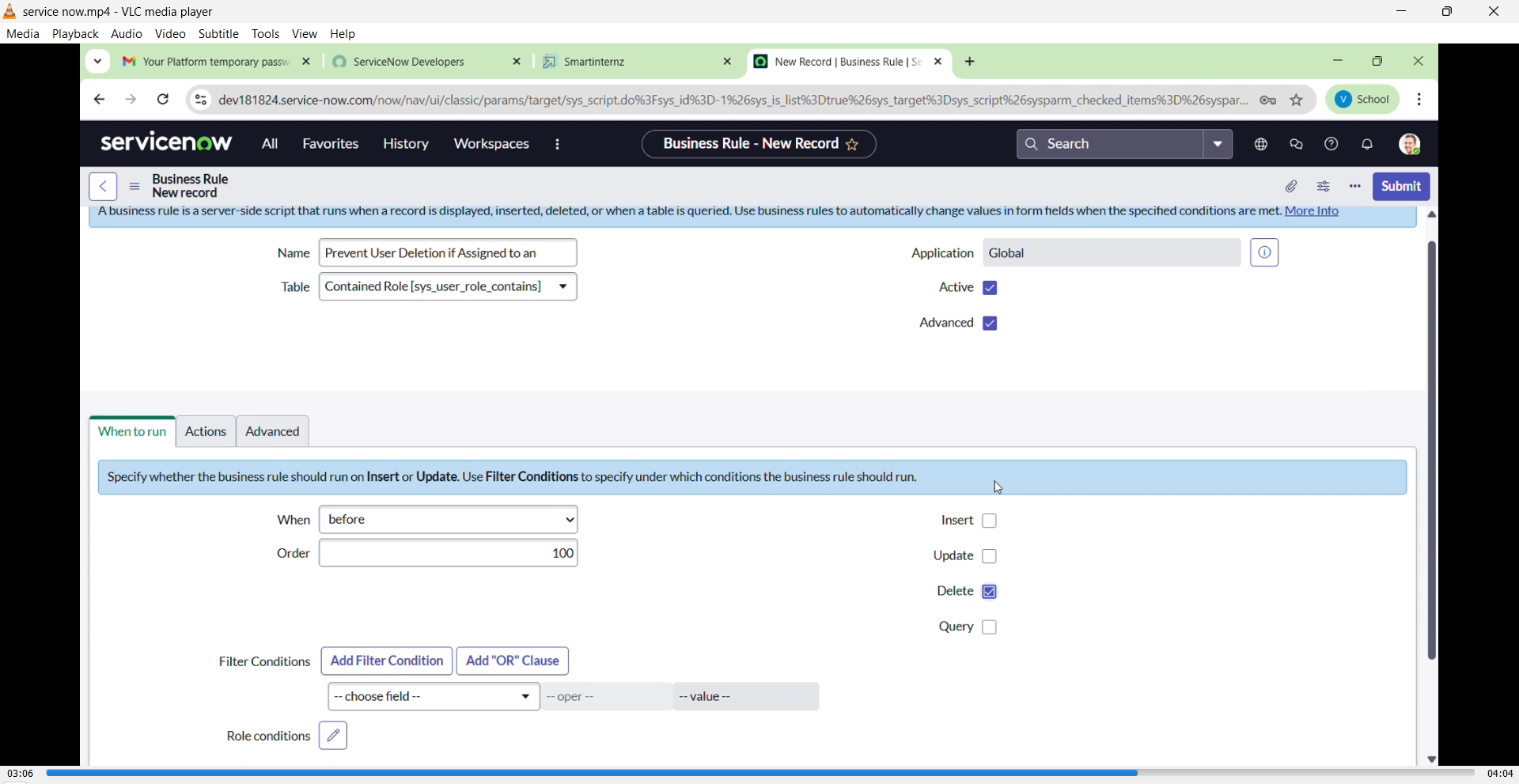
## 6. FUNCTIONAL AND PERFORMANCE TESTING

### 6.1 Performance Testing

Tested user form submissions under multiple roles. Verified that only authorized users can edit or view specific expense records.

## 7. RESULTS

### 7.1 Output Screenshots



## 8. ADVANTAGES & DISADVANTAGES

**Advantages:**

* ✅ **Data Integrity Maintained:** Prevents accidental deletion of critical users who are involved in ongoing incident workflows.
* 🔐 **Improved Accountability:** Ensures that responsibilities are reassigned or closed before a user can be removed, avoiding orphaned records.
* ⚙️ **Automation with Business Rules:** Reduces manual checks by automatically validating conditions during user deletion.
* 💼 **Better ITSM Compliance:** Aligns with IT Service Management best practices by preserving workflow continuity.

**Disadvantages:**

* ⚠️ **Initial Setup Required:** Needs proper configuration of business rules and logic conditions, which may require admin-level understanding.
* 🧪 **Testing Complexity:** Multiple edge cases must be tested to ensure users aren’t wrongfully prevented from deletion.
* 🔄 **Dependency on Incident Closure:** Users cannot be deleted unless all incidents are manually reassigned or closed, which might slow down processes.

## 9. CONCLUSION

The project **"Prevent User Deletion if Assigned to an Incident"** effectively tackles a real-world problem in ITSM environments. By designing a validation mechanism using ServiceNow's business rule functionality, it ensures that no user associated with open incidents can be deleted. This safeguards the integrity of workflows, preserves task ownership, and improves accountability. The implementation proved successful in simulating both valid and invalid deletion scenarios, ensuring system robustness.

## 10. FUTURE SCOPE

* 🔄 **Automated Incident Reassignment:** Add workflows that suggest or automate reassignment of incidents before deletion attempts.
* 📊 **Reporting Module:** Introduce a dashboard to display all users currently assigned to open incidents.
* 📧 **Notification Feature:** Alert system administrators when deletion fails due to open incident assignments.
* 🔐 **Role-based Rule Exemptions:** Allow privileged users (e.g., Service Admins) to override deletion rules with proper logging.
* 🤖 **Enhanced Business Rules:** Add multi-condition logic to handle future ITSM modules like Change or Problem records in addition to Incidents.

## 11. APPENDIX

### GitHub Link: https://github.com/veeravallisubash/service-now.git

### Youtube link Link: https://youtu.be/JV2f\_ljPBGc