IDEATION PHASE

Problem statement:

In an IT Service Management environment, users are frequently assigned to incidents for issue resolution and tracking. However, the current system lacks a validation mechanism to prevent the deletion of a user who is still actively assigned to incidents. This can lead to broken data references, loss of accountability, and disruption in workflow continuity.

There is a need to implement a safeguard that prevents such deletions unless all assigned incidents are closed or reassigned.

Objective:

The primary objective of preventing user deletion when they are assigned to an active incident in ServiceNow is to avoid data loss and maintain data integrity. Deleting a user record that is associated with an incident could lead to orphaned records, inconsistencies, and make it difficult to track the history and resolution of the incident. Instead of deletion, users should be deactivated or locked out to prevent them from logging in, but their records should be retained for historical and auditing purposes.

Challenges:

- Admins must manually identify and reassign incidents before deleting a user.
- In large systems, this process can be time-consuming and error-prone.
- If an employee leaves the organization, and they are still assigned to incidents, their deletion gets delayed.
- Some incidents may be closed, inactive, or forgotten, but still assigned.
- Automated scripts or integrations that delete users may fail unexpectedly if the user is assigned to an incident.
- Managers may not immediately understand why a user cannot be deleted.
- Some organizations want to allow deletion in specific cases (e.g., only for resolved incidents).
- In large-scale implementations, checking incident assignments for every user deletion request could slow down performance or impact database load.