

**Project Design Phase-II**  
**Solution Requirements (Functional & Non-functional)**

Date	14 February 2026
Team ID	LTVIP2026TMIDS26598
Project Name	Prevent User Deletion If Assigned to Incident
Maximum Marks	4 Marks

**Functional Requirements:**

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Deletion Attempt	Admin initiates user deletion from sys_user table System triggers <i>Before Delete Business Rule</i>
FR-2	Incident Validation Check	Query Incident table using GlideRecord Check if user is assigned to any Active incidents
FR-3	Deletion Blocking Logic	If incidents exist → Abort action (setAbortAction(true)) Display error message to administrator
FR-4	Safe Deletion Process	If no active incidents → Allow deletion Maintain data integrity between tables
FR-5	Error Handling & Messaging	Clear, user-friendly error notification Prevent partial or inconsistent data removal

**Non-functional Requirements:**

Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	<b>Usability</b>	Error message should be clear and understandable for administrators
NFR-2	<b>Security</b>	Only authorized admins can delete user records
NFR-3	<b>Reliability</b>	Business Rule must execute consistently before deletion
NFR-4	<b>Performance</b>	Incident validation check should execute quickly without delay
NFR-5	<b>Availability</b>	System must be available during deletion validation process
NFR-6	<b>Scalability</b>	Solution should handle large number of users and incidents