

Project Design Phase

Solution Architecture

Date	Team ID	Project Name	Maximum Marks
02 NOV 2025	NM2025TMID02905	EDUCATIONAL ORGANISATION USING SERVICE NOW	4 Marks

Goals of the Architecture

- Provide a system safeguard using a **Business Rule** in ServiceNow.
- Maintain **data integrity** between courses, students, and trainers.
- Prevent **accidental deletions** of active courses linked to users.
- Reduce **manual monitoring** required for academic data consistency.

Key Components

- **Course Table** – Stores all course records.
- **Student and Trainer Tables** – Contain user records linked to courses through enrollment or assignment relationships.
- **Business Rule (Before Delete)** – Configured on the Course table to monitor deletion actions.
- **Script Logic** – Performs validation to check for active student or trainer associations before permitting course deletion.

Development Phases

1. **Create Test Courses** – Examples include *Python Basics* and *Data Analytics*.
2. **Assign Students and Trainers** – Link active users to the respective test courses.

3. **Implement the Business Rule** – Add the rule on the Course table to handle deletion logic.
 4. **Perform Deletion Tests** – Attempt to delete both assigned and unassigned courses to validate functionality.
-

Solution Architecture Description

The **solution architecture** is designed to maintain **data integrity** and **prevent accidental loss** of academic information within the ServiceNow platform.

It achieves this by implementing a “**Before Delete**” **Business Rule** on the **Course table**, ensuring that no course can be deleted while it is still associated with active students or trainers.

This architecture ensures **consistency between the Course, Student, and Trainer tables** by automatically checking for existing dependencies. If active associations or enrollments are detected, the deletion process is **blocked**, and a **notification message** is displayed to the user, explaining the reason for the restriction.

Development Process Includes:

- Creating sample course and user records.
 - Establishing relationships between courses, students, and trainers.
 - Applying the business rule logic for validation.
 - Testing deletion attempts on both dependent (assigned) and independent (unassigned) courses.
-

Benefits and Outcomes

This architecture:

- **Minimizes human error** by automating dependency checks.
- **Enhances system reliability** through rule-based control.
- **Upholds academic accountability** by safeguarding critical learning data.

By leveraging **native ServiceNow capabilities**, the design remains **efficient**,

scalable, and easy to maintain, ensuring a **secure and automated approach** to academic data management and system operations.