

ServiceNow Smartinternz Project Report

Project Title: Educational Organisation using ServiceNow

Team Id: LTVIP2025TMID60763

Team Size:4

Team Leader: Akkarapaku Harshavardhan

Team Member: Ampolu Sudheer kumar

Team Member: Arava Vara Lakshmi

Team Member: Asma Shaik

ABSTARCT

- 1 .Introduction
- 2.Ideation Phase
- 3.Requirement Analysis
- 4.Project Design Phase
- 5.Project Development
- 6.Result
 - Final Result Screenshots
 - Project Execution Video Recordings
 - Final Conclusion

1 .Introduction

Project Overview:

In today's fast-paced digital world, educational institutions are expected to manage a wide range of activities efficiently—from student admissions and attendance to faculty coordination and examination planning. Many colleges and schools still rely on manual or outdated systems, which leads to delays, errors, and poor communication.

To overcome these challenges, our team is developing a project titled "Educational Organization Using ServiceNow", which leverages the powerful capabilities of the ServiceNow platform. ServiceNow is widely used in the corporate world for streamlining workflows, and we are applying its features to an educational setting.

This project focuses on building a centralized platform that automates key administrative tasks within an educational institution. It aims to simplify and digitize processes such as student registrations, attendance tracking, faculty management, scheduling, and internal communication. By using forms, workflows, and ServiceNow's intuitive interface, the system enhances transparency, improves accuracy, and saves time for both staff and students.

Purpose of the Project

The main purpose of this project is to improve the efficiency and effectiveness of administrative processes in educational institutions by using the ServiceNow platform.

Our goal is to provide a digital solution that:

- Reduces the manual workload for faculty and staff
- Makes student services more accessible and faster
- Ensures data is properly stored, organized, and retrievable
- Enhances the overall communication and coordination between departments

By implementing this system, educational institutions can focus more on delivering quality education, while routine tasks like attendance, event planning, and data management are handled smoothly by the platform. Ultimately, this project is a step toward modernizing education management using reliable technology.

2.Ideation Phase

Problem Statement

Managing a large educational institution comes with many challenges, especially when most of the tasks are handled manually. Students, teachers, and administrators face delays when applying for certificates, submitting requests, or reporting issues. There's often confusion about whom to contact and how long a process will take.

Our goal is to fix these problems by using ServiceNow to create a centralized Educational Management System. This platform will handle everything from student admissions and faculty requests to progress tracking and IT support. It will make the entire process smoother, faster, and more user-friendly.

Goals and Objectives:

- Create one platform for students, teachers, and staff to manage all services
- Simplify and speed up the admission process
- Enable users to easily track the status of their requests
- Automate approval processes using workflows
- Improve transparency and communication across departments
- Reduce manual effort and time wasted on repetitive tasks

Target Users

1. Students:
 - Apply for certificates, admission, or raise complaints
 - Want a system that is quick, clear, and trackable
2. Teachers/Faculty:
 - Submit resource or maintenance requests
 - Need timely approvals and updates on their issues

3. Admin Staff:

- Manage incoming requests efficiently
- Assign tasks and monitor progress in real-time

Key Ideas from Our Team

Feature	Description
Student Service Portal	A portal for students to apply for certificates, hostels, and more
Faculty Request System	Teachers can submit resource, leave, or issue-related requests
Admission Module	Simplify the new student admission and verification process
Request Tracker	Users get live updates on the progress of their applications
Admin Dashboard	Staff can view, assign, and complete tasks from one place

ServiceNow Tools We Plan to Use

- **Service Catalog:** To display services for students and staff
- **Flow Designer:** To automate request approval and assignment
- **Knowledge Base:** To answer frequently asked questions
- **Reports & Dashboards:** For real-time tracking and insights

The Ideation Phase gave us a clear picture of the problems faced by educational institutions. With this understanding, we planned a solution using ServiceNow to make all essential services digital and accessible from one place. This system will reduce time, manual errors, and make the lives of students, faculty, and staff much easier.

3 .Requirement Analysis

Objective

The goal of the requirement analysis phase is to clearly understand what our users (students, teachers, and admin staff) need from the Educational Management System. This system is meant to simplify day-to-day processes like admissions, managing student and teacher data, handling requests, and tracking progress.

By collecting and analyzing requirements, we ensure that the solution we build in ServiceNow meets real-world needs and improves institutional efficiency.

Methods Used for Requirement Gathering

Team Brainstorming – Discussed possible features and problems with teammates

User Feedback – Noted common complaints and requests from students and staff

Process Observation – Looked at how tasks like admissions and support are done manually in colleges

Functional Requirements

These are the main features the system must provide:

1. Student Services Module

- Submit requests for certificates, hostel, and complaints

View request status in real-time

Access academic records and track attendance

2. Teacher/Faculty Module

Manage attendance and upload grades

Request classroom resources or report issues

Apply for leave or teaching duty changes

3. Admission Management

Online student registration and document upload

Automated verification and approval process

Admission status tracking for students

Non-Functional Requirements

These define how the system should behave, not what it does.

Usability: Simple, clean UI that's easy to use for students and staff

Reliability: Should work 24x7 without unexpected breakdowns

Performance: Fast load times, even when many users access it

Scalability: Must support future expansion for new features or more users

Security: Protect sensitive user data and only allow authorized access

Accessibility: Works on mobile phones, tablets, and desktops

These are the people who are either using or managing the system:

- Students
- Faculty Members
- Administrative and IT Staff
- Project Development Team

At the end of this phase, we have a complete understanding

This Requirement Analysis Phase helped us convert user problems and expectations into structured system features. We now have a strong foundation for the system we'll be designing in ServiceNow. With these clearly defined requirements, our next step is to begin the Design Phase and map out the workflows, forms, and automation.

4. Project Design Phase

In this project, we are building a digital solution for managing key activities in an educational institution using the ServiceNow platform. The goal is to create a user-friendly system where staff can handle student admissions, maintain academic records, and manage important data efficiently.

This Design Phase helped us shape the structure of our system through the use of tables, forms, and simple user flows. We focused on creating a practical and clear design that can be easily used by non-technical users like teachers, administrators, and office staff.

Design Overview

The system is designed using the features of the ServiceNow platform, especially focusing on custom tables and forms.

The project contains three main modules:

1. Salesforce Table (Basic Student Info)
2. Admission Table (Full Admission Details)
3. Student Progress Table (Academic Performance Tracking)

Each of these modules is built with its own form and table to organize and store important information.

1. Salesforce Table

The Salesforce Table is created to store basic student data that might later be used for reporting or sharing with other platforms like Salesforce CRM.

► Fields included:

- Admission Number (Auto-generated)
- Admin Date
- Student Name
- Father Name
- Mother Name

- Father Cell Number
- Mother Cell Number
- Grade

This form focuses on basic student identity and contact details.

The form is designed with a clean layout, placing identity fields on the left and contact details on the right. It is useful for syncing or exporting data.

2. Admission Table

The Admission Table stores full details about a student when they join the institution. This is the main form used by admission staff.

► Fields included:

- Admission Number (Auto-generated Unique ID)
- Student Name
- Father Name / Mother Name
- Admin Date
- Grade and Fee
- Purpose of Join
- Admission Status
- Father Cell / Mother Cell
- House No., Area, Mandal
- District, City, Pincode
- School Area and School
- Comments (for extra notes)

This form ensures that all essential information is collected at the time of admission. With dropdowns and reference fields, we reduce errors during data entry.

The form is arranged in two columns for clarity and faster input.

Important fields like student name and contact information are placed at the top. Additional details like address and comments are added in lower sections.

3. Student Progress Table

The Student Progress Table is used to record students' marks in various subjects and automatically calculate results.

► **Fields included:**

- Admission Number (linked to Admission Table)
- Marks for Hindi, English, Telugu, Maths, Social
- Scheme
- Total
- Average
- Percentage
- Result (Pass/Fail)

Teachers can enter the student's marks, and the system calculates the rest. This saves time and avoids manual calculation errors.

The form begins with the Admission Number to identify the student. Marks are filled in next, and calculated fields like total, average, and percentage are automatically shown (non-editable). This ensures the correctness of the result.

User Flow Summary

1. Admissions Staff fills the Salesforce and Admission forms while enrolling a student.
2. Teachers update the Student Progress form after exams.
3. Admins view and manage all records using list and form views in ServiceNow.
4. All records are linked via the Admission Number, which acts as a unique ID across all tables.

Design Purpose

The design ensures:

- Easy and clean data entry
- Automatic generation of Admission Numbers
- Dropdowns and reference fields for accuracy
- Proper linking of student records across modules
- Readable and well-organized forms for different roles

The design of our Educational Organization project in ServiceNow focuses on simplicity, functionality, and clarity. Each form is created with the user in mind, making it easy for staff to complete tasks efficiently. The use of three separate but connected modules helps in organizing student data effectively. This phase has laid a strong foundation for the next steps of development and implementation.

5. Project Development

After completing the project design phase, we moved into the **Development Phase** of our project. This is where our ideas and designs were brought to life inside the ServiceNow platform. We began creating actual tables, forms, fields, and modules, and ensured that the application worked as intended.

This phase helped us transform a basic concept into a working digital solution for managing educational activities such as student admissions, academic performance tracking, and record storage.

Development Activities

During the development phase, we used the **ServiceNow Developer Instance** to build the application. Our main tasks included creating custom tables, designing forms, adding fields, setting up relationships between tables, and building navigation modules.

Here's a breakdown of the development work we completed:

Salesforce Table Development

- We created a new custom table named **Salesforce**.
- Added fields like:
 - **Admin Number** (Auto-generated ID)
 - **Student Name**
 - **Father Name / Mother Name**
 - **Admin Date**
 - **Grade**
 - **Parent Contact Numbers**
 - *This form stores basic student information and can be used for future integrations or reporting.*

Admission Table Development

- Designed a comprehensive **Admission Table** to handle full student enrollment details.

- Fields added include:
 - **Admission Number** (Auto-generated)
 - **Student Name, Father Name, Mother Name**
 - **Grade, Fee, Admin Date**
 - **Purpose of Join, Admission Status**
 - **Address Fields** (House No., Area, Mandal, City, District, Pincode)
 - **Contact Numbers**
 - **School, School Area**
 - **Comments**
 - *Dropdowns were used for fields like Grade and Admission Status to reduce errors and improve consistency.*

Student Progress Table Development

- Built the **Student Progress Table** to record academic marks and generate results.
- Fields added:
 - **Admission Number** (Reference to Admission Table)
 - **Subject Marks:** Hindi, English, Telugu, Maths, Social
 - **Scheme**
 - **Calculated Fields:** Total, Average, Percentage, Result
 - *We ensured that the calculated fields are read-only and auto-updated based on subject marks. The Admission Number links progress records to the main student profile.*

Module Creation

- Created a main application menu called “**Education Management**”.
- Under it, created modules for:
 - **Salesforce Records**
 - **Admission Management**
 - **Student Progress Tracking**
 - These modules help users quickly navigate to forms and lists related to each section of the application

Form Layout & User Interface

- Designed all forms in a **clean and user-friendly layout**.
- Grouped fields in logical order (e.g., personal details at top, address below).
- Used a **two-column layout** to avoid long scrolling forms.
- Enabled reference fields to connect tables (e.g., linking progress to admission).
- *All forms were tested for correct field alignment, readability, and flow.*

Challenges Faced

- Adjusting form layout to make it clean and user-friendly
- Managing dropdown values and data consistency
- Linking reference fields properly between tables
- Handling automatic field calculations

Each challenge was tackled step-by-step through practice, team discussions, and ServiceNow platform help.

Key Learnings

- Learned how to create custom applications in ServiceNow
- Understood how tables, forms, and reference fields work
- Improved teamwork and communication during development
- Gained experience in UI layout, field types, and data handling

The development phase was the most exciting and important part of our project. We took our ideas and turned them into a working solution using ServiceNow's features. With well-designed tables and user-friendly forms, we now have a system that helps educational institutions manage admissions and student progress smoothly. This phase not only helped us apply technical skills but also improved our ability to work as a team and solve real-world problems through software.

6.Result:

Output Screenshots

- CREATING UPDATE SET

The screenshot shows the 'Update Set - Educational Organisation' form in ServiceNow. The form includes fields for Name, State, Parent, Release date, Install date, Installed from, and Description. It also has a section for Application, Created, Created by, and Merged to. The 'Update' button is visible at the bottom left of the form. Below the form, there are 'Related Links' and a 'Customer Updates' section with tabs for 'Update Set Logs', 'Child Update Sets', and 'Install History'.

Created	Type	View	Target name	Updated by	Remote update set	Action
---------	------	------	-------------	------------	-------------------	--------

- CREATING ADMISSION TABLE:

The screenshot shows the 'Dictionary Entry - Pincode' form in ServiceNow. The form includes fields for Table, Type, Column label, and Column name. It also has a section for Application, Active, Function field, Read only, Mandatory, and Display. The 'Choice List Specification' section is expanded, showing a 'Choice' dropdown with 'Dropdown with -- None --'. The 'Related Links' section includes 'Show Table', 'Run Point Scan', and 'Advanced view'.

dev314430.service-now.com/now/ui/classic/params/target/sys_admission_table.do%3Fsys_id%3D-1

ervicenow All Favorites History Workspaces Admin Admission Table - New Record

Tables

- System Archiving
- Archive Tables
 - Archive Knowledge Use
 - Archive Audit Result
- System Clone
 - Clone Definition
 - Exclude Tables
- System Definition
 - Tables
 - Tables & Columns
 - Table Rotations
 - Decision Tables
 - Remote Tables
 - Tables
 - Definitions
 - System Diagnostics
 - Stats
 - Table ICSStats
 - Session Debug
 - Debug SQL (Large Tables)

Admin Status: --None-- House No: District: Area: City: Comments: Fee: \$ 0.00 Mandal: School Area: --None-- Purpose of join: --None-- Pincodes: --None-- School: --None--

Submit

6:32 dev314430.service-now.com/now/ui/classic/params/target/sys_db_object.do%3Fsys_id%3D8fa406c3d8a650c3fb33e40131b3%26sysparm_domain%3Dnu%26sysparm_scope%3Dnu%26sys...

ervicenow All Favorites History Workspaces Admin Table - Admission Table

Table Columns for text Search

Dictionary Entries

Column label	Type	Reference	Max length	Default value	Display
District	String	(empty)	40	false	
City	String	(empty)	40	false	
School Area	Choice	(empty)	40	false	
Pincodes	Choice	(empty)	40	false	
House No	String	(empty)	40	false	
Mandal	String	(empty)	40	false	
Area	String	(empty)	40	false	
Purpose of join	Choice	(empty)	40	false	
Admin Status	Choice	(empty)	40	false	
Comments	String (Full UTF-8)	(empty)	255	false	
Sys ID	Sys ID (GUID)	(empty)	32	false	
School	Choice	(empty)	40	false	
Fee	Price	(empty)	20	false	
Mother Name	String	(empty)	40	false	
Updated by	String	(empty)	40	false	
Updates	Integer	(empty)	40	false	
Grade	Choice	(empty)	40	false	
Class	System Class Name	(empty)	80	javascript:current.getTableNames();	false
Updated	Date/Time	(empty)	40	false	

- CREATING SALESFORCE TABLE:

ServiceNow recommends creating custom tables in scoped applications. To learn more about creating scoped applications, click [here](#).

A table is a collection of records in the database. Each record corresponds to a row in a table, and each field on a record corresponds to a column on that table. Applications use tables and records to manage data and processes. [More info](#)

* Label: Student Progress
 * Name: u_student_progress
 Extends table:

Application: Global
 Create module: ☒
 Create mobile module: ☒
 Add module to menu: Salesforce
 Remote Table: ☐

Columns Controls Application Access

Table Columns: for text Search

Column label	Type	Reference	Max length	Default value	Display
Insert a new row...					

Subtask Cancel

Related Links

Column label	Type	Reference	Max length	Default value	Display
Created by	String	(empty)	40		false
Scheme	String	(empty)	40		false
Maths	String	(empty)	40		false
Percentage	String	(empty)	40		false
Hindi	String	(empty)	40		false
Sys ID	Sys ID (GUID)	(empty)	32		false
Created	Date/Time	(empty)	40		false
English	String	(empty)	40		false
Telugu	String	(empty)	40		false
Updated by	String	(empty)	40		false
Total	String	(empty)	40		false
Updates	Integer	(empty)	40		false
Result	String	(empty)	40		false
Social	String	(empty)	40		false
Updated	Date/Time	(empty)	40		false

Delete Update Delete All Records

Related Links

- .LAYOUT FORM:

The screenshot shows the ServiceNow interface for the 'Table - Student Progress'. The left sidebar contains a navigation menu with options like 'Protected Tables', 'Log Protection', 'Protected Table Log', 'System Archiving', 'Archive Tables', 'System Clone', 'Clone Definition', 'Exclude Tables', 'System Definition', 'Tables', 'Tables & Columns', 'Table Rotations', 'Decision Tables', 'Remote Tables', 'Tables', 'Definitions', 'System Diagnostics', and 'Stats'. The main area displays the table structure for 'Student Progress' with columns: Updates (Integer, empty, 40, false), Result (String, empty, 40, false), Social (String, empty, 40, false), and Updated (Date/Time, empty, 40, false). Below the table structure, there are buttons for 'Delete', 'Update', and 'Delete All Records'. A 'Related Links' section includes links for 'Form Builder', 'Design Form', 'Layout Form', 'Layout List', 'Show List', 'Show Schema Map', 'Add to Service Catalog', 'Run Point Scan', and 'Explore REST API'. At the bottom, there is a table for 'Access Controls' with columns: Name, Decision Type, Operation, Type, Active, Updated by, and Updated. The table shows two entries for 'u_student_progress' with 'Allow If' decision type and 'read' or 'write' operations.

The screenshot shows the 'Configuring Table form' interface in ServiceNow. The left sidebar is the same as the previous screenshot. The main area is divided into two sections: 'Available' and 'Selected'. The 'Available' section lists various fields like 'Created', 'Created by', 'Updated', 'Updated by', 'Updates', '[- begin_split -]', '[- split -]', '[- and_split -]', '* Annotation', '* Chart', 'Activities (filtered)', 'Contextual Search Results', 'Ratings', and 'Attachments'. The 'Selected' section lists fields like 'Admission Number', 'Hindi', 'English', 'Telugu', 'Maths', 'Scheme', '[- split -]', 'Percentage', 'Social', 'Total', 'Average', and 'Result'. Below these sections are buttons for 'Cancel' and 'Save'. The 'Form view and section' section shows 'View name' as 'Default view' and 'Section' as 'Student Progress'. The 'Create new field' section has fields for 'Name', 'Type' (String), and 'Field length' (Small (40)), with an 'Add' button. At the bottom, there is a 'Related Links' section with a link for 'Show versions'.

- FORM DESIGN:

The screenshot displays the ServiceNow 'Form Design' tool for the 'Student Progress' table. On the left, there is a sidebar with 'Fields' and 'Field Types' tabs. The 'Fields' tab is active, showing a list of fields including 'Created', 'Created by', 'Updated', 'Updated by', 'Updates', 'Activities (filtered)', 'Contextual Search Results', and 'Ratings'. The main area is titled 'Form Design' and shows a table of field configurations for the 'Student Progress [s_student_progress]' table. The table has 2 columns and lists various fields with their respective settings.

Student Progress [s_student_progress]		2 Columns
Admission Number		
Hindi		
English		
Telugu		
Maths		
Scheme		
Percentage		
Social		
Total		
Average		
Result		

- CREATING NUMBER MAINTENANCE

ServiceNow

number mainta

Number - New Record

Table: Salesforce

Prefix: SAL

Number: 1,000

Application: Global

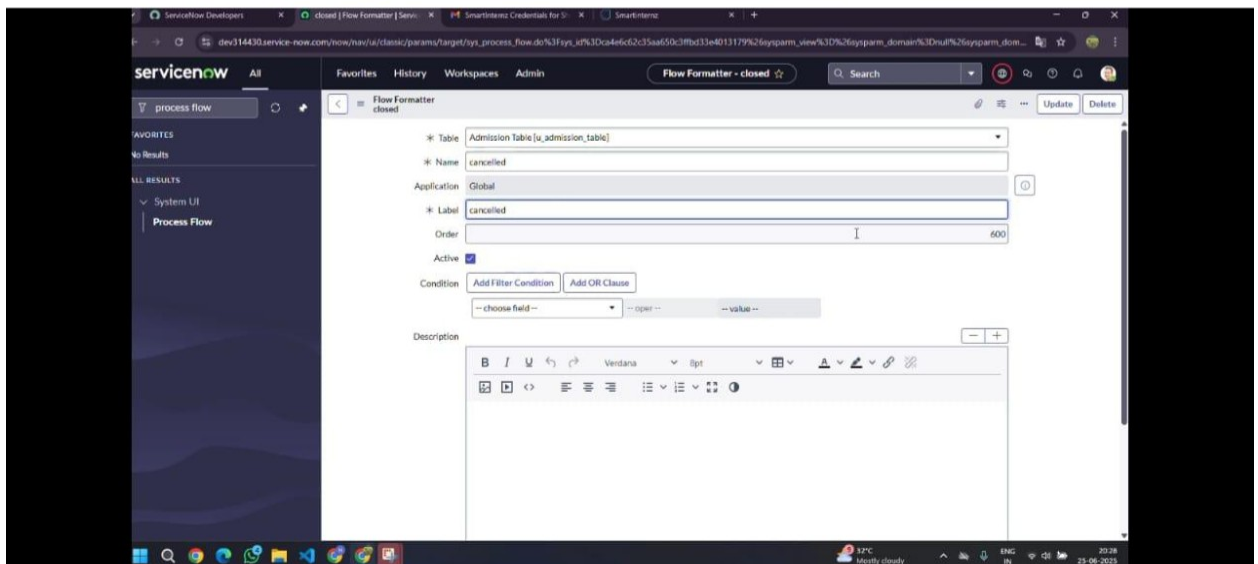
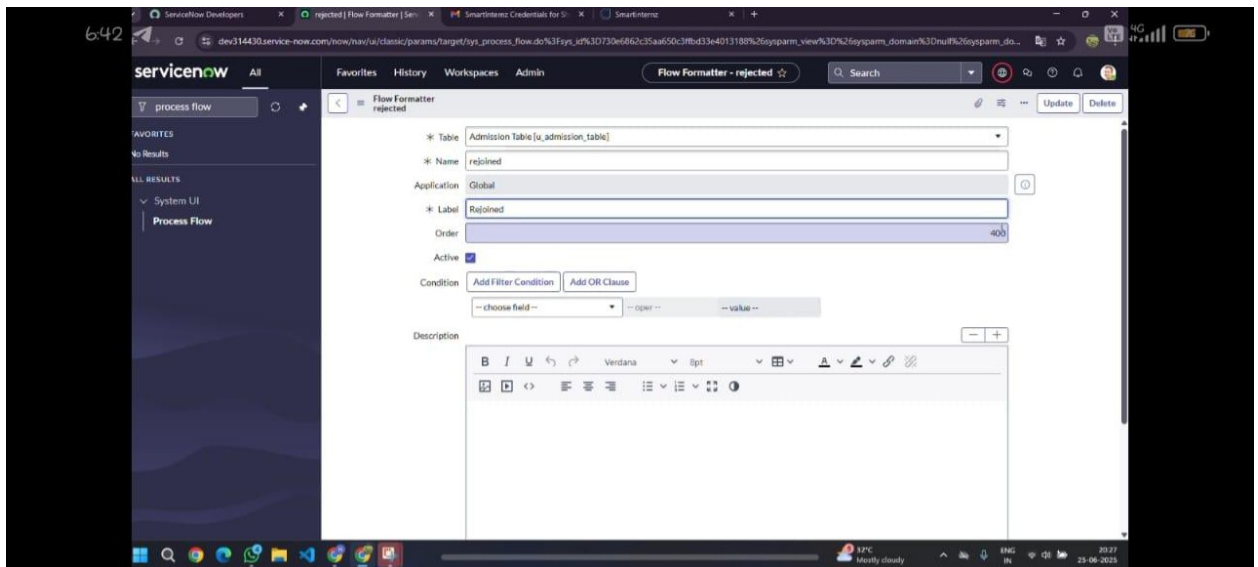
Number of digits: 7

Submit

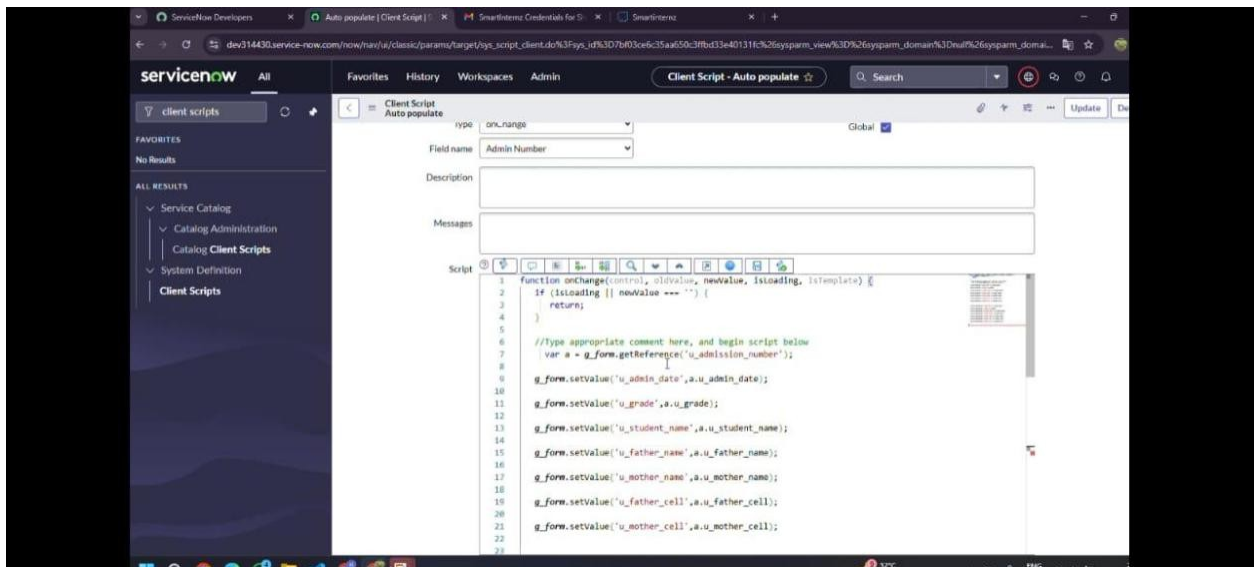
Related Links

Show Counter

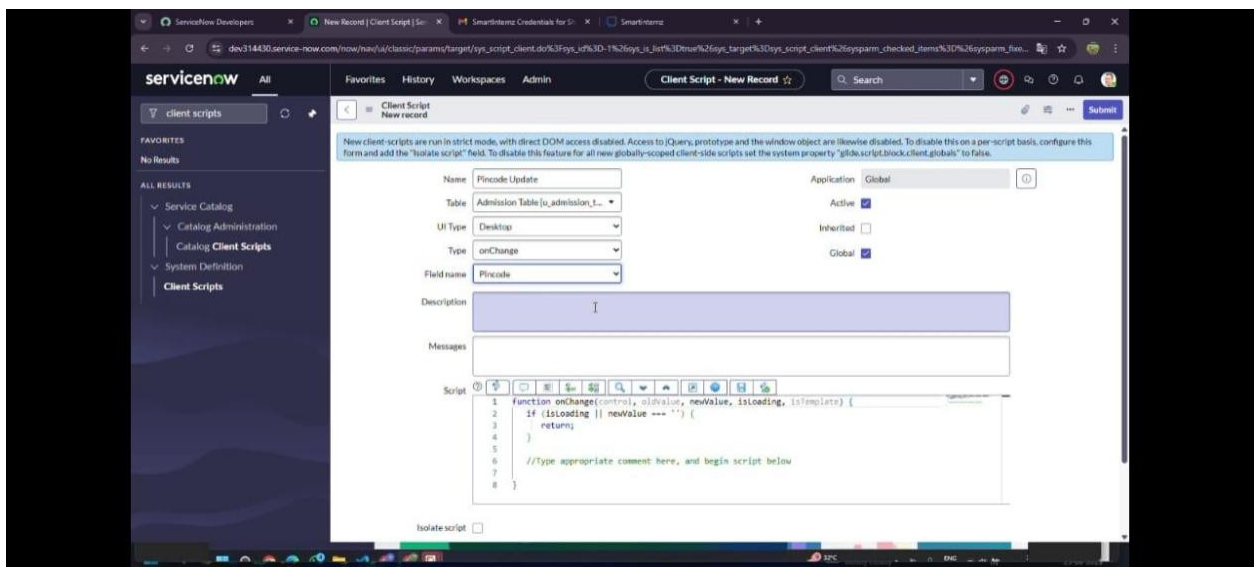
- PROCESS FLOW:

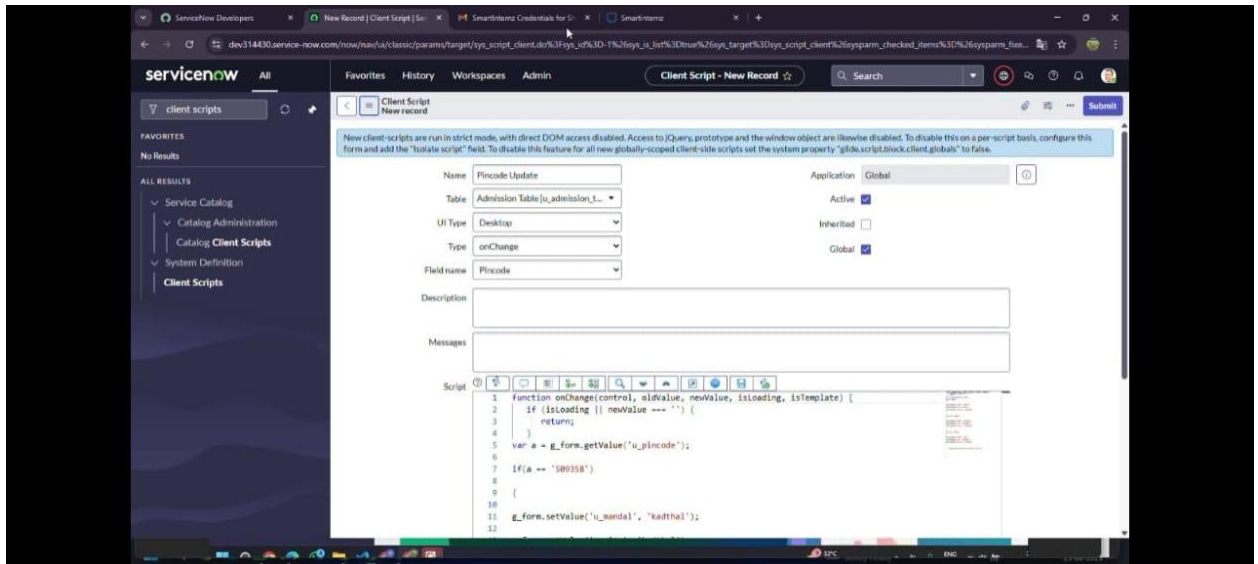


- CLIENT SCRIPT (AUTO POPULATE)

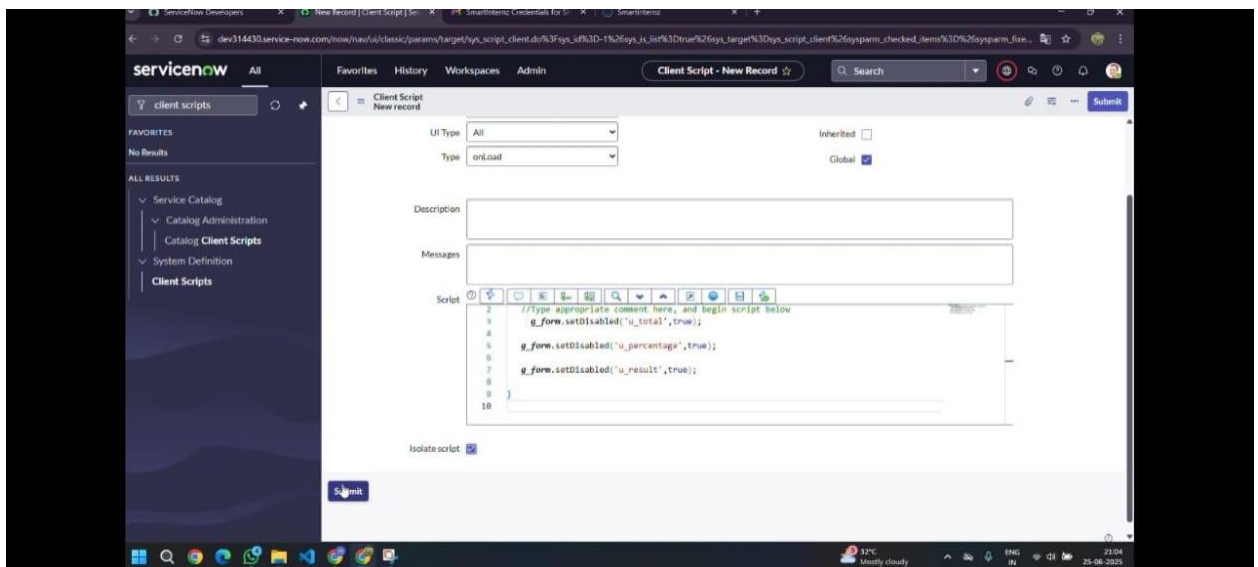


- CLIENT SCRIPT (PINCODE UPDATE)





- CLIENT SCRIPT (DISABLE FIELDS)



CLIENT SCRIPT (TOTAL UPDATE)

The screenshot shows the 'Client Script - New Record' form in ServiceNow. The left sidebar contains a navigation menu with 'client scripts' selected. The main form area has a title bar with 'Client Script - New Record' and a 'Submit' button. Below the title bar is a warning message about strict mode. The form fields are as follows:

- Name: Total Update
- Table: Student Progress [u_student_prog]
- UI Type: All
- Type: onChange
- Field name: Social
- Application: Global
- Active: ☒
- Inherited: ☐
- Global: ☒
- Description: (empty text area)
- Messages: (empty text area)
- Script: (code editor with the following content)

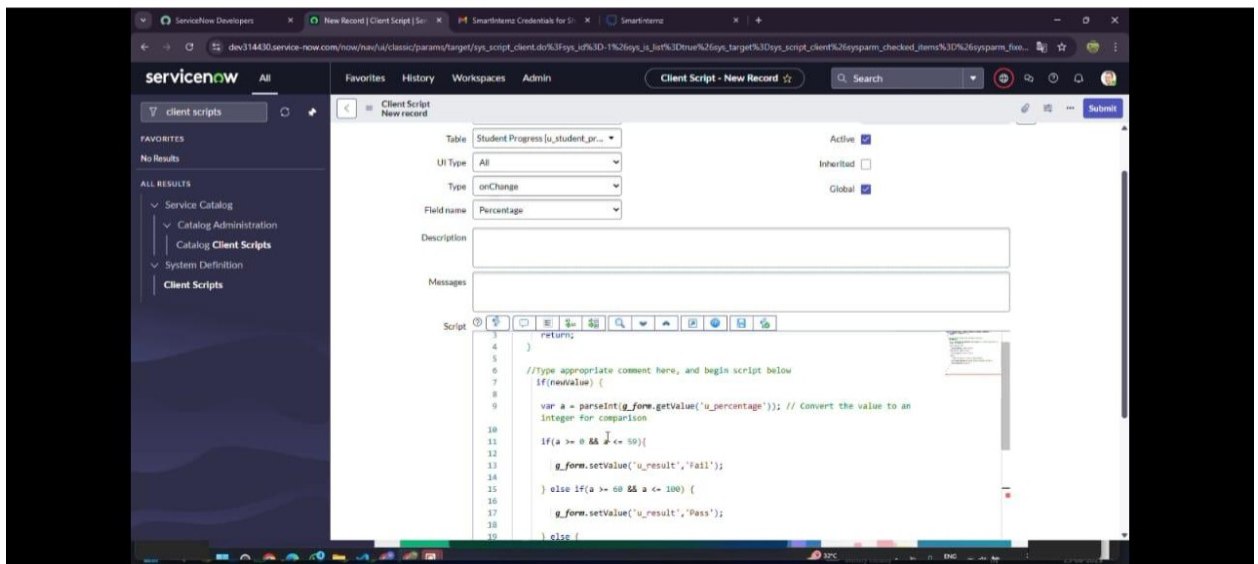
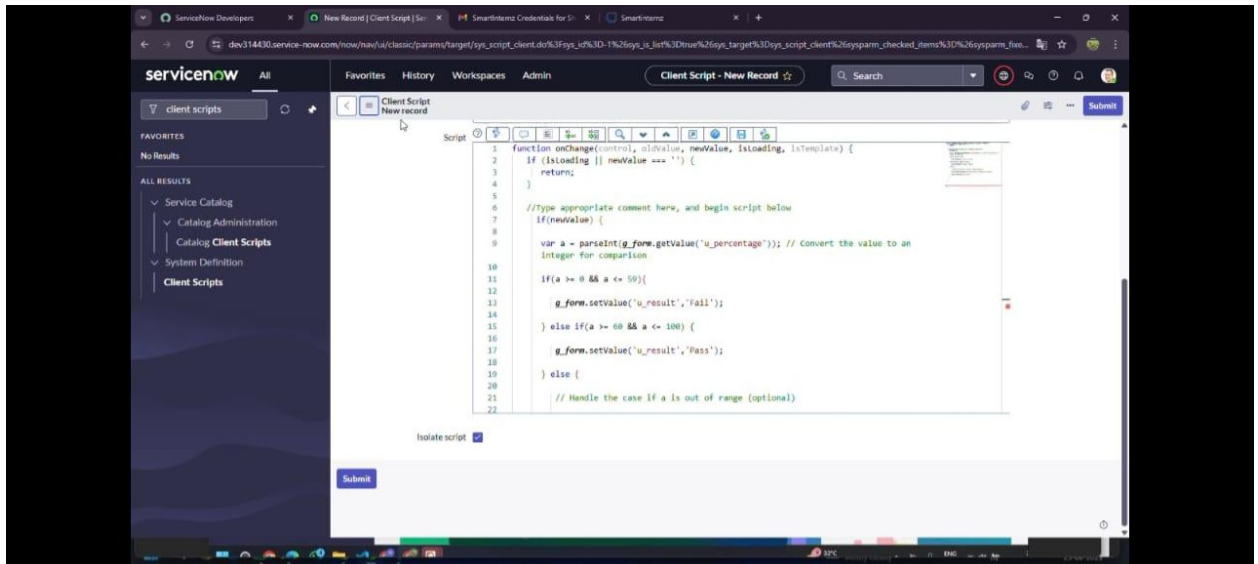
```
1 function onChange(control, oldValue, newValue, isLoading, isTemplate) {  
2   if (!isLoading || newValue === '') {  
3     return;  
4   }  
5  
6   //Type appropriate comment here, and begin script below  
7  
8 }
```
- Isolate script: ☐

The screenshot shows the 'Client Script - Total Update' form in ServiceNow. The left sidebar contains a navigation menu with 'client scripts' selected. The main form area has a title bar with 'Client Script - Total Update' and 'Update' and 'Delete' buttons. Below the title bar is a warning message about strict mode. The form fields are as follows:

- Name: Total Update
- Table: Student Progress [u_student_prog]
- UI Type: All
- Type: onChange
- Field name: (empty dropdown)
- Application: Global
- Active: ☒
- Inherited: ☐
- Global: ☒
- View: (empty dropdown)
- Description: (empty text area)
- Messages: (empty text area)
- Script: (code editor with the following content)

```
function onChange(control, oldValue, newValue, isLoading, isTemplate) {  
  if (!isLoading || newValue === '') {  
    return;  
  }  
  
  //Type appropriate comment here, and begin script below  
  if (newValue) {  
    var a = parseInt(g_form.getValue('u_telugu'));  
  }  
}
```
- Isolate script: ☒

- CLIENT SCRIPT (RESULT):



- CLIENT SCRIPT (PERCENTAGE)

The screenshot shows the 'Client Script - New Record' form in ServiceNow. The form is titled 'Client Script - Percentage'. The 'Name' field is 'Percentage'. The 'Table' is 'Student Progress [u_student_prog]'. The 'UI Type' is 'All'. The 'Type' is 'onChange'. The 'Field name' is empty. The 'Application' is 'Global'. The 'Active' checkbox is checked. The 'Inherited' checkbox is unchecked. The 'Global' checkbox is checked. The 'View' field is empty. The 'Description' field is empty. The 'Messages' field is empty. The 'Script' field contains the following code:

```
function onChange(control, oldValue, newValue, isLoading, isTemplate) {
    if (isLoading || newValue === '') {
        return;
    }

    //Type appropriate comment here, and begin script below
    var Total = g_form.getValue("u_total");
    var Percentage = (Total/400)*100;
}
```

The 'Isolate script' checkbox is checked. The 'Update' and 'Delete' buttons are at the bottom right.

The screenshot shows the 'Client Script - New Record' form in ServiceNow. The form is titled 'Client Script - Percentage'. The 'Name' field is 'Percentage'. The 'Table' is 'Student Progress [u_student_prog]'. The 'UI Type' is 'All'. The 'Type' is 'onChange'. The 'Field name' is 'Total'. The 'Application' is 'Global'. The 'Active' checkbox is checked. The 'Inherited' checkbox is unchecked. The 'Global' checkbox is checked. The 'Description' field is empty. The 'Messages' field is empty. The 'Script' field contains the following code:

```
1 function onChange(control, oldValue, newValue, isLoading, isTemplate) {
2     if (isLoading || newValue === '') {
3         return;
4     }
5
6     //Type appropriate comment here, and begin script below
7
8 }
```

The 'Isolate script' checkbox is unchecked. A warning message is displayed at the top: 'New client-scripts are run in strict mode, with direct DOM access disabled. Access to jQuery, prototype and the window object are likewise disabled. To disable this on a per-script basis, configure this form and add the "isolate script" field. To disable this feature for all new globally-scoped client-side scripts set the system property "globe.script.block.client.globals" to false.'

The 'Submit' button is at the bottom right.

Final Result Screenshots:

This screenshot shows the 'Salesforce - Create SAL0001008' form in ServiceNow. The left sidebar contains a navigation menu with categories like 'System Archiving', 'System Clone', 'System Definition', and 'Tables'. The main form area is titled 'Salesforce New record' and includes a 'Submit' button. The form fields are organized into two columns:

Field	Value
Admin Number	SAL0001008
Admin Date	
Mother Name	
Grade	-- None --
Student Name	
Father Name	
Mother Cell	
Father Cell	

Below the form fields is a section titled 'New Section' with a 'Submit' button.

This screenshot shows the 'Admission Table - Create SAL0001007' form in ServiceNow. The left sidebar contains a navigation menu with categories like 'System Archiving', 'System Clone', 'System Definition', and 'Tables'. The main form area is titled 'Admission Table New record' and includes a 'Submit' button. The form fields are organized into two columns:

Field	Value
Admin Number	SAL0001007
Purpose of join	-- None --
Student Name	
Father Name	
Mother Name	
Admin Date	
Grade	-- None --
Fee	\$ 0.00
Father Cell	
Mother Cell	
Admin Status	-- None --
District	
City	
Comments	
School Area	-- None --
Pincode	-- None --
Mandal	
School	-- None --

Below the form fields is a section titled 'New Section' with a 'Submit' button.

Project Execution Video Recordings:

- Click here to view the complete screen recording of the project:(35 mins video)

<https://drive.google.com/file/d/1JICLP1AiFQnkR1291muzH8FTeKRE2iaC/view?usp=drivesdk>

- Click here to watch the demo video which include only the result of the project:(2 mins video)

https://drive.google.com/file/d/1kXDtG2KuXM_4wvu0nxYsUAz8cB5OcFx0/view?usp=drive_link

Final Conclusion

When we first began this project, we only had a simple idea — to make things easier for educational institutions. There was no complete clarity, no strong roadmap, and even some doubts about whether we could really build something meaningful using the ServiceNow platform. But slowly, as we started planning, designing, developing, and testing, we saw that we were creating something real, something impactful.

Our project, “**Educational Organization using ServiceNow**” has not just been about creating tables and forms. It has been about solving a real problem faced by many educational institutions — the burden of manual work, mismanaged student data, scattered academic records, and outdated admission systems.

Through this project, we developed a complete system where:

- Student admissions are managed smoothly with clean data entry.
- Academic progress is tracked and auto-calculated without errors.
- Essential student records are available in a structured and easy-to-use interface.

More than just a technical exercise, this project taught us:

- How digital systems can solve real-life problems.
- How platforms like ServiceNow, when used properly, can bring transformation even to non-IT sectors like education.

We also learned that building something is not just about reaching the final screen or form — it's about understanding the user's needs and delivering what actually helps them.