



K.S.K COLLEGE OF ENGINEERING AND TECHNOLOGY

**DEPARTMENT OF COMPUTER SCIENCE ENGINEERING
AND TECHNOLOGY**

EDUCATION ORGANIZATION USING SERVICENOW

**TEAM ID : NM2025TMID05621
TEAM SIZE : 4**

TEAM LEADER : PARASURAMAN G - 821022104035

TEAM MEMBER : JAI KUMAR R - 821022104018

TEAM MEMBER : RAKESH S - 821022104041

TEAM MEMBER : SIVAGURU A - 821022104047

PROBLEM STATEMENT:

Educational institutions often struggle with managing large volumes of student and staff information through manual or disconnected systems. Tasks such as admissions, student record management, and progress tracking become timeconsuming and prone to errors. The lack of a unified digital platform leads to inefficiency, communication gaps, and delays in decision-making. This project addresses these challenges by developing an Educational Management System on the ServiceNow platform to automate administrative workflows, centralize data, and improve institutional efficiency.

OBJECTIVES:

To design and implement a centralized system for managing student and teacher information using ServiceNow.

1. To automate key processes such as admissions, attendance, and progress monitoring.
2. To enable real-time access and updates of academic and administrative data.
3. To improve coordination among departments and reduce manual workloads.
4. To generate reports and analytics for better academic and administrative decision-making.
5. To enhance transparency and user experience for students, faculty, and administrators.

SKILLS AND TECHNOLOGIES USED:

Platform:

- ServiceNow

Skills:

- ServiceNow Administration and Configuration
- Form and Table Design
- Workflow and Flow Designer
- Reporting and Dashboard Creation

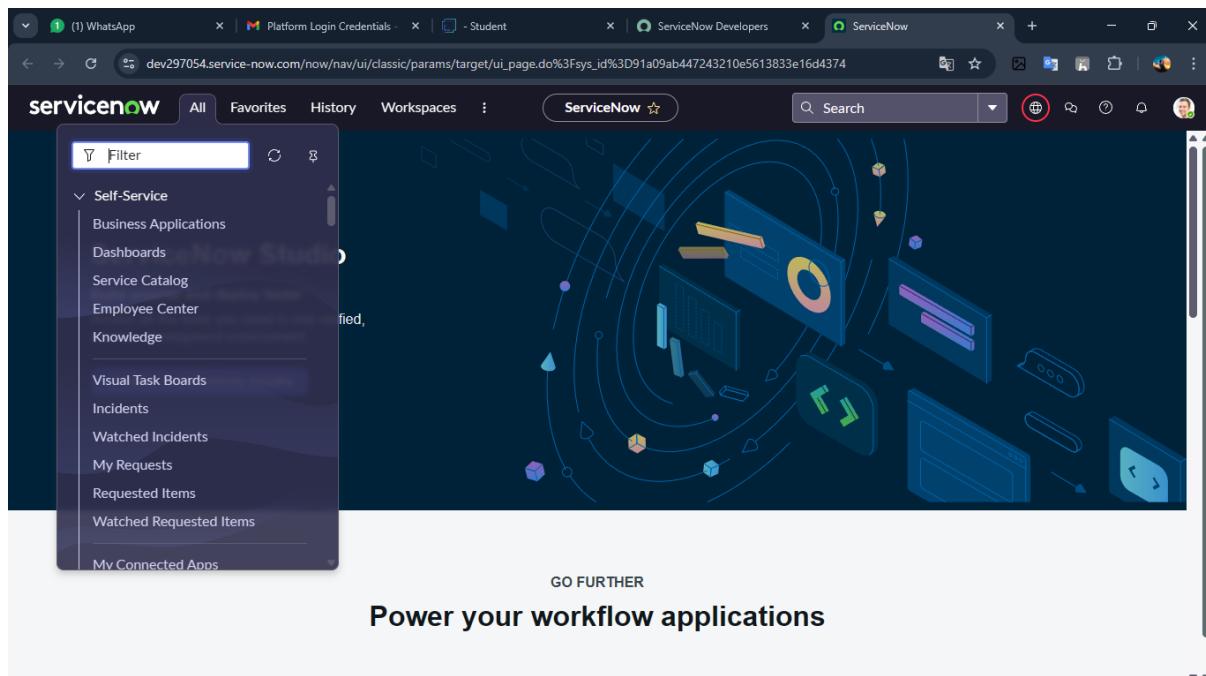
TASK INITIATION:

The Educational Organization Using ServiceNow project begins with the initiation phase, where the project scope, goals, and resources are clearly defined. The purpose of this phase is to establish a clear understanding of what the project aims to achieve and to prepare the foundation for successful implementation.

STEPS:

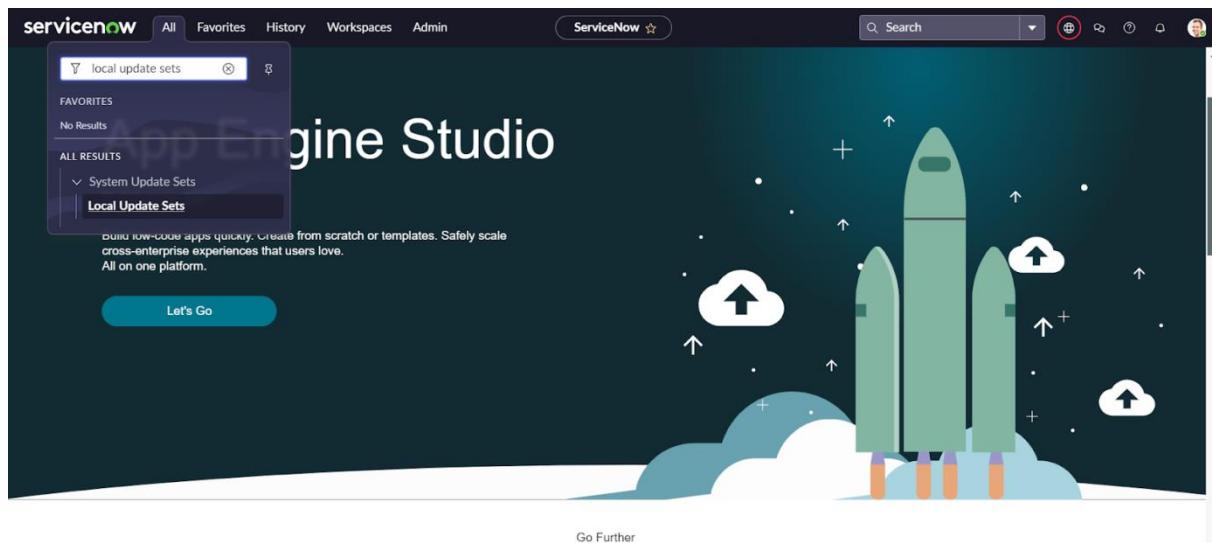
STEP 1: SETTING UP SERVICENOW INSTANCE

- 1.Sign up for a developer account on the ServiceNow Developer site "<https://developer.servicenow.com>".
- 2.Once logged in, navigate to the "Personal Developer Instance" section.
- 3.Click on "Request Instance" to create a new ServiceNow instance.
- 4.Fill out the required information and submit the request.
- 5.You'll receive an email with the instance details once it's ready.
- 6.Log in to your ServiceNow instance using the provided credentials.
- 7.Now you will navigate to the ServiceNow.



STEP 2: CREATING AN UPDATE SET

1.Click on All >> Local update sets .



2.Click on new

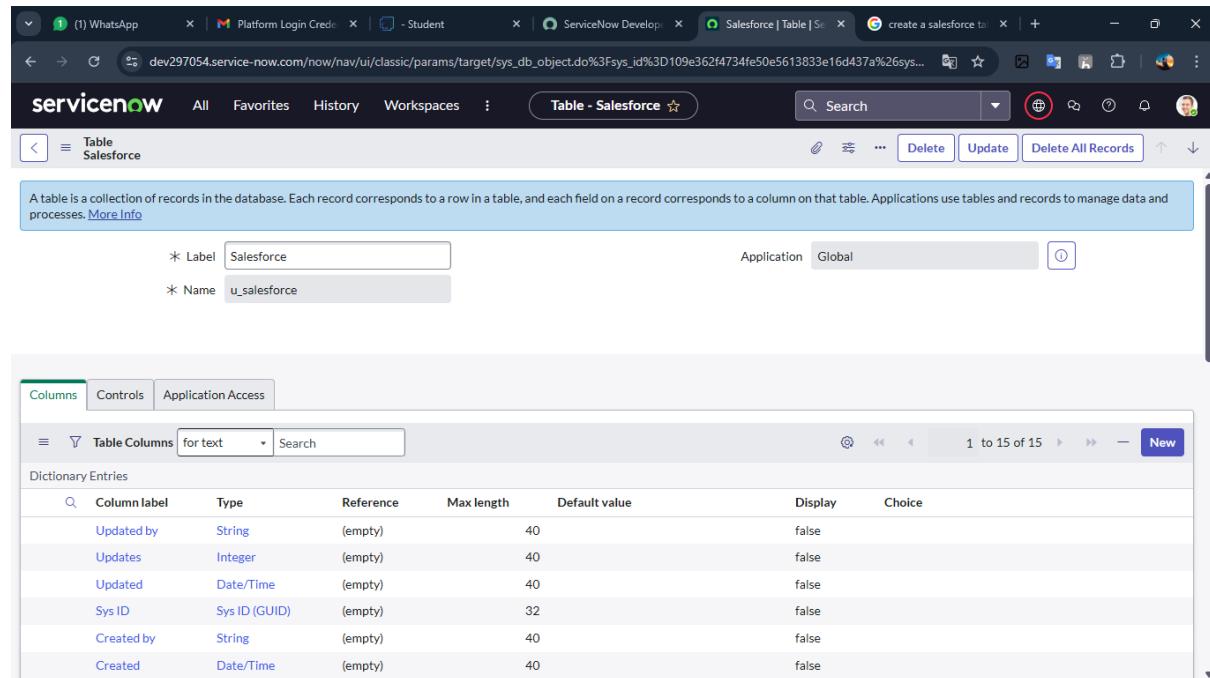
A screenshot of the 'Update Sets' list page in ServiceNow. The top navigation bar includes 'servicenow', 'All', 'Favorites', 'History', 'Workspaces', and 'Admin'. A search bar with the placeholder 'Search' is on the right. The main area shows a table with columns: Name, Application, State, Installed from, Created, Created by, Parent, and Batch Base. Two rows are listed: 'Default' (Application: App Engine Studio, State: In progress) and 'Default' (Application: Global, State: In progress). At the top right of the table, there is a 'New' button with a red arrow pointing to it. Above the table, there is a 'Actions on selected' dropdown menu with a red arrow pointing to it.

3.Enter the Details Name: Educational Organisation >> Click on Submit and make Current.

A screenshot of the 'Update Set' details page for 'Education Organization'. The top navigation bar includes 'servicenow', 'All', 'Favorites', 'History', 'Workspaces', and 'Admin'. A search bar with the placeholder 'Search' is on the right. The main area shows a form with fields: Name (Education Organization), State (In progress), Parent (dropdown with a search icon), Release date (date picker), Install date (date picker), Installed from (text input), and Description (text area). To the right of the form, there are additional details: Application (Global), Created (2025-10-31 03:39:11), Created by (admin), and Merged to (text input). At the top right of the form, there is a 'Submit' button with a red arrow pointing to it.

STEP 3: CREATING A TABLE:

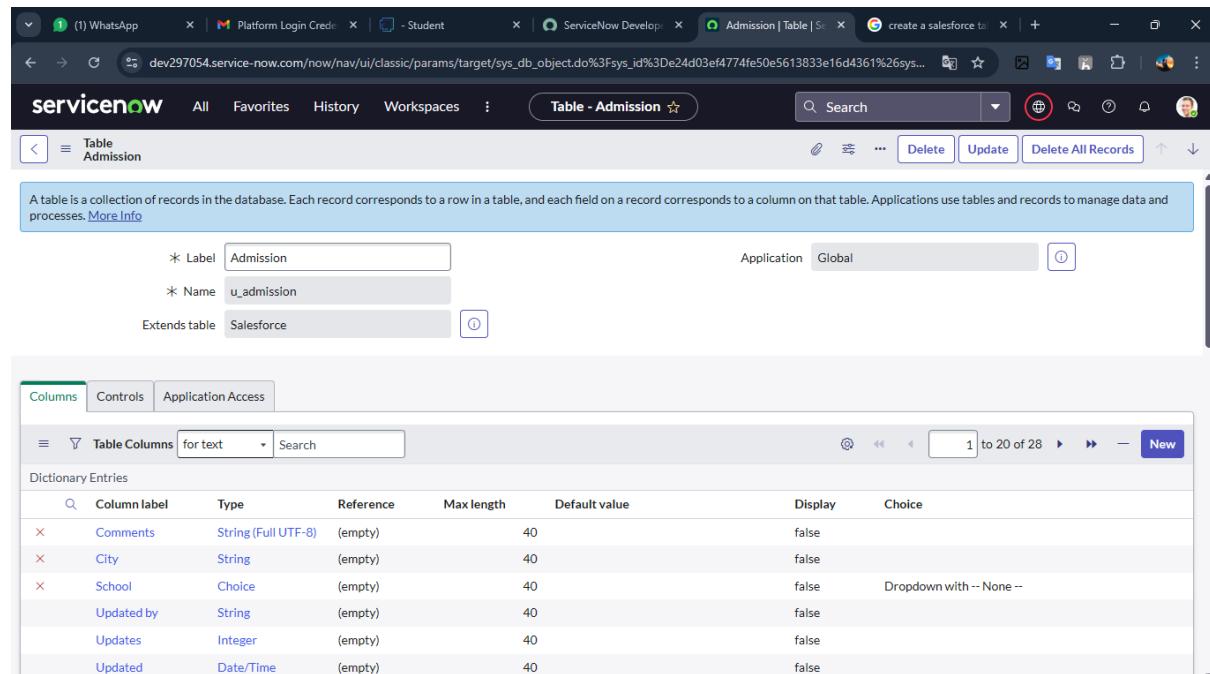
1.Salesforce:



The screenshot shows the ServiceNow Table - Salesforce configuration page. At the top, there are fields for Label ("Salesforce") and Name ("u_salesforce"), both marked with an asterisk. To the right, there is an "Application" dropdown set to "Global". Below this, the "Columns" tab is selected in the navigation bar. The "Table Columns" section lists several system columns: Updated by (String), Updates (Integer), Updated (Date/Time), Sys ID (Sys ID (GUID)), Created by (String), and Created (Date/Time). Each column has its type, reference, max length, default value, display setting (false), and choice (false) listed.

Column label	Type	Reference	Max length	Default value	Display	Choice
Updated by	String	(empty)	40		false	
Updates	Integer	(empty)	40		false	
Updated	Date/Time	(empty)	40		false	
Sys ID	Sys ID (GUID)	(empty)	32		false	
Created by	String	(empty)	40		false	
Created	Date/Time	(empty)	40		false	

2.Admission:



The screenshot shows the ServiceNow Table - Admission configuration page. At the top, there are fields for Label ("Admission") and Name ("u_admission"), both marked with an asterisk. Below this, there is a "Extends table" field set to "Salesforce". The "Columns" tab is selected in the navigation bar. The "Table Columns" section lists several columns: Comments (String (Full UTF-8)), City (String), School (Choice), Updated by (String), Updates (Integer), and Updated (Date/Time). The "School" column is marked with a red "X" and has a note indicating it is a "Dropdown with -- None --".

Column label	Type	Reference	Max length	Default value	Display	Choice
Comments	String (Full UTF-8)	(empty)	40		false	
City	String	(empty)	40		false	
School	Choice	(empty)	40		false	Dropdown with -- None --
Updated by	String	(empty)	40		false	
Updates	Integer	(empty)	40		false	
Updated	Date/Time	(empty)	40		false	

3.Student Progress:

The screenshot shows the ServiceNow Table - Student Progress configuration page. At the top, there are fields for Label (Student Progress), Application (Global), and Name (u_student_progress). Below these, it says "Extends table Salesforce". The main area is titled "Columns" and contains a table of dictionary entries. The columns are: Column label, Type, Reference, Max length, Default value, Display, and Choice. The entries include Telugu (String, empty, 40, false), Updated by (String, empty, 40, false), Updates (Integer, empty, 40, false), Updated (Date/Time, empty, 40, false), Sys ID (Sys ID (GUID), empty, 32, false), and Created by (String, empty, 40, false).

Column label	Type	Reference	Max length	Default value	Display	Choice
Telugu	String	(empty)	40		false	
Updated by	String	(empty)	40		false	
Updates	Integer	(empty)	40		false	
Updated	Date/Time	(empty)	40		false	
Sys ID	Sys ID (GUID)	(empty)	32		false	
Created by	String	(empty)	40		false	

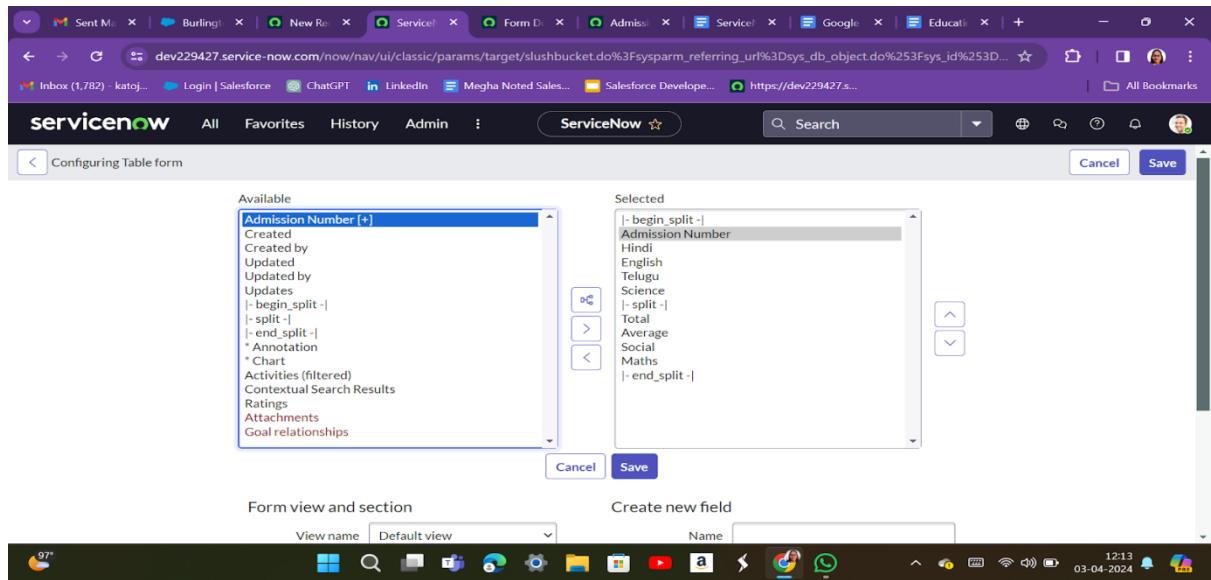
STEP 4: FORM LAYOUT

Configuring Table form for Student Progress Table

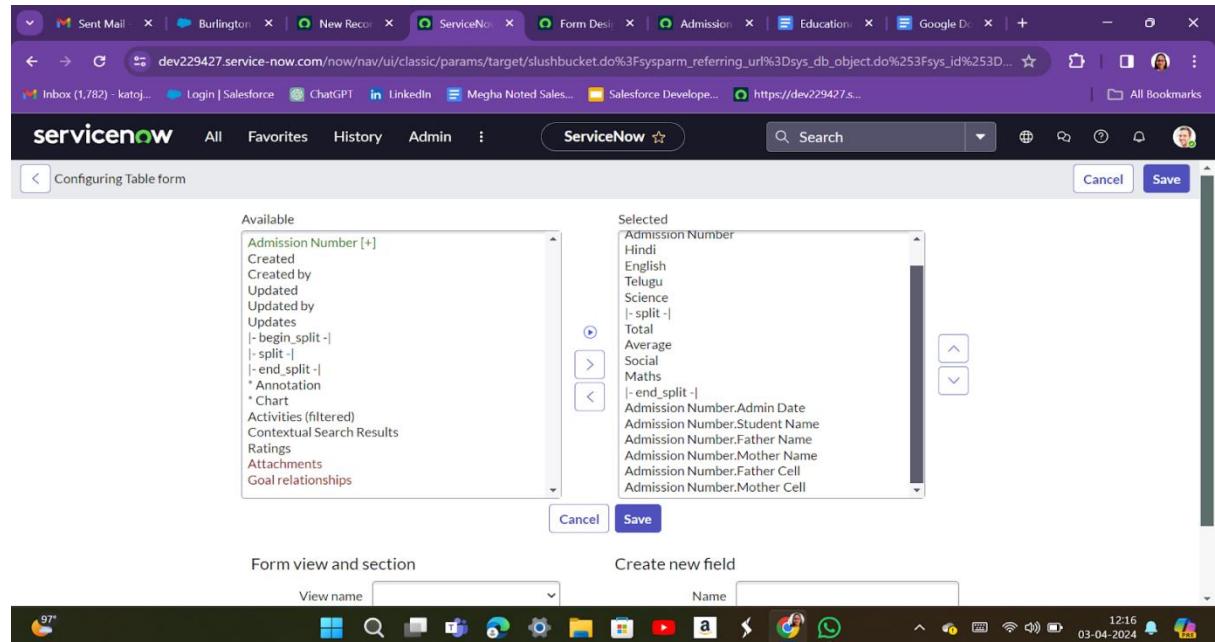
- In the Student Progress Table Page , Click on Layout form

The screenshot shows the layout configuration for the Student Progress table. It displays two fields: Telugu (String, empty, 40) and Total (String, empty, 40). Below the fields are buttons for Update, Delete, and Delete All Records. A "Related Links" section is present at the bottom left, listing options like Design Form, Layout Form (which is highlighted), Layout List, Show Form, Show List, Show Schema Map, Add to Service Catalog, Run Point Scan, and Explore REST API.

- Click on Admission Number [+].



- Select below Admission Number fields in Available side and send it to selected side as below >> save.



STEP 5: FORM DESIGN

CREATING FORM DESIGN FOR SALESFORCE TABLE

1.All >> System Definition >> Tables .

The screenshot shows the ServiceNow Tables View: Tables page. A search bar at the top contains the text 'salesforce'. Below it is a table with columns: Name, Extends table, Extensible, and Updated. The table lists various system tables, including 'sn_sub_man_st_account_level_entitlement' and 'v_st_sc_catalog'. At the bottom of the table, there is a message: 'https://dev297054.service-now.com/sys_db_object_list.do?sysparm_userpref_module=7e7ca89ac0a8000901594ba32f405461&sysparm_query=sys_update_name!SNOTEMPTY%5E&sysparm_clear_stack=true&sysparm_clear_stack=true'. A red circle highlights the 'Actions on selected rows...' button in the top right corner of the table header.

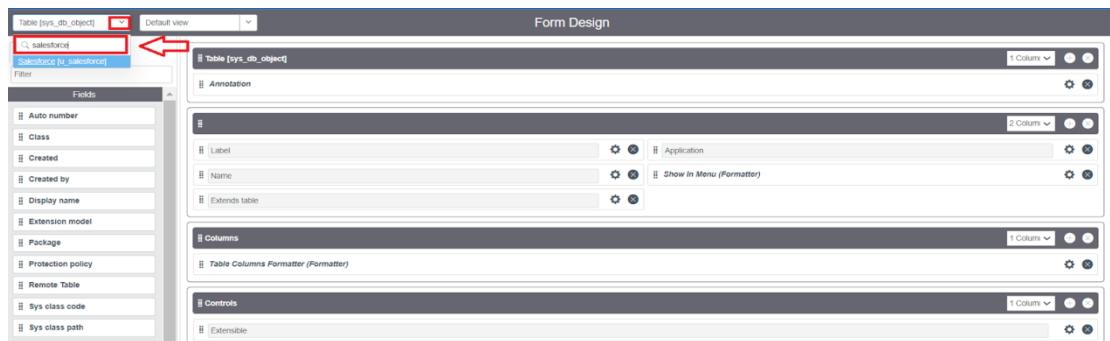
2.In Label Search for Salesforce and open .

The screenshot shows the ServiceNow Tables View: Tables page. A search bar at the top contains the text 'salesforce'. Below it is a table with columns: Name, Extends table, Extensible, and Updated. The table lists various system tables, including 'adaptive_auth_event' and 'agent_assist_recommendation'. A red arrow points to the 'Label' column where 'salesforce' is listed.

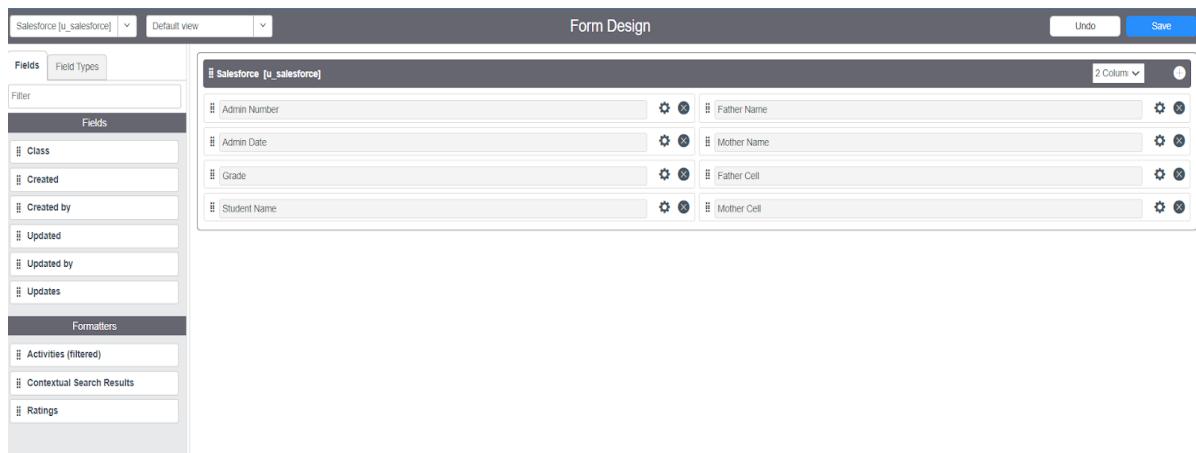
3.Right Click on top Toggle >> Configure >> Form Design.

The screenshot shows the ServiceNow Table - Salesforce configuration page. A context menu is open over the 'Label' field, with 'Configure' highlighted by a red arrow. The table below shows dictionary entries for the 'Salesforce' table, including fields like 'Class', 'Created by', 'Created', 'Sys ID', etc. The 'Configure' menu also includes options like 'Form Builder', 'Form Design', 'View', 'Create Favorite', 'Copy URL', 'Copy sys_id', 'Show XML', 'History', and 'Reload form'.

4.In drop down select Salesforce(u_salesforce).



5.Drag and drop the fields to the left side as below.

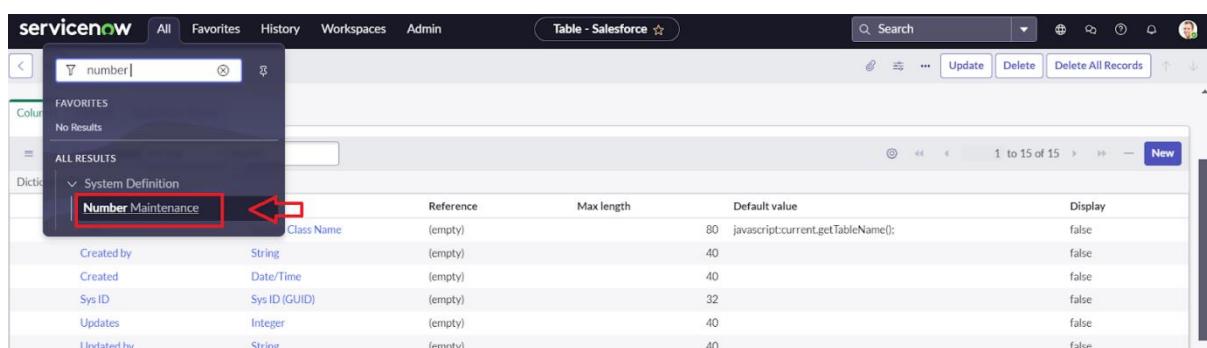


6.Save.

STEP 6: NUMBER MAINTAINANCE

CREATING NUMBER MAINTENANCE FOR ADMIN NUMBER

- All >> Number Maintenance >> New



- Fill the details >> Submit.

Number - SAL

* Table: Salesforce

Prefix: SAL

* Number: 1.000

Application: Global

Number of digits: 7

Update **Delete**

Related Links
Show Counter

STEP 7: PROCESS FLOW

CREATING PROCESS FLOW FOR ADMISSION TABLE

- All >> Process Flow>> New.
- Fill the Details as given Below

Flow Formatter New

* Table: Admission [u_admission]

* Name: New

Application: Global

* Label: New

Order:

Active:

Condition: Add Filter Condition | Add "OR" Clause

Admin Status is New

Description

Save
Insert **Insert and Stay** **Analyze Access**
Show File Properties
Move to Application...
Show Latest Update
Configure >
Export >
View >
Create Favorite
Copy URL
Copy sys_id
Show XML
History >
Reload form

Update **Delete**

- Right Click on toggle and click on the save .
- Replace the Name and Label as below and click on Insert on stay.

Flow Formatter In progress

* Table: Admission [u_admission]

* Name: In progress

Application: Global

* Label: In progress

Order:

Active:

Condition: Add Filter Condition | Add "OR" Clause

Admin Status is Join in progress

AND **OR**

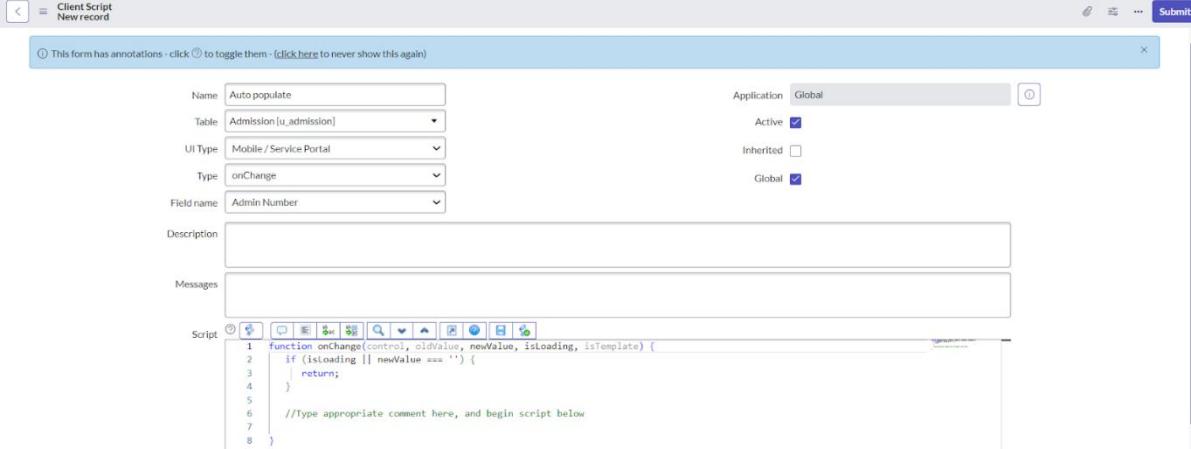
Save
Insert **Insert and Stay** **Analyze Access**
Show File Properties
Move to Application...
Show Latest Update
Configure >
Export >
View >
Create Favorite
Copy URL
Copy sys_id
Show XML
History >
Reload form

- Replace the Name and Label in order and click on Insert on stay.
Joined >> Rejected >> Rejoined >> Closed >> Cancelled.
- Order should be New >> InProgress >> Joined >> Rejected >> Rejoined >> Closed >> Cancelled.

STEP 8: CLIENT SCRIPT

CREATING “AUTO POPULATE” CLIENT SCRIPTS FOR ADMISSION TABLE

- All >> Client Scripts >> New.
- Fill the Details as given.



This form has annotations - click to toggle them - [\(click here to never show this again\)](#)

Name:	Auto populate	Application:	Global
Table:	Admission [u_admission]	Active:	<input checked="" type="checkbox"/>
UI Type:	Mobile / Service Portal	Inherited:	<input type="checkbox"/>
Type:	onChange	Global:	<input checked="" type="checkbox"/>
Field name:	Admin Number		
Description:			
Messages:			
Script:	<pre> 1 function onChange(control, oldValue, newValue, isLoading, isTemplate) { 2 if (isLoading newValue === '') { 3 return; 4 } 5 //Type appropriate comment here, and begin script below 6 7 8 }</pre>		

- Write the Code as below, Enable Isolate script and Save.
- ```
function onChange(control, oldValue, newValue, isLoading, isTemplate) {
if (isLoading || newValue === '') {
return;
}
```

```
//Type appropriate comment here, and begin script below
var a = g_form.getReference('u_admission_number');
g_form.setValue('u_admin_date',a.u_admin_date);
g_form.setValue('u_grade',a.u_grade);
g_form.setValue('u_student_name',a.u_student_name);
g_form.setValue('u_father_name',a.u_father_name);
g_form.setValue('u_mother_name',a.u_mother_name);
g_form.setValue('u_father_cell',a.u_father_cell);
g_form.setValue('u_mother_cell',a.u_mother_cell);
```

```
g_form.setDisabled('u_admin_date',a.u_admin_date);
g_form.setDisabled('u_grade',a.u_grade);
g_form.setDisabled('u_student_name',a.u_student_name);
```

```

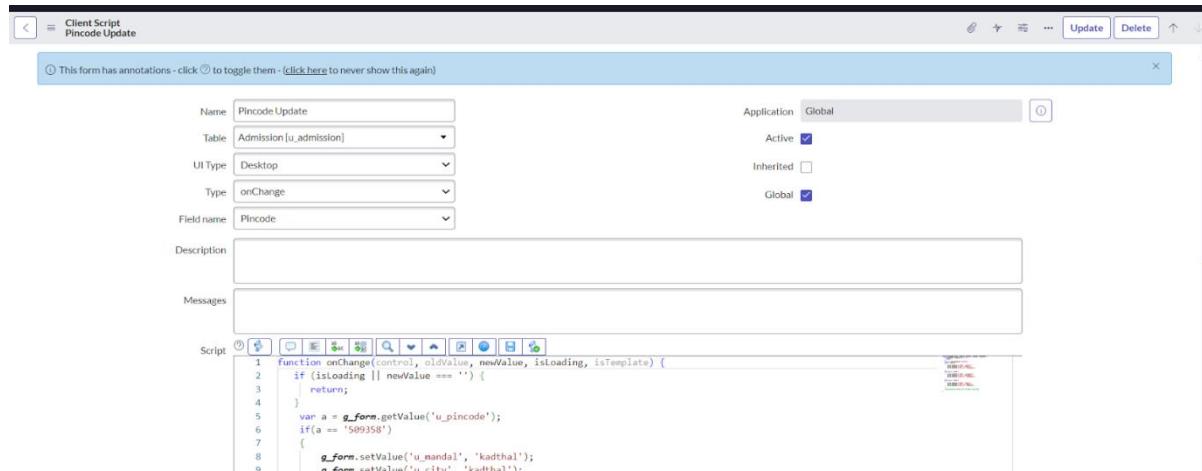
g_form.setDisabled('u_father_name',a.u_father_name);
g_form.setDisabled('u_mother_name',a.u_mother_name);
g_form.setDisabled('u_father_cell',a.u_father_cell);
g_form.setDisabled('u_mother_cell',a.u_mother_cell);
}

```

**Note:** Make sure the Field names should be the same as you created .

## CREATING “PINCODE UPDATE” CLIENT SCRIPTS FOR ADMISSION TABLE

- Fill the Details as given.



- Write the Code as below, Enable Isolate script and Save.

```

function onChange(control, oldValue, newValue, isLoading, isTemplate) {
 if (isLoading || newValue === "") {
 return;
 }
 var a = g_form.getValue('u_pincode');
 if(a == '509358')
 {
 g_form.setValue('u_mandal', 'kadthal');
 g_form.setValue('u_city', 'kadthal');
 g_form.setValue('u_district', 'RangaReddy');

 }
 else if(a == '500081')
 {
 g_form.setValue('u_mandal', 'karmanghat');
 g_form.setValue('u_city', 'karmanghat');
 g_form.setValue('u_district', 'RangaReddy');
 }
}

```

```

 }
 else if(a == '500079')
 {
 g_form.setValue('u_mandal', 'Abids');
 g_form.setValue('u_city', 'AsifNagar');
 g_form.setValue('u_district', 'Hyderabad');
 }

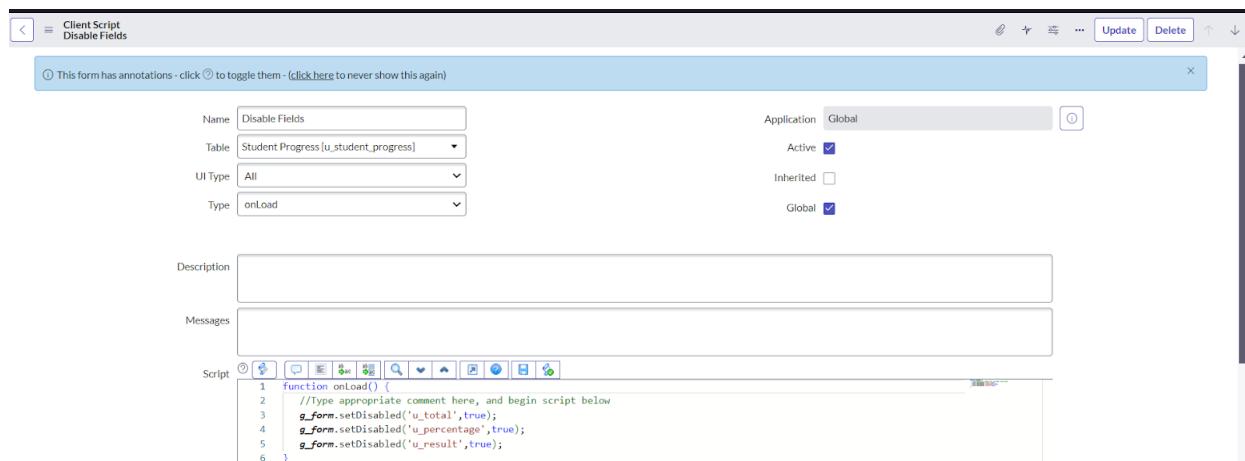
```

//Type appropriate comment here, and begin script below

```
}
```

## CREATING “DISABLE FIELDS” CLIENT SCRIPTS FOR STUDENT PROGRESS TABLE

- Fill the Details as given.



- Write the Code as below, Enable Isolate script and Save.

```
function onLoad() {
 //Type appropriate comment here, and begin script below
 g_form.setDisabled('u_total',true);
 g_form.setDisabled('u_percentage',true);
 g_form.setDisabled('u_result',true);
}
```

## CREATING “TOTAL UPDATE” CLIENT SCRIPTS FOR STUDENT PROGRESS TABLE

- Fill the Details as given.

The screenshot shows the ServiceNow Client Script configuration interface. The top navigation bar includes 'servicenow', 'All', 'Favorites', 'History', 'Workspaces', and 'Admin'. The title bar says 'Client Script - Total Up...'. The main area has sections for 'Name' (Total Update), 'Table' (Student Progress [u\_student\_progress]), 'UI Type' (All), 'Type' (onChange), 'Field name' (Social), 'Description', 'Messages', and 'Script'. The 'Script' section 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 if (newValue){
8 var a = parseInt(g_form.getValue('u_telugu'));
9 var b = parseInt(g_form.getValue('u_hindi'));
10 var c = parseInt(g_form.getValue('u_english'));
11 var d = parseInt(g_form.getValue('u_maths'));
12 var e = parseInt(g_form.getValue('u_science'));
13 var f = parseInt(g_form.getValue('u_social'));
14 var Total = parseInt(a+b+c+d+e+f);
15 g_form.setValue('u_total', Total);
16 }
17 }

```

At the bottom left of the script editor, there is a checkbox labeled 'Isolate script' which is checked.

- Write the Code as below, Enable Isolate script and Save.  
function onChange(control, oldValue, newValue, isLoading, isTemplate) {  
if (isLoading || newValue === "") {  
return;  
}

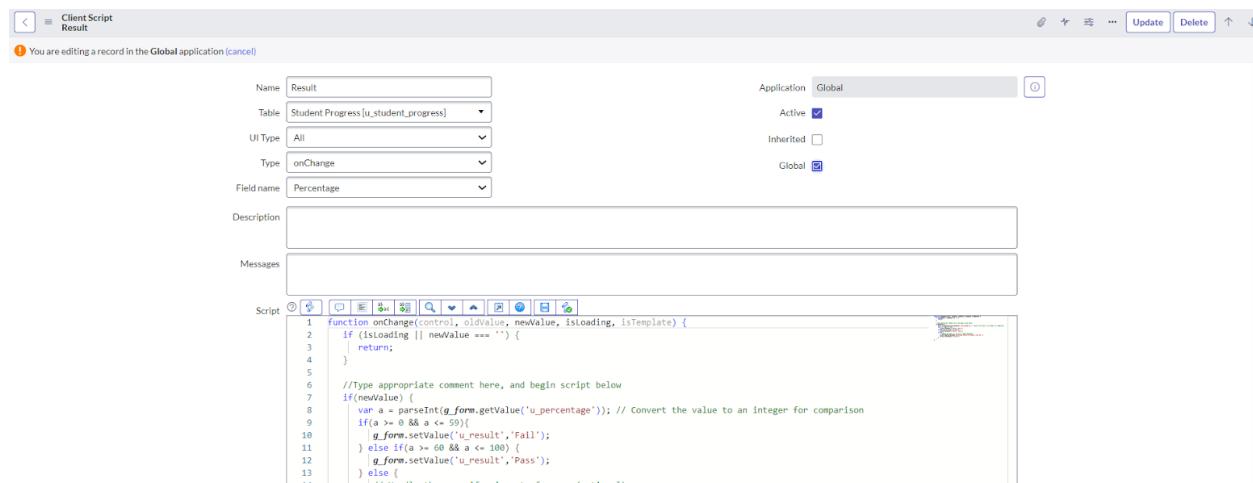
```

//Type appropriate comment here, and begin script below
if (newValue){
var a = parseInt(g_form.getValue('u_telugu'));
var b = parseInt(g_form.getValue('u_hindi'));
var c = parseInt(g_form.getValue('u_english'));
var d = parseInt(g_form.getValue('u_maths'));
var e = parseInt(g_form.getValue('u_science'));
var f = parseInt(g_form.getValue('u_social'));
var Total = parseInt(a+b+c+d+e+f);
g_form.setValue('u_total', Total);
}
}

```

## CREATING “RESULT” CLIENT SCRIPTS FOR STUDENT PROGRESS TABLE

- Fill the Details as given.



- Write the Code as below, Enable Isolate script and Save.

```
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_percentage')); // Convert the value to an
integer for comparison
 if(a >= 0 && a <= 59){
 g_form.setValue('u_result','Fail');
 } else if(a >= 60 && a <= 100) {
 g_form.setValue('u_result','Pass');
 } else {
 // Handle the case if a is out of range (optional)
 g_form.addErrorMessage('Percentage should be between 0 and 100.');
 g_form.clearValue('u_result');
 }
 }
}
```

## CREATING “PERCENTAGE” CLIENT SCRIPTS FOR STUDENT PROGRESS TABLE

- Fill the Details as given.

The screenshot shows the configuration of a client script named 'Percentage'. The script is triggered on change ('onChange') for the 'Total' field in the 'Student Progress [u\_student\_progress]' table. The script calculates a percentage by dividing the total by 600 and multiplying by 100, then sets the value of the 'u\_percentage' field.

```

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/600)*100;
 g_form.setValue('u_percentage',Percentage+'%');
}

```

- Write the Code as below, Enable Isolate script and Save.
- ```

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/600)*100;
    g_form.setValue('u_percentage',Percentage+'%');
}

```

RESULT:

The ServiceNow-based Educational Management System successfully demonstrates the automation of student admission, academic progress, and information management. Data is centralized, workflows are simplified, and administrators can easily generate reports and monitor academic progress.

This screenshot shows a 'New record' form for a student. It includes fields for Admin Number (SAL0001078), Admin Date, Grade, and Student Name. On the right side, there are fields for Father Name, Mother Name, Mother Cell, and Father Cell. A 'Submit' button is located at the bottom left of the form area.

The image displays two screenshots of ServiceNow forms. The top screenshot shows the 'Admission' form with fields for Student Name, Father Name, Mother Name, Admin Date, Grade, Fee, Father Cell, Mother Cell, Admin Status, and Comments. It includes a workflow step bar with stages: New, In progress, Joined, Rejected, Rejoined, Closed, and Cancelled. The bottom screenshot shows the 'Student Progress' form with fields for Telugu, Hindi, English, Maths, Science, Social, Total, Percentage, and Result.

FEATURES:

1. Centralized management of student and teacher information on a single ServiceNow platform.
2. Automated workflows for admissions, attendance, and progress tracking.
3. Real-time data access with reporting and dashboard analytics.
4. User-friendly interface for administrators, faculty, and students to improve efficiency.

CONCLUSION:

The project “Educational Organisation Using ServiceNow” effectively addresses the common challenges faced by educational institutions in handling administrative and academic data. Through automation and centralization, it enhances efficiency, transparency, and collaboration across departments.

By leveraging ServiceNow’s powerful workflow and reporting tools, the system reduces human error, saves time, and supports data-driven decisions. This project stands as a practical example of how modern cloud platforms like ServiceNow can revolutionize educational management systems, paving the way for a smarter and more connected academic environment.