

Educational Organisation Using Service Now

Team Id: NM2025TMID14575

Team Members: 4

Team Leader: MOHAMED HUSSAIN Y

Team Member 1 : VISHVA G

Team Member 2 : DAVID J

Team Member 3:

TAMIL VANAN G

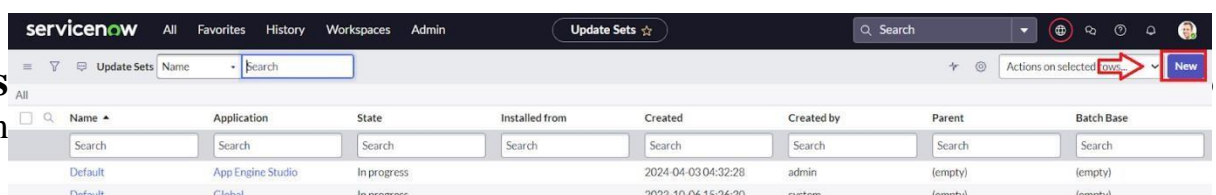
Problem Statement :

Enhancing Operational Efficiency in an Educational Institution Using ServiceNow Educational institutions often face challenges in managing IT services, administrative workflows, student support, and facility operations due to fragmented systems and manual processes. These inefficiencies lead to delayed responses, poor user experience, and increased operational costs.

Objective :

To implement ServiceNow as a centralized platform that streamlines service management, automates workflows, improves response times, and enhances transparency across academic and administrative functions.

**Skills
stream**



servicenow							
All Favorites History Workspaces Admin							
Update Sets ☆							
Search							
Update Sets Name Search							
Actions on selected loss New							
Name	Application	State	Installed from	Created	Created by	Parent	Batch Base
Search	Search	Search	Search	Search	Search	Search	Search
Default	App Engine Studio	In progress		2024-04-03 04:32:28	admin	(empty)	(empty)
Default	Global	In progress		2023-10-06 15:26:30	system	(empty)	(empty)

Milestone 1 : Setting up servicenow instance

Activity 1:

1. Create a ServiceNow developer account on the ServiceNow Developer website.
2. After signing in, go to the Personal Developer Instance section.
3. Select Request Instance to spin up a new ServiceNow environment.
4. Provide the necessary details and submit your request.
5. Once the instance is ready, you'll receive an email with the login information.
6. Use the credentials provided to sign in to your new ServiceNow instance.
7. After logging in, you can start exploring ServiceNow.

Educational Organisation Using Service Now

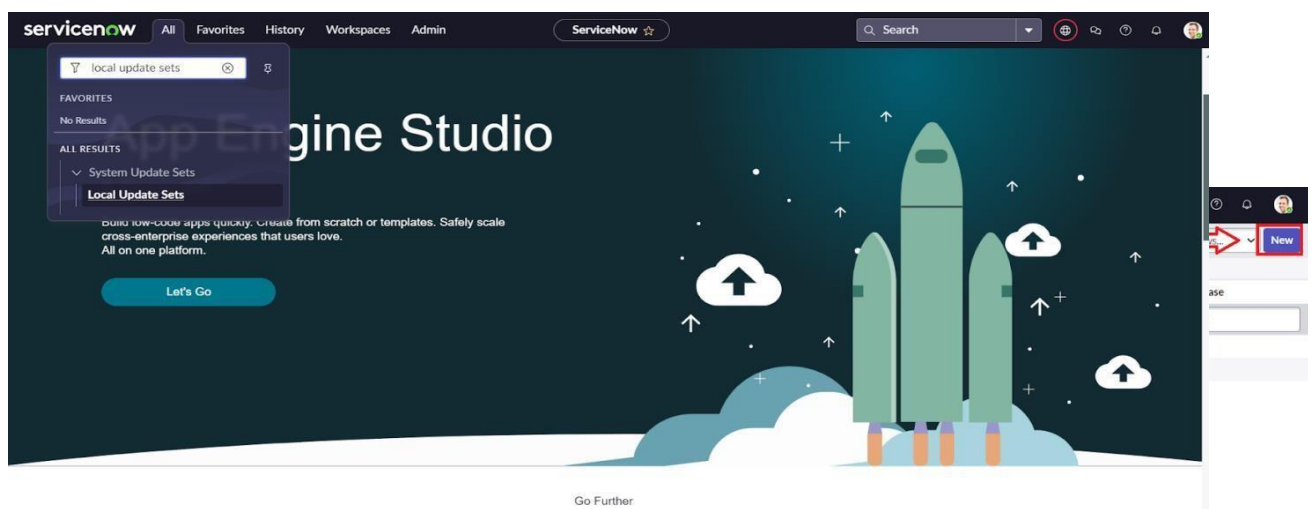
The Educational Management System is a comprehensive platform designed to streamline administrative tasks within educational institutions. It facilitates efficient management of student and teacher data, simplifies the admission process, and provides tools for monitoring student progress.

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.

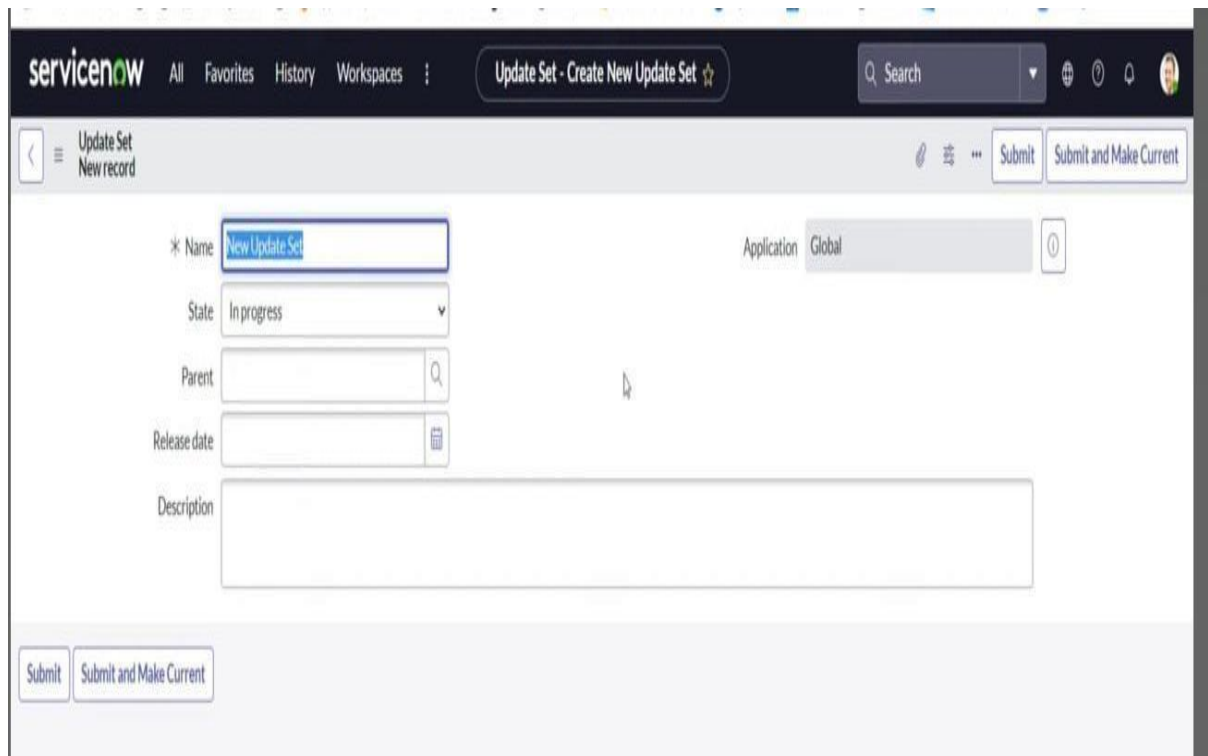
Creating a Update Set

1. Click on All >> Local update sets .



2. Click on new

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



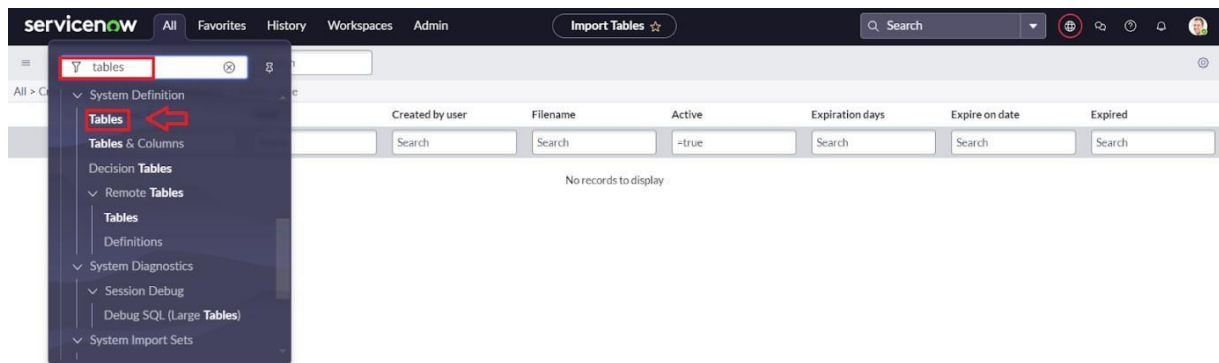
The screenshot shows the ServiceNow interface for creating a new update set. The header bar includes the ServiceNow logo, navigation links (All, Favorites, History, Workspaces), a search bar, and a breadcrumb trail: 'Update Set - Create New Update Set'. Below the header, the page title is 'Update Set' with a sub-header 'New record'. The form contains the following fields:

- Name:** A text field with the value 'New Update Set'.
- Application:** A dropdown menu with the value 'Global'.
- State:** A dropdown menu with the value 'In progress'.
- Parent:** A text field with a search icon.
- Release date:** A text field with a calendar icon.
- Description:** A large text area.

At the bottom of the form, there are two buttons: 'Submit' and 'Submit and Make Current'.

Creating a Table Creating Salesforce Table

- All >> Tables.



- Click on new

The screenshot shows the 'Tables' page in ServiceNow. The top navigation bar is the same as in the previous image. Below it, there is a search bar and a 'New' button, which is highlighted with a red box. The main content area displays a table with the following columns: 'Label', 'Name', 'Extends table', 'Extensible', and 'Updated'. The table contains several rows of data, including 'Adaptive Authentication Event', 'Agent Assist Recommendation', 'MID Server File', 'Record Producer Configuration', 'Search Actions', 'AI Search ACL Overrides', and 'AI Search Active Table Ingestion Tracker'.

Label	Name	Extends table	Extensible	Updated
Search	Search	Search	Search	Search
Adaptive Authentication Event	adaptive_auth_event	(empty)	false	2024-03-29 07:44:59
Agent Assist Recommendation	agent_assist_recommendation	Application File	false	2023-10-06 15:44:09
MID Server File	agent_file	(empty)	false	2023-10-06 15:29:08
Record Producer Configuration	aisa_rp_config	Application File	false	2023-10-06 15:47:19
Search Actions	aisa_ui_action	Application File	false	2023-10-06 15:47:19
AI Search ACL Overrides	ais_acl_overrides	Application File	false	2023-10-06 15:15:20
AI Search Active Table Ingestion Tracker	ais_active_table_ingestion_tracker	(empty)	false	2023-10-06 15:15:18

- Enter the Label(Anything you want): Salesforce >> Click on Name it will Automatically generate Api name.

- Create columns as given below, Double Click on Column label and Enter the Column labels and click on the tick mark >> Give Type as given .

X

Table Salesforce

This record is in the Global application, but Educational Organisation is the current application. To edit this record click [here](#).

Admin Date	Date	(empty)	40	false
Admin Number	String	(empty)	40	javascript:getNextObjNumberPadded();
Father Cell	String	(empty)	40	false
Father Name	String	(empty)	40	false
Grade	Choice	(empty)	40	false
Mother Cell	String	(empty)	40	false
Mother Name	String	(empty)	40	false
Student Name	String	(empty)	40	false

- For “Admin Number” Give Display as True and right click on the toggle bar on top >> save.

Table Salesforce

Columns Controls Application Access

Table Columns for text Search

Dictionary Entries

Column label	Type	Reference	Max length	Display
Class	System Class Name	(empty)	80	false
Created by	String	(empty)	40	false
Created	Date/Time	(empty)	40	false
Sys ID	Sys ID (GUID)	(empty)	32	false
Updates	Integer	(empty)	40	false
Updated by	String	(empty)	40	false
Updated	Date/Time	(empty)	40	false

Save

- Analyze Access
- Show File Properties
- Move to Application...
- Show Latest Update
- Create Child Table
- Show Dictionary Record
- Configure >
- Export >
- View >
- Create Favorite
- Copy URL
- Copy sys_id
- Show XML
- History >
- Reload form

- Click on controls >> Enable Extensible.

Columns **Controls** Application Access

Extensible ☒ Live feed ☐

Prefix SAL

Number 1,000

Number of digits 7

Create access controls ☒

* User role u_salesforce_user

- Click on “Admin Number” column, In Related Links Click on Advanced View >> Default View (Enable Use dynamic default) >> select Get Next Padded Number in Dynamic default value >> Update .

Choice List Specification Calculated Value **Default Value**

Use dynamic default ☒

Dynamic default value Get Next Padded Number

- Click on “Grade” Column >> Click on Choices and give Label, Value and Sequence as given below.

Table Admission

Columns Controls Application Access

Table Columns for text Search

Dictionary Entries

Column label	Type	Reference	Max length	Default value	Display
Sys ID	Sys ID (GUID)	(empty)	32		false
Admin Status	Choice	(empty)	40		false
Admission Number	Reference	Salesforce	32		false
Area	String	(empty)	40		false
City	String	(empty)	40		false
Comments	String (Full UTF-8)	(empty)	255		false
District	String	(empty)	40		false
Fee	Price	(empty)	20		false
House No	String	(empty)	40		false
Mandal	String	(empty)	40		false
Pincode	Choice	(empty)	40		false
Purpose of Join	Choice	(empty)	40		false
School	Choice	(empty)	40		false
School Area	Choice	(empty)	40		false
Class	System Class Name	(empty)	80	javascript:current.getTableName();	false

- Create choice for Admin Status as:

Dictionary Entry Admin Status

Create Choice List Delete Column Update

Related Links

Show Table Run Point Scan Advanced view

Access Controls Choices (7) Attributes Labels (1) Dictionary Overrides

Label Search

Choices

Label	Value	Language	Sequence	Inactive	Updated
New	New	en	1	false	2024-04-02 21:10:25
Join in progress	In progress	en	2	false	2024-04-02 21:11:03
Joined	Joined	en	3	false	2024-04-02 21:11:26
Rejected	Rejected	en	4	false	2024-04-02 21:12:00
Closed	Closed	en	5	false	2024-04-02 21:13:05
Rejoined	Rejoined	en	6	false	2024-04-02 21:13:08
Cancelled	Cancelled	en	7	false	2024-04-02 21:13:27

Insert a new row...

- Create choice for Pincode as:

Access Controls Choices (3) Attributes Labels (1) Dictionary Overrides

Label Search

Choices

Label	Value	Language	Sequence	Inactive	Updated
509358	509358	en	1	false	2024-04-02 21:15:19
500079	500079	en	2	false	2024-04-02 21:15:46
500081	500081	en	3	false	2024-04-02 21:16:05

Insert a new row...

- Create choice for Purpose of Join as:

Access Controls Choices (3) Attributes Labels (1) Dictionary Overrides

Label Search

Choices

Label	Value	Language	Sequence	Inactive	Updated
Tuition	Tuition	en	1	false	2024-04-02 21:17:09
Coaching	Coaching	en	2	false	2024-04-02 21:17:31
Teacher	Teacher	en	3	false	2024-04-02 21:17:53

Insert a new row...

- Create choice for School as:

Form Layout

Configuring Table form for Student Progress Table

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

Table student progress			
✖	Telugu	String	(empty) 40
✖	Total	String	(empty) 40
+	Insert a new row...		

Update Delete Delete All Records

Related Links

- [Design Form](#)
- [Layout Form](#)
- [Layout List](#)
- [Show Form](#)
- [Show List](#)
- [Show Schema Map](#)
- [Add to Service Catalog](#)
- [Run Point Scan](#)
- [Explore REST API](#)

- Click on Admission Number [+].

Configuring Table form

Available

- Admission Number [+]
- Created
- Created by
- Updated
- Updated by
- Updates
- {% begin_split %}
- {% split %}
- {% end_split %}
- * Annotation
- * Chart
- Activities (filtered)
- Contextual Search Results
- Ratings
- Attachments
- Goal relationships

Selected

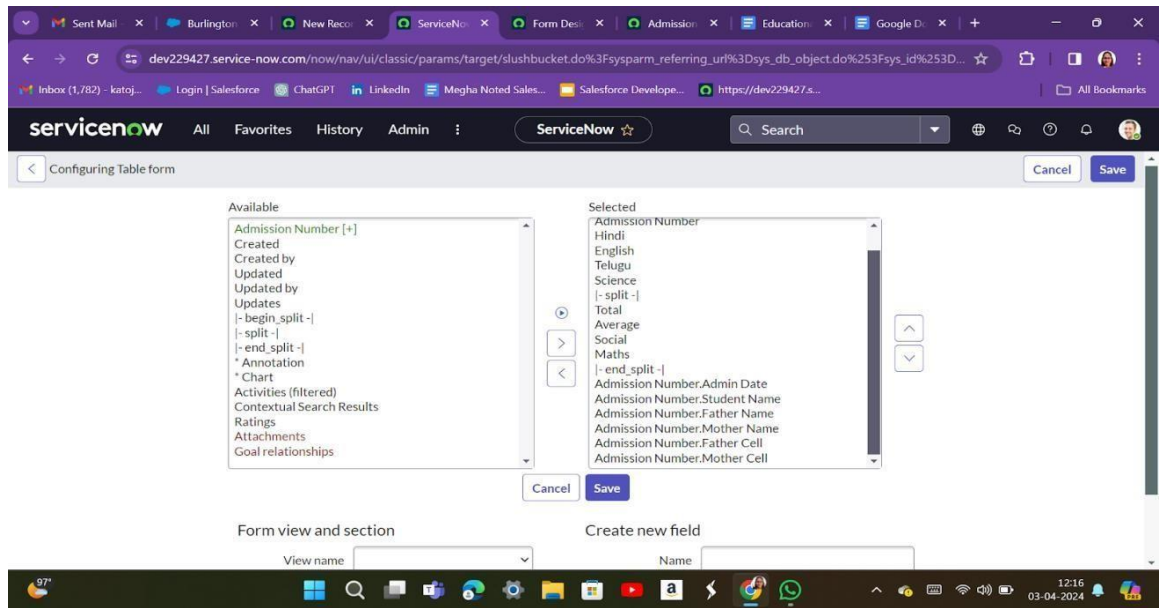
- {% begin_split %}
- Admission Number
- Hindi
- English
- Telugu
- Science
- {% split %}
- Total
- Average
- Social
- Maths
- {% end_split %}

Form view and section

View name: Default view

Create new field: Name

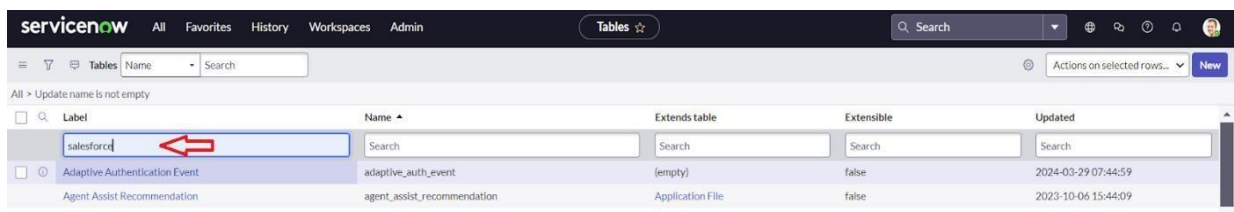
- Select below Admission Number fields in Available side and send it to selected side as below >> save.



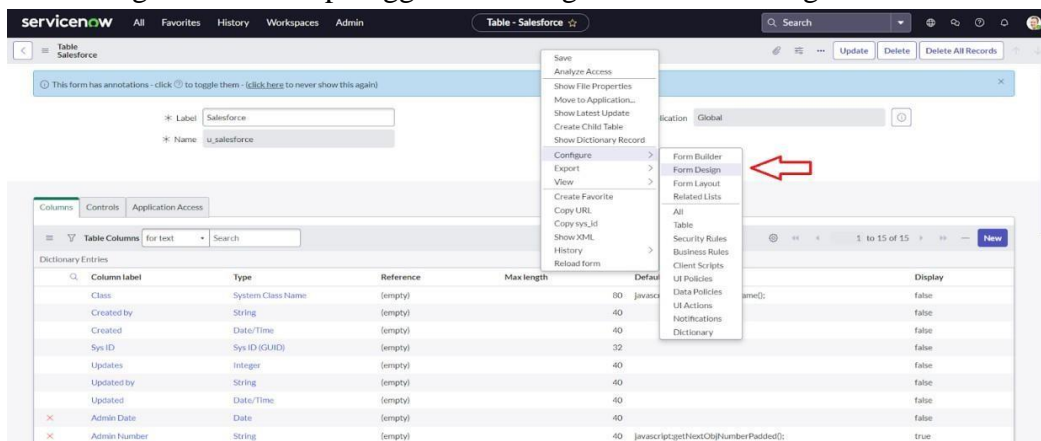
Form Design

Creating Form Design for Salesforce Table

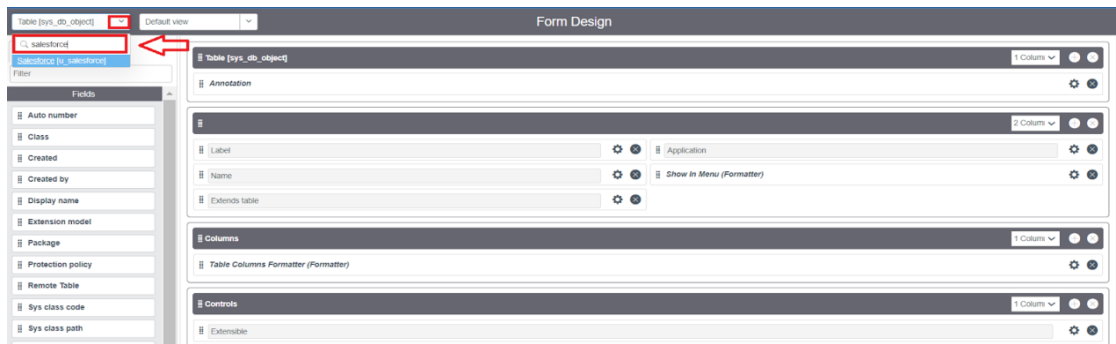
1. All >> System Definition >> Tables .
2. In Label Search for Salesforce and open .



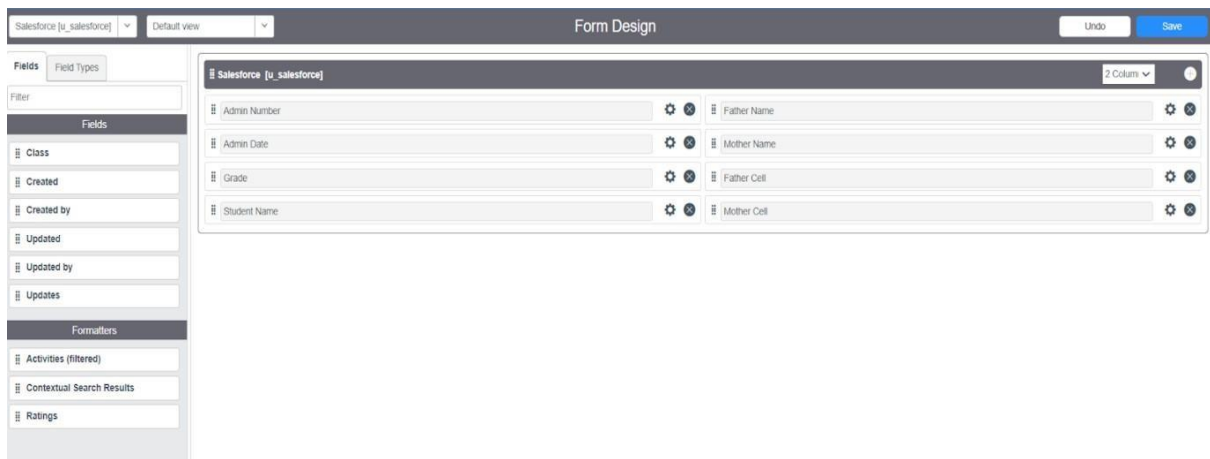
3. Right Click on top Toggle >> Configure >> Form Design.



4. In drop down select Salesforce(u_salesforce).



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



6. Save.

Creating Form Design for Admission Table

Follow the same steps as Activity1, Configure the fields as below and Save.

The screenshot shows the 'Form Design' interface for the 'Admission' table. The left sidebar contains a 'Fields' list with categories like 'Admin Number', 'Class', 'Created', 'Updated', and 'Formatters'. The main area displays the form structure with sections: 'Admission [u_admission]', 'Process Flow (Formater)', 'Admission Details' (with fields like Admission Number, Admin Date, Purpose of join, Grade, Student Name, Fee, Father Name, Father Cell, Mother Name, Mother Cell, Admin Status), 'Comments', 'School Details' (with School Area and School), and 'Address' (with Pincode, Area, Mandal, City, House No, and District). Each section has a '1 Col' or '2 Col' dropdown and a 'Save' button.

Creating Form Design for Student progress Table

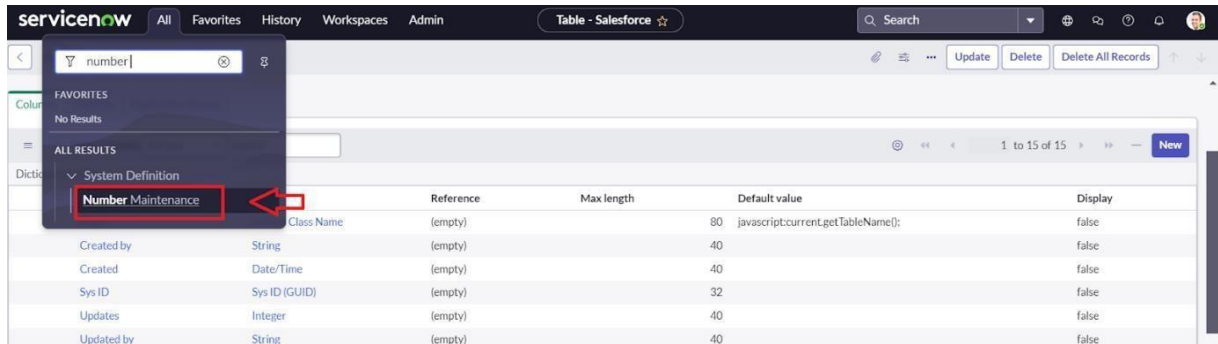
Follow the same steps as Activity1, Configure the fields as below and Save.

The screenshot shows the 'Form Design' interface for the 'Student Progress' table. The left sidebar contains a 'Fields' list with categories like 'Class', 'Created', 'Updated', and 'Formatters'. The main area displays the form structure with sections: 'New Section' (with Admission Number), 'Admission Details' (with fields like Admission Number Grade, Admission Number Father Name, Admission Number Mother Name, Admission Number Student Name, Admission Number Father Cell, and Admission Number Mother Cell), and 'Student Progress' (with fields like Telugu, Hindi, English, Maths, Science, Total, Percentage, and Result). Each section has a '1 Col' or '2 Col' dropdown and a 'Save' button.

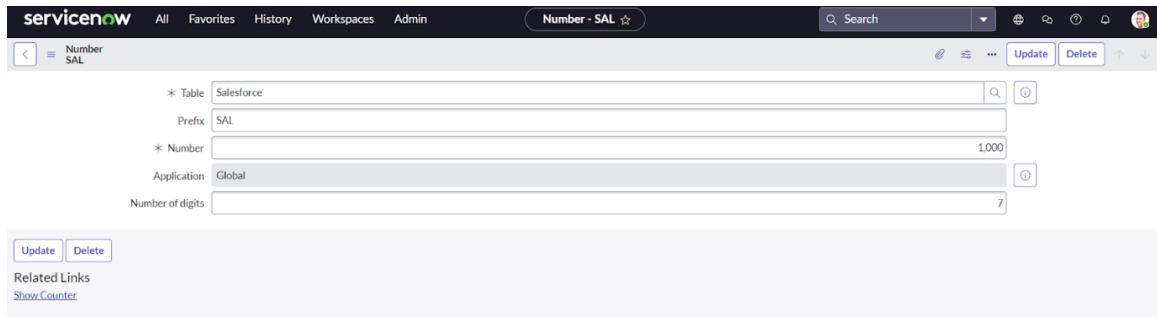
Number Maintenance

Creating Number Maintenance for Admin Number

- All >> Number Maintenance >> New



- Fill the details >> Submit.



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:

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

Description:

Update Delete

- 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.

Client Script

Creating “Auto populate” Client Scripts for Admission Table

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

Client Script
New record

This form has annotations - click ⓘ to toggle them - (click here to never show this again)

Name: Auto populate

Table: Admission [u_admission]

UI Type: Mobile / Service Portal

Type: onChange

Field name: Admin Number

Application: Global ⓘ

Active: ☒

Inherited: ☐

Global: ☒

Description:

Messages:

Script ⓘ

```
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 }
```

- 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.

The screenshot shows the 'Client Script' configuration window for a field named 'Pincode'. The window has a title bar 'Client Script Pincode Update' and buttons for 'Update', 'Delete', and navigation arrows. A blue notification bar at the top states: 'This form has annotations - click ⓘ to toggle them - (click here to never show this again)'. The configuration fields are as follows:

- Name: Pincode Update
- Table: Admission [u_admission]
- UI Type: Desktop
- Type: onChange
- Field name: Pincode
- Application: Global
- Active: ☒
- Inherited: ☐
- Global: ☒

Below these fields are empty text areas for 'Description' and 'Messages'. At the bottom is a 'Script' editor with a toolbar and the following code:

```
1 function onChange(control, oldValue, newValue, isLoading, isTemplate) {
2   if (isLoading || newValue === '') {
3     return;
4   }
5   var a = g_form.getValue('u_pincode');
6   if(a == '509358')
7   {
8     g_form.setValue('u_mandal', 'kadthal');
9     g_form.setValue('u_city', 'kadthal');
```

- 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');
  }
}
```

Creating “Disable Fields” Client Scripts for Student progress Table

- Fill the Details as given.

The screenshot shows the ServiceNow Client Script editor for a script named 'Disable Fields'. The details are as follows:

- Name:** Disable Fields
- Table:** Student Progress [u_student_progress]
- UI Type:** All
- Type:** onLoad
- Application:** Global
- Active:** ☒
- Inherited:** ☐
- Global:** ☒

The Description and Messages fields are empty. The Script area contains the following code:

```
1 function onLoad() {  
2     //Type appropriate comment here, and begin script below  
3     g_form.setDisabled('u_total',true);  
4     g_form.setDisabled('u_percentage',true);  
5     g_form.setDisabled('u_result',true);  
6 }
```

- 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 editor for a script named 'Total Update'. The details are as follows:

- Name:** Total Update
- Table:** Student Progress [u_student_progress]
- UI Type:** All
- Type:** onChange
- Field name:** Social
- Application:** Global
- Active:** ☒
- Inherited:** ☐
- Global:** ☒

The Description and Messages fields are empty. The Script area contains the following code:

```
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     var a = parseInt(g_form.getValue('u_telugu'));  
7     var b = parseInt(g_form.getValue('u_hindi'));  
8     var c = parseInt(g_form.getValue('u_english'));  
9     var d = parseInt(g_form.getValue('u_maths'));  
10    var e = parseInt(g_form.getValue('u_science'));  
11    var f = parseInt(g_form.getValue('u_social'));  
12    var Total = parseInt(a+b+c+d+e+f);  
13    g_form.setValue('u_total', Total);  
14 }  
15  
16  
17
```

At the bottom, the 'Isolate script' checkbox is checked.

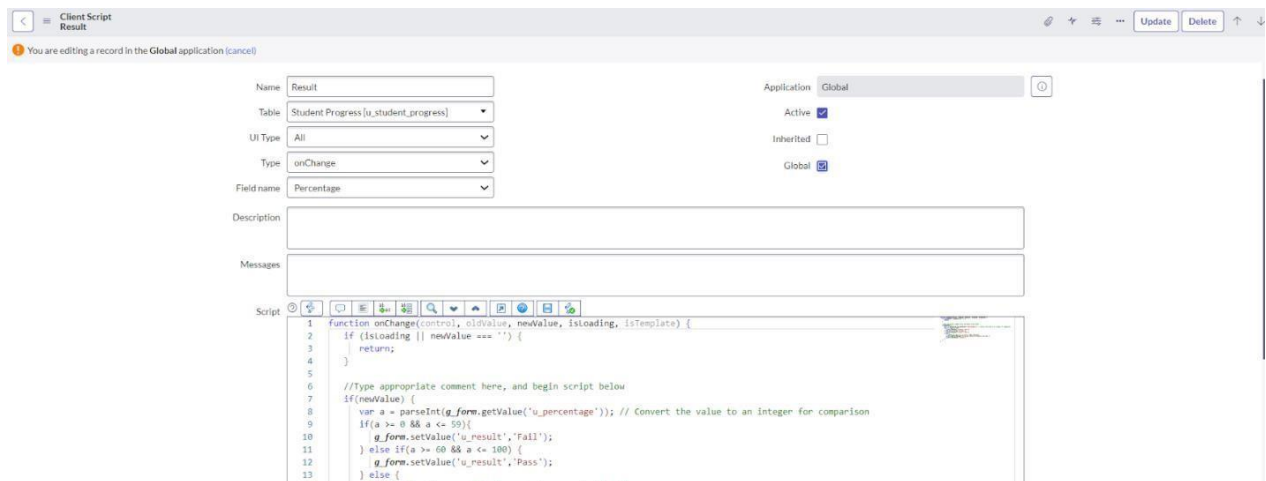
- Write the Code as below, Enable Isolate script and Save.

```
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.');
```

Creating “Percentage” Client Scripts for Student progress Table

- Fill the Details as given.

The screenshot shows the Salesforce Client Script configuration page for a field named "Percentage". The configuration details are as follows:

- Name:** Percentage
- Table:** Student Progress[u_student_progress]
- UI Type:** All
- Type:** onChange
- Field name:** Total
- Application:** Global
- Active:** ☒
- Inherited:** ☐
- Global:** ☒

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     var total = g_form.getValue('u_total');
8     var Percentage = (total/600)*100;
9     g_form.setValue('u_percentage',Percentage+'%');
10 }
```

The **Isolate script** checkbox 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
var Total = g_form.getValue('u_total');
var Percentage = (Total/600)*100;
g_form.setValue('u_percentage',Percentage+'%');
```

Result

Salesforce
New record

Admin Number

Admin Date

Grade

Student Name

Father Name

Mother Name

Mother Cell

Father Cell

Submit

Admission
New record

New
In progress
Joined
Rejected
Rejoined
Closed
Cancelled

Admin Number

Purpose of Join

Student Name

Father Name

Mother Name

Comments

Admin Date

Grade

Fee \$

Father Cell

Mother Cell

Admin Status

School Details
Address

School Area

School

Submit

New Section
New record

Admission Number

Grade

Student Name

Father Name

Mother Name

Father Cell

Mother Cell

Student Progress

Telugu

Hindi

English

Maths

Science

Social

Total

Percentage

Result

Submit

Conclusion:

The implementation of ServiceNow in an educational organization demonstrates how technology can simplify and improve institutional processes. By digitizing workflows such as student management, faculty support, and administrative services, the platform enhances efficiency, reduces manual errors, and ensures timely responses.

This project shows that ServiceNow is not limited to corporate environments but can also provide significant value in education by centralizing services, enabling self-service portals, and streamlining communication between students, teachers, and administrators. The use of automation and dynamic workflows ensures better resource utilization and a smoother experience for all stakeholders.

Overall, the project highlights that adopting ServiceNow in educational institutions leads to improved productivity, better transparency, and a modernized service delivery model. This approach not only supports day-to-day operations but also helps the organization move closer to digital transformation goals in the education sector.